

PESQUISAS APLICADAS EM SISTEMAS DE INFORMAÇÃO

Antonio Oseas de Carvalho Filho,
Deborah Maria Vieira Magalhães,
Flavio Henrique Duarte de Araujo,
Francisco Airton Pereira da Silva,
Romuere Rodrigues Veloso e Silva.

ORGANIZADORES



Antonio O. Carvalho Filho, Deborah M. V. Magalhães,
Flávio H. D. Araújo, Francisco A. P. Silva, Romuere R. V. Silva

Pesquisas Aplicadas em Sistemas de Informação

© 2021 Antonio O. Carvalho Filho, Deborah M. V. Magalhães, Flávio H. D. Araújo, Francisco A. P. Silva, Romuere R. V. Silva & Sistemas de Informação - UFPI
Qualquer parte desta publicação pode ser reproduzida, desde que citada a fonte.

Catalogação Informada pelos Autores

Carvalho-Filho, A. O.

Pesquisas Aplicadas em Sistemas de Informação. / Antonio O. Carvalho Filho, Deborah M. V. Magalhães, Flávio H. D. Araújo, Francisco A. P. Silva, Romuere R. V. Silva. – Picos-PI: Sistemas de Informação - UFPI, 2021.

Bibliografia.

ISBN 978-65-00-27116-4.

1. Sistemas de Informação. 2. Pesquisas Aplicadas. 3. Computação.

Prefácio

Caro leitor,

Você gosta de assuntos que envolvem tecnologias emergentes? Antes de responder esta dúvida, nos permita uma breve apresentação. Nós, autores e organizadores desta obra integramos o corpo docente do curso de Bacharelado em Sistemas de Informação da Universidade Federal do Piauí (UFPI), campus Senador Helvídio Nunes de Barros (CSHNB). O CSHNB fica localizado na cidade de Picos, na região centro-sul do Piauí, em pleno seminário nordestino. Picos é uma cidade aconchegante de pessoas receptivas, produz um mel maravilhoso e uma castanha de caju de sabor único.

Sabe aquela pergunta inicial, pois bem, nós também gostamos, para falar a verdade, nós somos loucos por tecnologia. Mas aí você pode estar se perguntando, o que um curso de tecnologia, em pleno semiárido nordestino, tem a oferecer em termos de tecnologia? Essa é uma excelente pergunta, e melhor ainda, é a resposta, a qual te convidamos a conhecer em nossa obra. Ah! outra coisa, nós não gostamos de dar spoilers, mas nesse caso, é por um bom motivo. Em nossa obra você será emergido num conjunto de trabalhos desafiadores produzidos por nossos brilhantes e esforçados alunos, que demonstram vários métodos computacionais capazes de automatizar tarefas fundamentais para as nossas vidas.

Mas nem tudo são flores, nós e nossos alunos encontramos muitas limitações, mas nem assim, nos deixamos abater, de correr atrás, de buscar soluções, e com muito esforço e dedicação, conseguimos desenvolver soluções robustas, reconhecidas internacionalmente, como nós carinhosamente costumamos dizer, do Piauí para o mundo.

Cabe aqui, um agradecimento especial a todos os alunos envolvidos nesse projeto, sabemos o quanto difícil e árduo é o caminho da ciência, principalmente nos dias atuais, mas o que nos motiva a continuar são vocês, que mesmo com todas as dificuldades, não desanimam e sempre levantam mais fortes e dedicados.

Por fim, ficamos contentes que você chegou até aqui, sinal que gostou de conhecer um pouco sobre nós. Ficaremos imensamente gratos por conhecer nossa obra detalhadamente, com certeza te ajudaremos a conhecer novos conceitos e perspectivas nessa leitura.

Nosso mais singelo e verdadeiro, agradecimento!

Oseas, Deborah, Flávio, Airton & Romuere.

Sumário

I	Processamento de dados médicos	5
1	Análises de Mapas de Texturas como Entradas pra Extração de Características em Redes Neurais Convolucionais	7
2	Mapas de textura como entrada em CNNs 3D aplicados para classificar nódulos em imagens de TC	23
3	Segmentação do Disco Óptico em Imagens Retinoscópicas Utilizando a Rede Neural Convolucional U-Net	37
4	Determinação automática do glaucoma utilizando índices de diversidade taxonômica	51
5	Análise de Textura de Nódulos Pulmonares para Sistemas CAD Utilizando Índice de Diversidade Funcional	65
6	Metodologia automática para classificação de lesões de mama utilizando Transfer Learning	79
7	Método Automático para Classificação de Imagens Histopatológicas da Mama Utilizando Transfer Learning e Análise de Textura	95
8	Classificação automática de tecidos mamários em malignos e benignos utilizando Índices de diversidade	115
9	Análise filogenética aplicada a exploração de textura de imagens de pele para auxílio no diagnóstico do melanoma	135
10	Classificação de imagens em COVID-19 e Não-COVID-19 usando CNN para extração de características e múltiplos classificadores	149
11	Uma Abordagem Baseada em <i>Capsule Network</i> para a Identificação de Glaucoma em Imagens de Retina	161
12	Índices COVID: Uma Abordagem Baseada em Textura para Classificar Lesões Pulmonares	181
13	Diagnóstico Automático da Presença de Exsudatos em Imagens de Retina Utilizando Aprendizado Profundo	203
14	Classificação automática de texturas de imagens de pele em melanoma e não melanoma utilizando índices de diversidade	215
15	Segmentação de Nódulos Pulmonares em Imagens de Tomografia Computadorizada	233
16	Redes Neurais Convolucionais para Identificação de Metástase em Imagens Histopatológicas.	247
17	Redes Neurais Convolucionais Aplicadas na Detecção de Câncer Mamário em Imagens Histológicas	261
18	Uso de Transferência de Aprendizado para Detecção Automática de Melanoma	277
19	Detecção e Segmentação do Disco Óptico em Imagens da Retina	293
II	Avaliação de Desempenho de Sistemas	307
20	Avaliação de Desempenho e Disponibilidade da Mobile Edge Computing usando Redes de Petri Estocásticas	309
21	Uma Plataforma Distribuída para Avaliação de Desempenho em Aplicativos Móveis	333
22	Avaliação de Desempenho de Placas de Rede 10GbE: Uma Estratégia Baseada em Análise de Sensibilidade	359
23	SISMAC: Um Sistema para o Monitoramento de Colmeias em Tempo Real	381
24	Avaliação de Desempenho de Contêineres sob Dois Aspectos: Planejamento de Arquiteturas e Armazenamento de Dados	399
25	Computação veicular em nuvem: modelagem de disponibilidade e análise de sensibilidade	421
26	Envelhecimento da plataforma Docker: uma avaliação sistemática de desempenho e previsão do consumo de recursos	443
27	Avaliação de Desempenho de Arquiteturas VANETs com Semáforos Inteligentes Cooperativos	467
28	Alocação de Servidores em Computação na Borda Móvel Usando Redes de Petri Estocásticas	489
29	Avaliação de Desempenho e Disponibilidade de uma Arquitetura de Hospital Inteligente	513

30 Metodologia para Recomendação de Rotas de Evacuação em Cenário de Desastre Natural	539
III Processamento de sinais	567
31 Extração Manual De Características Para Classificação De Sons Urbanos	569
32 Processamento e Classificação de Cenários Acústicos para Sons Urbanos	589
IV Ferramentas Computacionais	605
33 GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos	607
V Mapeamento Sistemático para Sistemas Computacionais	631
34 Mapeamento Sistemático de Características Musicais para Reconhecimento Automático de Emoções	633
35 Mineração de dados espaço-temporais do Twitter no contexto de desastres naturais	661
36 Mapeamento Sistemático das Técnicas de Reconhecimento Facial para Sistemas de Controle de Acesso	681

Parte I

Processamento de dados médicos

Análises de Mapas de Texturas como Entradas pra Extração de Características em Redes Neurais Convolucionais

Autores: Daniel Veloso da Silva; Romuere Rodrigues Veloso e Silva.

Resumo: O glaucoma é uma das causas mais significativas de cegueira no mundo. Os médicos usam imagens digitais para detectar essas doenças. A detecção precoce da doença aumenta as chances de tratamento, reduzindo os efeitos adversos. Este trabalho propõe uma avaliação de combinações de mapas de textura como entrada para Redes Neurais Convolucionais para classificação de glaucoma em imagens de retinais. Em nossos experimentos, usamos os mapas de texturas GLCM, HOG e wavalet, três arquiteturas Redes Neurais Convolucionais, VGG16, VGG19 e ResNet50, e seis classificadores, *Gaussian Process*, *Decision Tree*, *Random Forest*, *Linear and Quadratic Decision*, *Nearest Neighbors* e *AdaBoost*. Utilizando o banco de dados RIM-ONE-V2 alcançamos um $Kappa = 0,716 \pm 0,116$ e uma Precisão = $0,876 \pm 0,057$, e no conjunto de dados Drishti alcançamos um $Kappa = 0,716 \pm 0,116$ e uma precisão de $0,876 \pm 0,057$. Concluímos que o uso da combinação de mapas de textura pode melhorar a detecção automática de glaucoma em comparação com entradas de canal único e pode ser usado por métodos de última geração para melhorar suas taxas de classificação.

Keywords: Mapas de Textura. CNN. Glaucoma. Descritor de Textura.

1.1 Introdução

Sistemas inteligentes vem sendo empregados para auxiliar profissionais em diversos cenários. Neste trabalho abordaremos a utilização da tecnologia na área da saúde, frisando a detecção precoce do glaucoma.

O glaucoma é uma doença progressiva, e o não tratamento da mesma pode resultar na perda da visão. Dados divulgados pela Organização Mundial de Saúde(OMS) apresentam esta doença como uma das principais causas de cegueira mundial. Projeções da OMS indicam que o glaucoma afetará 80 milhões de pessoas em 2020 e 111,5 milhões em 2040 ([TURBIANI, 2019](#)). O glaucoma pode se apresentar de 4 tipos diferentes, sendo eles glaucoma de ângulo fechado (agudo), glaucoma de ângulo aberto (crônico), glaucoma

congênito e glaucoma secundário. Embora o glaucoma não possua uma cura, todos os tipos apresentados possuem tratamento eficaz quando detectado precocemente (THYLEFORS; NEGREL, 1994).

O glaucoma é causado por uma lesão no Disco Óptico (DO), que é responsável por transmitir o que vemos ao cérebro (LENSCOPE, 2018). Usualmente o glaucoma é diagnosticado em uma imagem e para isso é necessário delimitar a região do DO e da Cavidade do DO (CDO), que é a região mais interna. O indicador mais utilizado para determinar a presença do glaucoma é o aumento do CDO. Para isso, calculada-se o *Cup-to-Disc Ratio* (CDR), que é a razão entre o CDO e o DO, quanto mais próximo de 1 maior a probabilidade da presença do glaucoma.

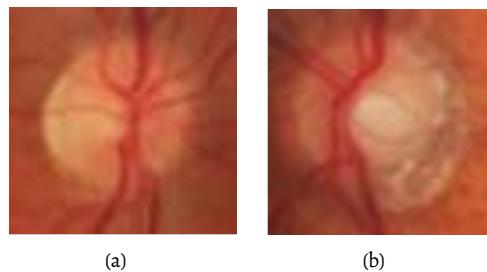


Figura 1 – Exemplo de imagens do disco óptico saudável (a) e glaucomatosa (b).

O diagnóstico de algumas doenças na área da oftalmologia, implica na análise de muitas imagens, mesmo que seja apenas para um único paciente, esse processo de análise das imagens no diagnóstico da doença, em grande parte ainda é feito de forma manual, através de um processo demorado e subjetivo, cujo resultado varia substancialmente entre os profissionais, resultando em uma elevada carga de trabalho para os especialistas. (SEVERO, 2014)

O processamento digital de imagens constitui-se em um conjunto de técnicas para manipular imagens com o auxílio do computador. Essas técnicas permitem extraer e identificar características das imagens e melhorar a qualidade visual de certos aspectos estruturais, facilitando a compreensão humana e a interpretação automática por meio dos computadores (PEDRINI; SCHWARTZ, 2008a).

Santos et al. (2009) afirmam que as imagens podem ser divididas em partes ou segmentos para serem analisadas de forma isolada e facilitar a extração das características através da análise de textura ou forma. A análise de textura possui uma vasta quantidade de propriedades. Através da análise de textura, é possível a extração de informações como: rugosidade, aspereza, maciez, ondulações, etc.

Apesar do CRD ser uma forma bastante utilizada e confiável para detectar automaticamente o glaucoma em imagens, ele requer que as delimitações do DO e do CDO sejam feitas a priori. Estas necessitam de um grau de precisão próximo ao real para que o cálculo do CRD seja confiável. Dessa forma, a fim de ajudar na identificação da doença em seus estágios iniciais, desenvolvemos uma proposta de identificação do glaucoma, utilizando descritores de textura combinados a Redes Neurais Convolucionais (*Convolutional Neural Network*) - CNN e classificadores, que não necessitam da etapa de segmentação de estruturas prévias.

Esta proposta desenvolvida não possui o objetivo de ultrapassar o estado da arte, visamos avaliar se outro tipo de entrada em redes neurais melhoram os resultados quando comparados com a imagem original. Os resultados obtidos com o método desenvolvido, visam ajudar os especialistas no diagnóstico, oferecendo uma segunda opinião.

1.1.1 Objetivos Gerais

O objetivo deste trabalho, propõe-se em avaliar outros tipos de entradas em redes neurais convolucionais através das transformações das imagens em mapas de texturas, para averiguar se os resultados produzidos superam as entradas tradicionais, ou seja, utilizando somente as imagens como entrada na rede.

1.1.2 Objetivos Específicos

- Avaliar outras formas de transformações para entrada em uma CNN;

- Testar as novas imagens nas arquiteturas de CNN's e em redes neurais artificiais; e,
- Avaliar e comparar os resultados dessas transformações .

1.2 Trabalhos Relacionados

Os trabalhos relacionados expostos nesta seção, apresentam as diversas técnicas utilizadas no campo de visão computacional com enfase na detecção de glaucoma.

O trabalho apresentado por ([LI et al., 2018](#)), utiliza o conjunto de dados *LabelMe-2016*, de onde foram retiradas 70.000 imagens das 200.000 existentes para uma amostragem aleatória, e selecionaram 48.116 imagens para a rotulagem de um algoritmo de aprendizado para a detecção da neuropatia óptica glaucomatosa. Os autores realizaram a normalização das imagens com os pixels no intervalo de 0 a 1, e a redução para uma matriz de 299x299. Com isso obtiveram área sob a curva ROC (AUC) de 0,986, sensibilidade (Sens) de 95,6% e especificidade (Espec) de 92%.

O trabalho ([RAGHAVENDRA et al., 2018](#)) utilizou uma CNN de 18 camadas, proposta pelos mesmos. Dividiram o banco de dados em treino e teste. O banco de dados utilizado é o *Kasturba Medical College, Manipal India*, com um total de 589 imagens normais e 837 com glaucoma, onde com as mesmas obtiveram uma acurácia (Acc) de 98,13%. Para o desenvolvimento deste trabalho, os autores criaram a conjunto de dados, os quais não estão disponíveis.

([BISNETO; OSEAS; MAGALHÃES, 2020](#)) apresentou um método para o diagnóstico precoce de glaucoma utilizando uma Rede Adversária Generativa aliada a atributos de textura definidos a partir de índices de diversidade taxonômica. Para validação dos resultados, os autores utilizaram duas bases de imagens públicas, RIM-ONE e *Drishti-GS*. Os outros autores obtiveram uma acurácia de 100% e uma curva ROC de 1.

O trabalho apresentado em ([ARAUJO, 2018](#)), aborda uma metodologia para diagnóstico do glaucoma a partir das imagens de fundo de olho, utilizando os índices de diversidade, como descritores de textura para a extração das características de textura. Os algoritmos genéticos são utilizados para seleção das características mais relevantes. Após a extração das características, os autores utilizaram a máquina de vetores de suporte (SVM) para realizar a classificação obtendo acurácia de 93,41%, sensibilidade de 92,36% e especificidade de 95,05%.

O trabalho de Júnior et al em ([CARVALHO-JÚNIOR et al., 2017](#)), desenvolveu um método para detecção automática do glaucoma com imagens da retina utilizando imagens do banco de dados RIM-ONE. Utilizaram a técnica de segmentação do DO aplicando o algoritmo de *Otsu*. Para a extração de características utilizaram descritores de textura baseados em índices de diversidade filogenética para classificar utilizando o framework WEKA. Os resultados apresentados foram acurácia de 98,6%, sensibilidade 92,9% e Especificidade de 100%.

No artigo ([SILVA et al., 2018](#)) é proposto uma metodologia para o diagnóstico do glaucoma através da análise da textura da imagem representada usando LBP (*Local Binary Pattern*) e estatística espacial aplicada em imagens médicas de retinografias. Os autores obtiveram acurácia de 95.08%. O método foi aplicado em imagens segmentadas baseada em especialista que continha na base de imagem RIM-ONE.

No trabalho ([AL-BANDER et al., 2017](#)) foi empregada uma metodologia para a detecção do glaucoma utilizando redes neurais e o classificador SVM. Em seu trabalho é utilizado o banco de dados RIM-ONE, onde foi realizado o redimensionamento das imagens para 227x227 pixels. Utilizou uma CNN pré-treinada (Alexnet) para a extração de características e por fim e utilizou o classificador SVM, obtendo assim acurácia de 88,2%, sensibilidade 85% e especificidade de 90,8%.

O trabalho ([AHN et al., 2018](#)) utiliza em seu desenvolvimento um banco de dados privado. A sua metodologia consiste na utilização da CNN *GoogleNet Inception v3* como extrator e classificador de características. Neste trabalho os autores redimensionaram as imagens na largura fixa de 800 pixels, e logo em seguida utilizaram uma CNN pré-treinada para a extração e classificação de características. O Resultado apresentado pelos autores foi uma AUC de 0,93.

A Tabela 1 apresenta um resumo dos trabalhos relacionados, mostrando a base de dados utilizada e os resultados obtidos.

Tabela 1 – Comparativo dos trabalhos do estado da arte.

Autor	Base de Imagens	Acc (%)	AUC	Sens (%)	Espec (%)
(LI et al., 2018)	LabelMe 2016	-	0,99	95,60	92,00
(RAGHAVENDRA et al., 2018)	Kasturba Medical College	98,13	-	-	-
(ARAUJO, 2018)	RIM-ONE V2	93,41	-	92,36	95,05
(CARVALHO-JÚNIOR et al., 2017)	RIM-ONE	98,60	-	92,90	100
(SILVA et al., 2018)	RIM-ONE	95,08	-	-	-
(AL-BANDER et al., 2017)	RIM-ONE	88,20	-	85,00	90,80
(BISNETO; OSEAS; MAGALHÃES, 2020)	RIM-ONE e Drishti	100,00	1	-	-
(AHN et al., 2018)	Privada	-	0,93	-	-

Utilizamos no presente trabalho, três algoritmos de textura e três tipos de CNN's diferentes. Realizamos os testes em dois conjuntos de dados públicos disponíveis para a replicação de testes. Classificamos as características com seis algoritmos diferentes. Considerando o atual trabalho com os demais podemos destacar uma maior exploração das técnicas de entrada de dados nas CNN's.

1.3 Referencial Teórico

Neste trabalho foram utilizados conhecimentos das áreas de visão computacional com aplicação de mapas de textura na extração de informações obtidas através de imagens retinográficas. Nas seções seguintes, serão descritas a base de dados, os métodos para extração dos mapas de textura e de atributos, bem como a técnica utilizada para a classificação das imagens.

1.3.1 Base de Imagens

Neste trabalho utilizamos duas bases de imagens, sendo elas a *RIM-ONE-V2* e *Drishti*. O banco de imagens *RIM-ONE-V2* consiste em imagens fornecidas por três hospitais. Sua composição consiste em 455 imagens, sendo 200 glaucomatosas e 255 saudáveis ([RIMONE, 2015](#)). Na Figura 2, apresentamos duas amostras de ambas as imagens contidas no *RIM-ONE-V2*.

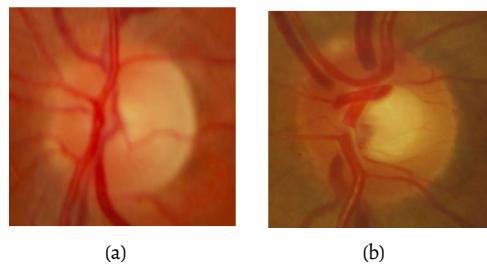


Figura 2 – Exemplos de imagens da base de dados RIM-ONE-V2. Em (a) uma imagem de uma retina saudável e em (b) uma glaucomatosa.

O conjunto de dados *Drishti* consiste em um total de 101 imagens, sendo elas divididas duas classes, glaucomatosas com 31 imagens e normais com 70 imagens. Este conjunto de dados é de uma única população, sendo que todas as imagens foram coletadas no hospital de olhos *Aravind Eye Hospital*, Madurai, Índia

([SIVASWAMY et al., 2015](#)). Na Figura 3, apresentamos duas amostras de ambas as imagens contidas no *Drishti*.

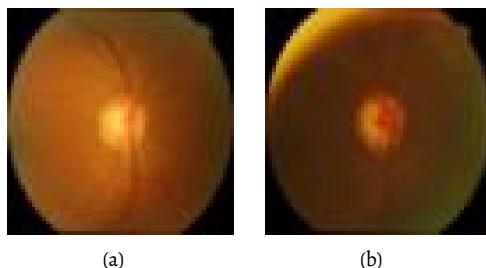


Figura 3 – Exemplos de imagens da base de dados *Drishti*. Em (a) uma imagem de uma retina saudável e em (b) uma glaucomatosa.

1.3.2 Mapas de Textura

A análise de mapas de textura é um método quantitativo de pré-processamento para extração de dados ([STANZIONE et al., 2019](#)). Nesta seção serão apresentados algoritmos clássicos para obtenção de mapas de textura e que foram utilizados neste trabalho.

1.3.2.1 Wavelet - WAV

A transformada Wavelet é comumente usada em muitos problemas de visão computacional. No WAV é realizada uma decomposição de wavelet 2-D de nível único, que requer uma função de dimensionamento 2-D, e três wavelets 2-D, onde o índice $i = H, V, D$ é a representação das variações de intensidade ao longo de colunas ou arestas horizontais e ao longo das diagonais ([MALLAT, 1989](#)). A Figura 4 mostra a aplicação do descritor em uma das imagens do banco de dados RIM-ONE-V2.

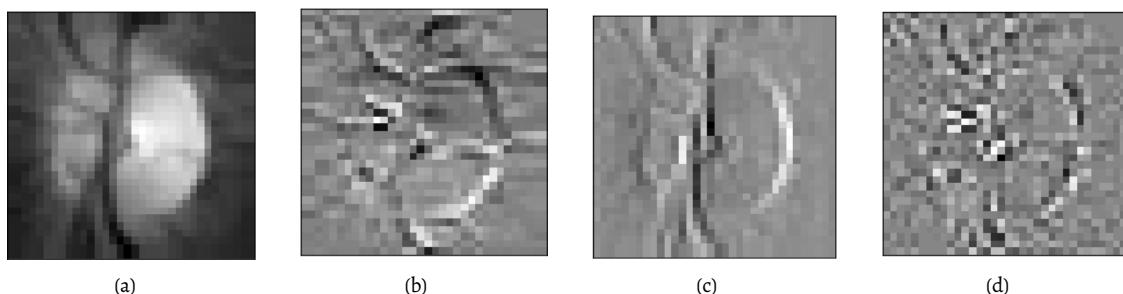


Figura 4 – Imagem com células glaucomatosa da Base RIM-ONE com a aplicação do WAV. (a) Aproximação, (b) Detalhe horizontal, (c) Detalhe vertical, (d) Detalhe diagonal.

1.3.2.2 GLCM

A Matriz de Co-ocorrência de Nível de Cinza (GLCM) é um histograma de valores de escala de cinza, que registra co-ocorrências em um determinado deslocamento sobre uma imagem. O presente algoritmo mostra com que frequência cada nível de cinza ocorre em um pixel localizado em uma posição geométrica fixa relativa a cada outro pixel ([HARALICK et al., 1979](#)). Essa frequência gera uma matriz de co-ocorrência, que será utilizada como mapas de textura. A média da GLCM nas 4 direções ($0^\circ, 45^\circ, 90^\circ, 135^\circ$), usando a distância = 1. Observe o funcionamento do descritor na Figura 5.

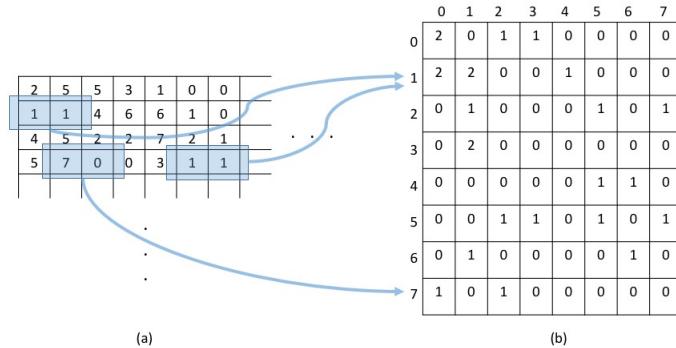


Figura 5 – Funcionamento do descriptor GLCM. (a) imagem, (b) matriz GLCM. Fonte: ([SILVA et al., 2017](#))

1.3.2.3 HOG

O Histograma de Orientação dos Gradientes (HOG) combina a descrição de características de forma e textura. Ele preconiza que a aparência e forma de um objeto local pode ser distinguida pelas direções das bordas ou pela distribuição do gradiente da intensidade. As etapas do HOG consistem em uma normalização de imagem (opcional), computação dos gradientes para obtenção da magnitude e direção. Em seguida são computados os histogramas de gradiente a partir de células ([DALAL; TRIGGS, 2005](#)).

Ao final é realizada uma normalização de blocos para obtenção do vetor de atributos. Os parâmetros de entrada do HOG são quantidade de células e blocos. Neste trabalho utilizamos 3 células e 3 blocos. A Figura 6 é uma imagem com a aplicação do descriptor.

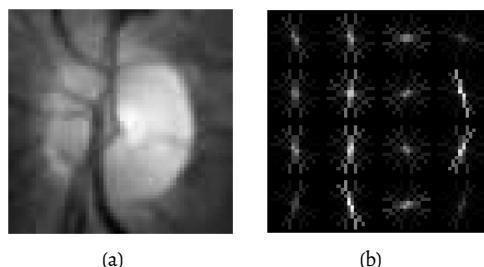


Figura 6 – Imagem com células glaucomatosa da Base RIM-ONE com a aplicação do HOG. (a) Imagem em tons de cinza, (b) Mapa de textura HOG.

1.3.3 Redes Neurais Convolucionais

Uma CNN é um tipo especial de redes neurais multicamada, projetadas para reconhecer padrões visuais diretamente de imagens de pixel com pré-processamento mínimo. Neste trabalho utilizamos três arquiteturas de CNNs, as mesmas foram escolhidas pela frequência com que são utilizadas em trabalhos relacionados. A seguir mostramos algumas informações sobre cada uma.

As redes VGG16 e VGG19 ([SIMONYAN; ZISSERMAN, 2014a](#)) são caracterizadas pela sua simplicidade, usando apenas camadas convolucionais 3×3 empilhadas umas sobre as outras em profundidade crescente. A redução do tamanho do volume é tratada pelo *Maxpool*. Duas camadas totalmente conectadas, cada uma com 4.096 nós. As redes VGG16 e VGG19 Os “16” e “19” representam o número de camadas de peso na rede. Todas as características extraídas para a classificação foram retiradas da penúltima camada da CNN. As Figuras 7 e 8 mostram a arquitetura das mesmas.

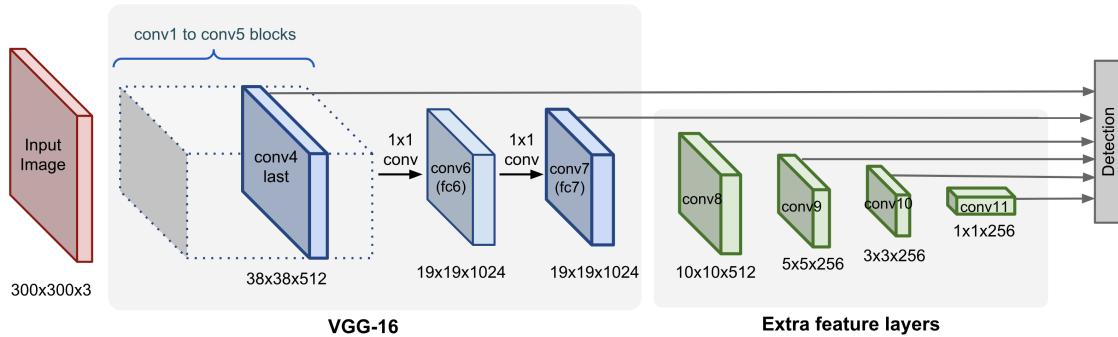


Figura 7 – Arquitetura da CNN VGG16

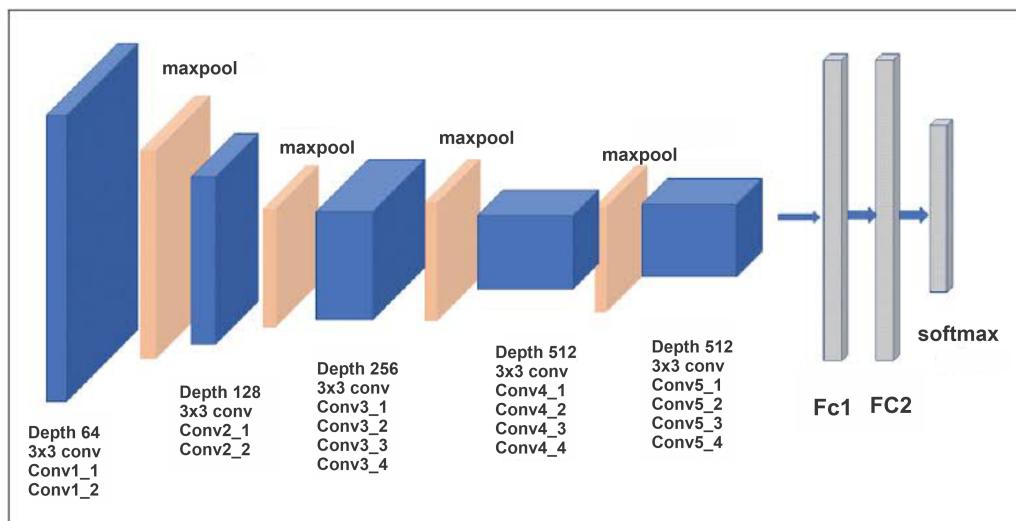
Fonte: Bibek Chaudhary ([CHAUDHARY, 2019](#)).

Fig. 3. VGG-19 network architecture

Figura 8 – Arquitetura da CNN VGG19

Fonte: Sai Charan ([ARISHANAPALLY, 2019](#)).

A ResNet50 é uma CNN baseada em redes simples, que insere "atalhos" fazendo com que a rede se torne residual, as saídas de uma camada são entradas de camadas mais a frente. Os atalhos podem ser aplicadas quando as entradas e as saídas tem as mesmas dimensões, mesmo não tendo as mesmas dimensões tem tratamentos ([HUANG; LIU; WEINBERGER, 2016](#)). A Figura 9 apresenta sua arquitetura.

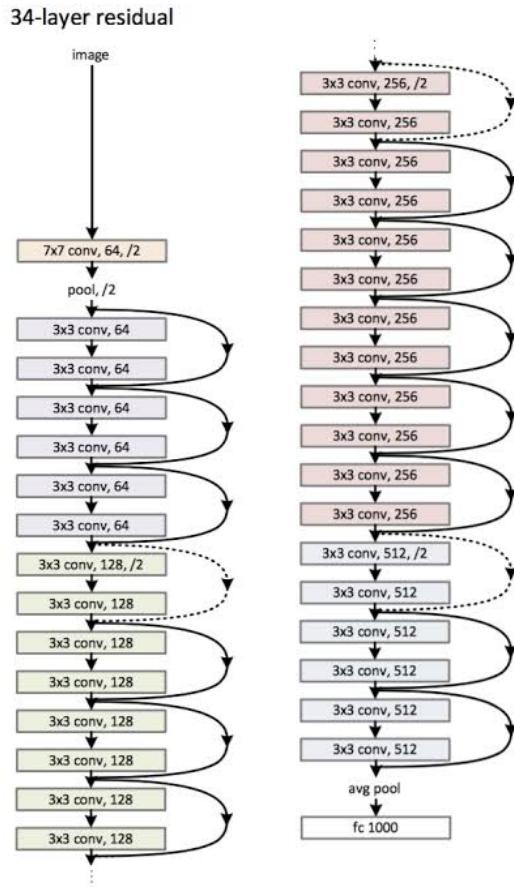


Figura 9 – Arquitetura da CNN ResNet50

Fonte: Andrew Dabydeen ([DABYDEEN, 2019](#)).

1.3.4 Classificadores

1.3.4.1 Gaussian Processes - GP

O classificador GP é um método genérico de aprendizado de máquina para resolver problemas de regressão e classificação probabilística. A primeira execução para classificação sempre é realizada a partir de valores iniciais de hiper-parâmetros do *kernel*, as execuções seguintes são realizadas a partir de valores de hiper-parâmetro que foram escolhidos aleatoriamente no intervalo de valores permitidos (RASMUSSEN, 2003).

1.3.4.2 Decision Tree - DT

DT é um método de aprendizado supervisionado não paramétrico utilizado para classificação e regressão. O DT prediz o valor de uma variável destino, aprendendo regras simples de decisão inferidas a partir dos recursos de dados (BREIMAN et al., 1984).

1.3.4.3 Random Forest - RF

O RF é um meta-estimador que se encaixa em vários classificadores de árvores de decisão. Quando aplicado em diversas subamostras de um conjunto de dados, o mesmo usa a média para melhorar a precisão preditiva e controlar o ajuste excessivo (??).

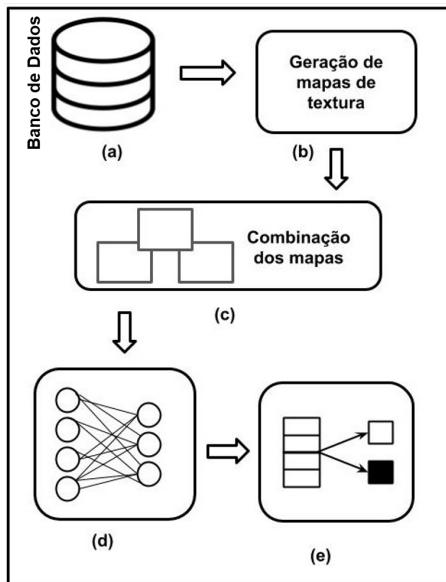


Figura 10 – Metodologia proposta. (a) Banco de Dados (b) Aplicação dos mapas de textura. (c) Combinação dos mapas de textura. (d) Extração das características utilizando CNN's. (e) Classificação.

1.3.4.4 AdaBoost

Assim como o RF, o *AdaBoost* é um meta-estimador. Na sua execução o mesmo começa ajustando um classificador no conjunto de dados original para depois ajusta cópias adicionais do classificador no mesmo conjunto de dados. Onde os pesos das instâncias foram classificadas incorretamente, os mesmos são ajustados para que os classificadores subsequentes se concentrem mais em casos difíceis (FREUND; SCHAPIRE, 1995).

1.3.4.5 Nearest Neighbors - NN

NN é um método de classificação baseado em vizinhos, o mesmo pode ser utilizado de forma supervisionada ou não supervisionada. A ideia de utilizar os vizinhos mais próximos é de encontrar um número predefinido de amostras de treinamento, que pode ser definida pelo usuário ou com base na densidade local de pontos. A distância pode, em geral, ser qualquer medida métrica: a distância euclidiana é a escolha mais comum (PEDREGOSA et al., 2011).

1.3.4.6 Linear and Quadratic Decision (QDA)

QDA é um classificado de decisão linear e quadrática. O mesmo pode ser usado para executar a redução supervisionada da dimensionalidade, projetando os dados de entrada em um subespaço linear que consiste nas direções que maximizam a separação entre as classes (PEDREGOSA et al., 2011).

1.4 Metodologia

O método proposto consiste em 5 etapas, apresentadas na Figura 10: a primeira etapa é a aquisição de imagens, onde utilizamos as bases de imagens RIM-ONE-V2 e *Drishti*; após isso, temos a etapa de geração dos mapas de textura onde utilizamos os descritores WAV, GLCM e HOG; o passo seguinte foi utilizar esses mapas de texturas como entrada para as CNNs, foram utilizadas as redes VGG16, VGG19 e ResNet50; a última etapa foi a classificação das imagens onde utilizamos os atributos extraídos pelas CNNs e classificamos nosso conjuntos de dados em saudáveis e glaucomatosas.

1.4.1 Cálculo dos Mapas

As técnicas explicadas na Seção 1.3 foram aplicadas nos bancos de dados públicos RIM-ONE V2 e *Drishti*. A primeira técnica aplicada nas imagens, foram os descritores de textura. Para cada descritor utilizamos todo o conjunto de imagens, gerando assim 3 novos conjuntos. Com os mapas de texturas criados, utilizamos a ideia da combinação de 2 mapas de textura com uma imagem em níveis de cinza para a criação de uma nova imagem. A ideia consiste na utilização da estrutura de uma imagem RGB, porém, ao invés de utilizar a imagem colorida, optamos por colocar mapas de texturas em cada canal. De forma abstrata a Figura 11 representa a metodologia utilizada para geração dos mapas de textura.



Figura 11 – Reconstrução da Imagem. (a) Ilustração dos canais RGB (b) Reconstrução da imagem com os mapas de texturas.

Com os três descritores de texturas conseguimos obter três combinações: 1) WAV + imagem em níveis de cinza + HOG (WCH), 2) GLCM + imagem em níveis de cinza + HOG (GCH), e 3) WAV + imagem em níveis de cinza + GLCM (WCG). As imagens em níveis de cinza utilizadas para montar as combinações foram obtida pela Equação 1.1, que é o padrão mais utilizado em aplicações de processamento digital de imagens ([RECOMMENDATION, 2015](#)).

$$\text{Cinza} = 0,2126 * R + 0,7154 * G + 0,0721 * B, \quad (1.1)$$

onde R, G e B são os canais vermelho, verde e azul do modelo de cores RGB.

1.4.2 Extração de Características

Utilizamos as três CNN's para extrair características e por fim classificar as imagens com os algoritmos de classificação. Utilizamos a metodologia de transferência de aprendizado para extrair atributos com as CNN's. Para isso, utilizamos os pesos da base de imagens ImageNet ([DENG et al., 2009a](#)). Com isso, não foi necessário treinar as redes com os dados de imagens com e sem glaucoma. Da mesma forma que os descritores, utilizamos as CNN's uma por vez, gerando, assim, as características a serem classificadas.

1.4.3 Classificação

Para o processo de classificação, fora utilizados seis classificadores que estão disponíveis na literatura, que são eles: *Gaussian Processes* (GP) ([RASMUSSEN, 2003](#)), *Decision Tree* (DT) ([BREIMAN et al., 1984](#)), *Random Forest* (RF) ([??](#)), *AdaBoost* ([FREUND; SCHAPIRE, 1995](#)), *Linear and Quadratic Decision* (QDA) e *Nearest Neighbors* ([PEDREGOSA et al., 2011](#)). Nas subseções seguintes abordaremos as principais características de cada classificador.

1.4.3.1 Parametrização

No classificador *Gaussian Processes* utilizamos o otimizador L-BFGS-B e 100 como quantidade máxima de iteração para a predição. Para a *Decision Tree* utilizamos o índice Gini como função para medir a qualidade das divisões dos nós e a quantidade mínima de atributos para criar um novo nó foi 2. No classificador *Random Forest*, utilizamos 100 árvores de decisão e o índice Gini como função para medir a qualidade das divisões dos nós.

1.4.3.2 Treinamento

Na etapa de treinamento dos classificadores utilizamos 80% dos dados para treinamento e 20% para teste. Além disso, executamos cada classificação 10 vezes sempre modificando os conjuntos de treinamento e teste. Com isso, obtivemos para cada métrica utilizada uma média e o desvio padrão das 10 execuções.

1.4.3.3 Métricas e Validação

Utilizamos como métricas de avaliação dos métodos a acurácia (Acc), o índice *Kappa*, a área sob a curva ROC (AUC) e o F1-Score (F1). A Acc é a aproximação de um resultado com o seu valor de referência real. O *Kappa* é utilizado para descrever as concordâncias entre 2 ou mais situações (VIERA; GARRETT et al., 2005). A AUC é uma representação gráfica que ilustra a performance de um classificador binário à medida que a variamos o limiar de discriminação. O F1 é uma média ponderada entre a precisão e o *recall*.

1.5 Resultados e Discussão

Os resultados apresentados nesse trabalho foram obtidos utilizando três algoritmos para obtenção dos mapas de textura, três arquiteturas de CNN's e seis classificadores. Além disso, de modo a comparar a performance obtida com a combinação dos mapas de textura, executamos o processo utilizando como entrada na CNN os canais de uma imagem RGB separados (vermelho, verde e azul). Nas Tabelas 2, 3, 4, 5, 6 e 7, apresentamos os resultados para os classificadores Gaussian Process, Decision Tree, Random Forest, Linear and Quadratic Decision, Nearest Neighbors e AdaBoost respectivamente.

Tabela 2 – Resultados obtidos pela metodologia proposta utilizando o classificador Gaussian Process.

RIM-ONE-V2						Drishti		
	VGG16	VGG19	RESNET-50	VGG16	VGG19	RESNET-50		
Vermelho						Vermelho		
Acc	0,813±0,026	0,829±0,029	0,828±0,036	0,623±0,105	0,576±0,096	0,685±0,083		
Kappa	0,617±0,049	0,648±0,061	0,648±0,071	0,120±0,204	0,035±0,151	0,254±0,177		
AUC	0,808±0,023	0,822±0,033	0,828±0,032	0,561±0,115	0,521±0,077	0,622±0,085		
F1	0,776±0,029	0,791±0,038	0,796±0,040	0,383±0,149	0,340±0,098	0,479±0,150		
Verde						Verde		
Acc	0,812±0,034	0,7813±0,030	0,782±0,028	0,576±0,098	0,714±0,063	0,638±0,093		
Kappa	0,606±0,078	0,553±0,057	0,554±0,061	0,001±0,222	0,254±0,165	0,017±0,186		
AUC	0,801±0,038	0,775±0,029	0,777±0,029	0,505±0,114	0,625±0,086	0,501±0,082		
F1	0,762±0,055	0,743±0,040	0,746±0,044	0,289±0,180	0,433±0,158	0,259±0,170		
Azul						Azul		
Acc	0,748±0,087	0,757±0,041	0,808±0,023	0,657±0,066	0,647±0,102	0,661±0,058		
Kappa	0,482±0,181	0,498±0,079	0,609±0,042	0,203±0,141	0,183±0,203	0,196±0,130		
AUC	0,741±0,091	0,747±0,039	0,805±0,020	0,600±0,070	0,585±0,098	0,603±0,075		
F1	0,647±0,230	0,700±0,0478	0,778±0,018	0,440±0,135	0,429±0,128	0,415±0,103		
WCH						WCH		
Acc	0,801±0,035	0,725±0,081	0,859±0,030	0,771±0,092	0,709±0,083	0,557±0,079		
Kappa	0,610±0,062	0,391±0,256	0,698±0,071	0,465±0,158	0,297±0,138	0,046±0,133		
AUC	0,805±0,034	0,697±0,125	0,842±0,035	0,715±0,073	0,654±0,091	0,476±0,064		
F1	0,789±0,033	0,528±0,343	0,826±0,031	0,611±0,110	0,492±0,104	0,265±0,087		
GCH						GCH		
Acc	0,741±0,133	0,812±0,022	0,849±0,020	0,804±0,091	0,828±0,043	0,685±0,060		
Kappa	0,510±0,202	0,638±0,051	0,683±0,051	0,458±0,299	0,572±0,114	0,023±0,090		
AUC	0,759±0,103	0,817±0,026	0,846±0,021	0,704±0,157	0,769±0,066	0,490±0,044		
F1	0,738±0,242	0,83±0,033	0,869±0,020	0,574±0,262	0,680±0,104	0,055±0,112		
WCG						WCG		
Acc	0,803±0,032	0,748±0,095	0,859±0,021	0,785±0,146	0,790±0,090	0,723±0,041		
Kappa	0,596±0,075	0,415±0,274	0,708±0,054	0,490±0,338	0,465±0,238	0,133±0,142		
AUC	0,791±0,032	0,702±0,131	0,856±0,021	0,744±0,168	0,721±0,119	0,564±0,069		
F1	0,762±0,048	0,536±0,351	0,829±0,032	0,572±0,334	0,598±0,185	0,218±0,200		

A Tabela 2 mostra que o melhor resultado foi obtido utilizando a combinação WCG com um Kappa = 0,708±0,054. Por outro lado, a menor performance foi obtida pelo canal de cor verde do modelo RGB com Kappa = 0,554±0,061, isso quando o método foi aplicado na base de dados RIM-ONE-V2. No banco de dados

Drishti, o melhor resultado apresentado foi com a utilização do método nas combinações GCH, onde os mesmos fora extraídos pela VGG19, obtendo um Kappa = $0,572 \pm 0,114$.

Tabela 3 – Resultados obtidos pela metodologia proposta utilizando o classificador *Decision Tree*;

	RIM-ONE-V2			<i>Drishti</i>		
	VGG16	VGG19	RESNET-50	VGG16	VGG19	RESNET-50
	Vermelho			Vermelho		
Acc	$0,720 \pm 0,046$	$0,732 \pm 0,044$	$0,739 \pm 0,050$	$0,595 \pm 0,111$	$0,600 \pm 0,117$	$0,600 \pm 0,093$
Kappa	$0,42 \pm 0,090$	$0,458 \pm 0,090$	$0,465 \pm 0,096$	$0,055 \pm 0,244$	$0,068 \pm 0,225$	$0,104 \pm 0,165$
AUC	$0,716 \pm 0,048$	$0,728 \pm 0,045$	$0,731 \pm 0,045$	$0,530 \pm 0,118$	$0,524 \pm 0,125$	$0,563 \pm 0,103$
F1	$0,665 \pm 0,048$	$0,700 \pm 0,060$	$0,688 \pm 0,053$	$0,317 \pm 0,204$	$0,350 \pm 0,172$	$0,385 \pm 0,095$
	Verde			Verde		
Acc	$0,689 \pm 0,027$	$0,689 \pm 0,039$	$0,695 \pm 0,036$	$0,676 \pm 0,094$	$0,738 \pm 0,090$	$0,623 \pm 0,109$
Kappa	$0,364 \pm 0,056$	$0,366 \pm 0,082$	$0,382 \pm 0,070$	$0,228 \pm 0,163$	$0,337 \pm 0,260$	$0,097 \pm 0,273$
AUC	$0,683 \pm 0,025$	$0,683 \pm 0,041$	$0,694 \pm 0,037$	$0,609 \pm 0,086$	$0,662 \pm 0,129$	$0,568 \pm 0,164$
F1	$0,627 \pm 0,048$	$0,638 \pm 0,056$	$0,650 \pm 0,045$	$0,449 \pm 0,109$	$0,491 \pm 0,223$	$0,342 \pm 0,219$
	Azul			Azul		
Acc	$0,709 \pm 0,049$	$0,654 \pm 0,033$	$0,736 \pm 0,039$	$0,667 \pm 0,099$	$0,647 \pm 0,098$	$0,638 \pm 0,139$
Kappa	$0,409 \pm 0,095$	$0,297 \pm 0,061$	$0,464 \pm 0,076$	$0,223 \pm 0,257$	$0,160 \pm 0,210$	$0,167 \pm 0,264$
AUC	$0,705 \pm 0,045$	$0,650 \pm 0,030$	$0,732 \pm 0,037$	$0,622 \pm 0,131$	$0,578 \pm 0,111$	$0,582 \pm 0,128$
F1	$0,665 \pm 0,054$	$0,592 \pm 0,027$	$0,704 \pm 0,033$	$0,451 \pm 0,216$	$0,402 \pm 0,160$	$0,408 \pm 0,179$
	WCH			WCH		
Acc	$0,701 \pm 0,042$	$0,719 \pm 0,047$	$0,749 \pm 0,022$	$0,685 \pm 0,080$	$0,790 \pm 0,071$	$0,633 \pm 0,095$
Kappa	$0,405 \pm 0,075$	$0,423 \pm 0,100$	$0,485 \pm 0,053$	$0,294 \pm 0,191$	$0,416 \pm 0,261$	$0,197 \pm 0,190$
AUC	$0,698 \pm 0,032$	$0,703 \pm 0,055$	$0,736 \pm 0,048$	$0,649 \pm 0,098$	$0,717 \pm 0,142$	$0,606 \pm 0,096$
F1	$0,641 \pm 0,049$	$0,665 \pm 0,078$	$0,699 \pm 0,051$	$0,534 \pm 0,151$	$0,564 \pm 0,251$	$0,457 \pm 0,146$
	GCH			GCH		
Acc	$0,673 \pm 0,045$	$0,698 \pm 0,042$	$0,689 \pm 0,033$	$0,723 \pm 0,069$	$0,709 \pm 0,138$	$0,747 \pm 0,056$
Kappa	$0,342 \pm 0,095$	$0,359 \pm 0,091$	$0,356 \pm 0,061$	$0,363 \pm 0,157$	$0,355 \pm 0,314$	$0,357 \pm 0,149$
AUC	$0,676 \pm 0,044$	$0,689 \pm 0,047$	$0,670 \pm 0,020$	$0,695 \pm 0,081$	$0,696 \pm 0,168$	$0,713 \pm 0,097$
F1	$0,692 \pm 0,052$	$0,732 \pm 0,042$	$0,725 \pm 0,039$	$0,558 \pm 0,134$	$0,578 \pm 0,221$	$0,511 \pm 0,140$
	WCG			WCG		
Acc	$0,689 \pm 0,063$	$0,682 \pm 0,054$	$0,729 \pm 0,030$	$0,752 \pm 0,103$	$0,799 \pm 0,046$	$0,723 \pm 0,089$
Kappa	$0,346 \pm 0,124$	$0,345 \pm 0,122$	$0,412 \pm 0,088$	$0,383 \pm 0,191$	$0,527 \pm 0,120$	$0,309 \pm 0,204$
AUC	$0,671 \pm 0,065$	$0,672 \pm 0,069$	$0,705 \pm 0,040$	$0,692 \pm 0,088$	$0,783 \pm 0,076$	$0,673 \pm 0,115$
F1	$0,600 \pm 0,093$	$0,613 \pm 0,089$	$0,642 \pm 0,63$	$0,551 \pm 0,146$	$0,676 \pm 0,106$	$0,498 \pm 0,153$

Na Tabela 3 são mostrados os resultados da classificação utilizando o classificador DT, onde podemos observar que o método proposto foi melhor classificado, com relação a utilização das imagem em apenas canais nos dois bancos de imagens. Já na Tabela 4 as imagens do banco de imagens RIM-ONE-V2 separadas por canais obtiveram uma melhor classificação, porém com diferença mínima da classificação do método proposto. Já na base de imagens da *Drishti*, o método proposto com a combinação WCG extraída pela VGG19, onde obteve melhor resultado atingindo uma acurácia de 82%.

O classificador QDA apresentado na Tabela 5 teve resultados semelhantes, porém, a classificação das características extraídas pelo método proposto usando a VGG16 com os mapas de textura WCH, mostrou-se promissor, quando comparado com as imagens tradicionais em canais, utilizando a base de imagens RIM-ONE-V2. Enquanto a classificação das imagens no banco de dados RIM-ONE-V2 foi bem semelhante com todas as imagens em canais separados, o banco de imagens da *Drishti* obteve uma classificação bem superior utilizando o método proposto, quando comparado as imagens somente em canais.

O classificador *Nearest Neighbors* da Tabela 6 apresentou os melhores resultados nos canais Verde e Azul extraído pela Resnet50, já no método proposto os melhores resultados foram na transformação WCH também extraídos pela Resnet50. Podemos observar a diferença mínima dos resultados obtidos através dos canais, com o método proposto,utilizando o banco de imagens RIM-ONE-V2. A classificação utilizando o banco de imagens *Drishti*, obteve melhor calssificação quando aplicado ao método proposto nas imagens.

Por fim, os resultados obtidos na Tabela 7 com o classificador AdaBoost, obtiveram os melhores resultados, utilizando o método proposto nos dois bancos de imagens. Os resultados obtidos com os classificadores *Decision Tree* e *Random Forest*, *Linear and Quadratic Decision*, *Nearest Neighbors*, (Tabelas 3, 4, 5 e 6) tiveram re-

Tabela 4 – Resultados obtidos pela metodologia proposta utilizando o classificador *Random Forest*.

	RIM-ONE-V2			Drishti		
	VGG16	VGG19	RESNET-50	VGG16	VGG19	RESNET-50
	Vermelho			Vermelho		
Acc	0,758±0,055	0,760±0,060	0,780±0,031	0,714±0,099	0,676±0,084	0,757±0,080
Kappa	0,497±0,113	0,509±0,109	0,538±0,060	0,123±0,195	0,017±0,034	0,127±0,204
AUC	0,746±0,056	0,751±0,051	0,762±0,028	0,554±0,087	0,493±0,013	0,552±0,086
F1	0,695±0,067	0,697±0,065	0,708±0,040	0,184±0,218	0,0±0,0	0,175±0,225
	Verde			Verde		
Acc	0,720±0,054	0,745±0,051	0,734±0,051	0,761±0,090	0,714±0,106	0,723±0,079
Kappa	0,417±0,106	0,474±0,099	0,448±0,091	0,195±0,204	0,219±0,178	0,092±0,144
AUC	0,704±0,054	0,735±0,047	0,721±0,042	0,579±0,086	0,595±0,090	0,535±0,055
F1	0,626±0,078	0,690±0,055	0,639±0,062	0,257±0,223	0,287±0,199	0,113±0,176
	Azul			Azul		
Acc	0,702±0,042	0,712±0,043	0,775±0,041	0,709±0,100	0,704±0,101	0,709±0,086
Kappa	0,376±0,078	0,402±0,081	0,527±0,090	0,245±0,173	0,149±0,190	0,008±0,024
AUC	0,682±0,037	0,699±0,038	0,757±0,048	0,599±0,068	0,562±0,080	0,497±0,008
F1	0,584±0,068	0,632±0,050	0,694±0,071	0,310±0,199	0,198±0,226	0,0±0,0
	WCH			WCH		
Acc	0,709±0,031	0,687±0,062	0,742±0,053	0,747±0,087	0,695±0,118	0,671±0,078
Kappa	0,381±0,081	0,325±0,136	0,463±0,100	0,161±0,148	0,145±0,144	0,0±0,0
AUC	0,685±0,042	0,653±0,063	0,726±0,043	0,561±0,055	0,557±0,054	0,5±0,0
F1	0,614±0,053	0,542±0,100	0,639±0,077	0,201±0,177	0,185±0,169	0,0±0,0
	GCH			GCH		
Acc	0,719±0,045	0,668±0,042	0,712±0,058	0,776±0,079	0,785±0,071	0,728±0,095
Kappa	0,400±0,096	0,279±0,099	0,41±0,100	0,351±0,193	0,332±0,139	0,001±0,116
AUC	0,698±0,042	0,632±0,048	0,698±0,046	0,681±0,128	0,633±0,058	0,499±0,045
F1	0,769±0,052	0,749±0,049	0,782±0,049	0,456±0,188	0,424±0,125	0,053±0,107
	WCG			WCG		
Acc	0,698±0,035	0,658±0,055	0,702±0,032	0,809±0,073	0,828±0,057	0,742±0,083
Kappa	0,352±0,086	0,289±0,100	0,394±0,085	0,474±0,207	0,496±0,174	0,171±0,152
AUC	0,678±0,044	0,635±0,045	0,698±0,042	0,714±0,106	0,729±0,104	0,569±0,063
F1	0,602±0,058	0,519±0,089	0,589±0,076	0,564±0,197	0,582±0,171	0,229±0,200

sultados inferiores aos obtidos pelo *Gaussian Process* e *AdaBoost*. Isso mostra que estes classificadores, mesmo não tendo sido utilizados em nenhum dos trabalhos relacionados levantados, pode ser um dos algoritmos mais adequado para o problema em questão. Outro ponto que deve ser destacado é o descriptor WAV, que está presente nos melhores resultados obtidos, este apresentou-se como um dos melhores descritores para melhorar as características, obtendo assim uma melhor classificação.

Comparando os resultados dos 6 classificadores, podemos observar, que quando se muda o vetor de atributos, os resultados variam, este fato ocorre devido cada classificador possuir parâmetros específicos, estes podem melhorar ou piorar o resultado, dependendo da escolha dos mesmos.

1.6 Conclusão e trabalhos futuros

Este trabalho apresentou uma metodologia para utilização de mapas de textura para extrair atributos de redes profundas no diagnóstico automático do glaucoma utilizando as bases de imagens RIM-ONE-V2 e Drishti. Foram utilizados os algoritmos de textura WAV, GLCM e HOG para geração de mapas de textura de onde foram extraídas características com as CNN's VGG16, VGG19 e RESNET. Ao final as características foram classificadas pelos algoritmos *Gaussian Process*, *Decision Tree*, *Random Forest*, *Linear and Quadratic Decision*, *Nearest Neighbors* e *AdaBoost*.

Os Resultados obtidos foram promissores, em especial para a combinação dos mapas WAV - Imagem em níveis de cinza - GLCM para o conjunto de imagens do RIM-ONE-V2. Para o conjunto de dados da Drishti obteve melhor classificação com a combinação de mapas WAV - Imagens em níveis de Cinza - HOG. Além disso, observamos que a combinação da imagens em níveis de cinza com mapas de textura pode melhorar os resultados quando comparado com o uso somente da imagem. Os resultados indicam que a metodologia,

Tabela 5 – Resultados obtidos pela metodologia proposta utilizando o classificador QDA.

	RIM-ONE-V2			Drishti		
	VGG16	VGG19	RESNET-50	VGG16	VGG19	RESNET-50
	Vermelho				Vermelho	
Acc	0,584±0,062	0,568±0,042	0,560±0,055	0,561±0,097	0,571±0,085	0,533±0,055
Kappa	0,147±0,129	0,120±0,087	0,102±0,107	0,018±0,168	0,063±0,176	0,034±0,147
AUC	0,573±0,064	0,561±0,044	0,552±0,055	0,488±0,077	0,464±0,089	0,479±0,084
F1	0,495±0,079	0,491±0,062	0,461±0,067	0,259±0,095	0,220±0,157	0,274±0,126
	Verde				Verde	
Acc	0,570±0,038	0,582±0,049	0,565±0,031	0,547±0,083	0,576±0,088	0,638±0,093
Kappa	0,123±0,072	0,145±0,105	0,109±0,073	0,065±0,209	0,051±0,149	0,161±0,170
AUC	0,561±0,036	0,573±0,053	0,554±0,036	0,464±0,110	0,500±0,139	0,614±0,114
F1	0,488±0,048	0,496±0,072	0,474±0,066	0,213±0,153	0,211±0,127	0,375±0,140
	Azul				Azul	
Acc	0,579±0,065	0,551±0,054	0,530±0,049	0,595±0,107	0,571±0,082	0,566±0,083
Kappa	0,137±0,125	0,091±0,111	0,045±0,103	0,044±0,220	0,036±0,190	0,014±0,193
AUC	0,569±0,062	0,546±0,055	0,523±0,050	0,517±0,119	0,484±0,104	0,511±0,093
F1	0,497±0,067	0,476±0,075	0,439±0,069	0,316±0,157	0,245±0,188	0,285±0,198
	WCH				WCH	
Acc	0,589±0,042	0,568±0,067	0,572±0,032	0,642±0,100	0,519±0,080	0,557±0,102
Kappa	0,123±0,102	0,128±0,119	0,138±0,062	0,138±0,211	0,124±0,118	0,002±0,189
AUC	0,567±0,049	0,562±0,050	0,561±0,039	0,566±0,097	0,435±0,062	0,500±0,097
F1	0,449±0,063	0,440±0,109	0,532±0,043	0,368±0,170	0,201±0,113	0,308±0,141
	GCH				GCH	
Acc	0,576±0,031	0,559±0,063	0,568±0,032	0,709±0,091	0,623±0,054	0,571±0,097
Kappa	0,103±0,072	0,062±0,130	0,128±0,072	0,170±0,308	0,067±0,104	0,046±0,234
AUC	0,557±0,039	0,535±0,063	0,561±0,039	0,578±0,153	0,532±0,048	0,479±0,123
F1	0,662±0,023	0,632±0,064	0,582±0,053	0,352±0,262	0,210±0,128	0,228±0,216
	WCG				WCG	
Acc	0,563±0,053	0,560±0,032	0,542±0,072	0,690±0,095	0,585±0,090	0,633±0,085
Kappa	0,082±0,081	0,072±0,093	0,089±0,142	0,171±0,211	0,064±0,234	0,082±0,157
AUC	0,532±0,036	0,532±0,042	0,542±0,072	0,584±0,111	0,468±0,116	0,541±0,077
F1	0,386±0,069	0,386±0,109	0,502±0,079	0,363±0,194	0,167±0,191	0,329±0,128

embora necessite de melhorias, pode ser utilizada para auxiliar na detecção precoce do glaucoma. Outro ponto importante que pode ser destacado é que este método pode ser extrapolado para outros métodos do estado-da-arte, contribuindo para melhorar os níveis de acurácia desses métodos.

Para trabalhos futuros pretendemos implementar outros descritores de textura, tais como o descritor LBP e suas variações. Além disso, é possível aplicar o método em outras arquiteturas de CNN. Outro experimento possível é a combinação da saída de várias CNN's para usar como entrada no algoritmo de classificação. Visando esclarecer esses resultados, para trabalhos futuros pode-se utilizar o *grid-search*, para estimar os melhores parâmetros de cada classificador.

Tabela 6 – Resultados obtidos pela metodologia proposta utilizando o classificador *Nearest Neighbors*.

	RIM-ONE-V2			Drishti		
	VGG16	VGG19	RESNET-50	VGG16	VGG19	RESNET-50
	Vermelho			Vermelho		
Acc	0,750±0,046	0,738±0,029	0,785±0,047	0,638±0,117	0,628±0,066	0,676±0,087
Kappa	0,480±0,098	0,452±0,064	0,553±0,092	0,146±0,205	0,122±0,161	0,236±0,208
AUC	0,736±0,049	0,723±0,033	0,772±0,044	0,581±0,113	0,559±0,093	0,622±0,109
F1	0,681±0,065	0,667±0,051	0,727±0,057	0,379±0,149	0,345±0,151	0,456±0,164
	Verde			Verde		
Acc	0,734±0,037	0,764±0,020	0,770±0,041	0,623±0,083	0,723±0,051	0,719±0,086
Kappa	0,448±0,080	0,507±0,049	0,528±0,085	0,175±0,176	0,285±0,148	0,229±0,263
AUC	0,720±0,041	0,748±0,028	0,763±0,043	0,600±0,097	0,635±0,071	0,605±0,115
F1	0,664±0,062	0,692±0,048	0,721±0,057	0,431±0,139	0,463±0,132	0,383±0,231
	Azul			Azul		
Acc	0,701±0,041	0,686±0,027	0,773±0,031	0,695±0,133	0,652±0,092	0,685±0,074
Kappa	0,389±0,087	0,351±0,061	0,527±0,055	0,250±0,262	0,092±0,205	0,160±0,216
AUC	0,694±0,045	0,674±0,031	0,759±0,027	0,646±0,152	0,542±0,096	0,568±0,121
F1	0,641±0,056	0,612±0,049	0,714±0,036	0,446±0,169	0,286±0,162	0,350±0,205
	WCH			WCH		
Acc	0,698±0,038	0,661±0,023	0,735±0,043	0,776±0,042	0,747±0,056	0,571±0,076
Kappa	0,386±0,071	0,336±0,054	0,429±0,093	0,465±0,090	0,358±0,115	0,045±0,152
AUC	0,682±0,031	0,660±0,037	0,702±0,048	0,713±0,044	0,675±0,060	0,536±0,092
F1	0,637±0,065	0,632±0,038	0,629±0,068	0,605±0,070	0,527±0,094	0,344±0,093
	GCH			GCH		
Acc	0,701±0,035	0,658±0,054	0,692±0,029	0,780±0,053	0,767±0,086	0,642±0,064
Kappa	0,386±0,074	0,293±0,102	0,342±0,062	0,391±0,223	0,406±0,154	0,070±0,167
AUC	0,698±0,042	0,642±0,058	0,667±0,036	0,684±0,105	0,687±0,065	0,538±0,081
F1	0,756±0,032	0,703±0,054	0,742±0,037	0,528±0,210	0,558±0,105	0,280±0,156
	WCG			WCG		
Acc	0,682±0,031	0,663±0,075	0,719±0,036	0,738±0,064	0,804±0,091	0,652±0,097
Kappa	0,352±0,079	0,328±0,142	0,403±0,082	0,301±0,209	0,463±0,185	0,211±0,183
AUC	0,673±0,038	0,667±0,072	0,693±0,040	0,655±0,108	0,703±0,083	0,596±0,083
F1	0,612±0,057	0,613±0,083	0,632±0,067	0,462±0,199	0,578±0,129	0,404±0,125

Tabela 7 – Resultados obtidos pela metodologia proposta utilizando o classificador AdaBoost.

	RIM-ONE-V2				Drishti	
	VGG16		VGG19	RESNET-50	VGG19	
	Vermelho				Vermelho	
Acc	0,773±0,028	0,802±0,052	0,773±0,042	0,738±0,113	0,590±0,080	0,680±0,136
Kappa	0,539±0,052	0,587±0,109	0,534±0,087	0,395±0,204	0,091±0,143	0,193±0,279
AUC	0,770±0,024	0,793±0,052	0,769±0,044	0,695±0,097	0,544±0,071	0,605±0,155
F1	0,740±0,026	0,752±0,069	0,730±0,056	0,551±0,166	0,365±0,122	0,412±0,210
Verde						
Acc	0,765±0,046	0,754±0,043	0,760±0,040	0,642±0,135	0,680±0,102	0,709±0,080
Kappa	0,523±0,094	0,502±0,087	0,493±0,081	0,152±0,295	0,203±0,197	0,256±0,218
AUC	0,762±0,048	0,750±0,042	0,746±0,040	0,572±0,141	0,597±0,097	0,619±0,108
F1	0,733±0,062	0,715±0,050	0,690±0,053	0,393±0,223	0,414±0,147	0,450±0,197
Azul						
Acc	0,764±0,029	0,716±0,043	0,794±0,024	0,657±0,089	0,695±0,074	0,680±0,079
Kappa	0,517±0,059	0,427±0,084	0,580±0,048	0,157±0,146	0,287±0,127	0,237±0,229
AUC	0,758±0,029	0,716±0,041	0,790±0,023	0,575±0,067	0,643±0,063	0,612±0,103
F1	0,723±0,045	0,681±0,048	0,759±0,028	0,389±0,113	0,493±0,093	0,446±0,192
WCH						
Acc	0,743±0,035	0,785±0,023	0,809±0,035	0,767±0,093	0,876±0,057	0,752±0,094
Kappa	0,475±0,085	0,563±0,068	0,598±0,079	0,433±0,164	0,716±0,116	0,403±0,223
AUC	0,746±0,042	0,779±0,035	0,793±0,041	0,731±0,101	0,850±0,062	0,717±0,128
F1	0,693±0,052	0,758±0,038	0,752±0,054	0,586±0,119	0,804±0,074	0,563±0,177
GCH						
Acc	0,748±0,045	0,750±0,032	0,829±0,024	0,780±0,095	0,738±0,090	0,647±0,080
Kappa	0,476±0,085	0,495±0,073	0,648±0,056	0,447±0,249	0,314±0,202	0,162±0,141
AUC	0,736±0,048	0,743±0,036	0,824±0,021	0,716±0,115	0,658±0,109	0,581±0,070
F1	0,767±0,040	0,785±0,023	0,846±0,027	0,606±0,229	0,472±0,180	0,406±0,098
WCG						
Acc	0,762±0,048	0,736±0,036	0,823±0,035	0,847±0,055	0,776±0,108	0,771±0,076
Kappa	0,513±0,023	0,459±0,070	0,648±0,072	0,583±0,182	0,434±0,260	0,396±0,164
AUC	0,758±0,056	0,720±0,039	0,824±0,034	0,800±0,110	0,715±0,138	0,698±0,078
F1	0,716±0,068	0,689±0,045	0,793±0,044	0,673±0,172	0,581±0,208	0,548±0,136

Mapas de textura como entrada em CNNs 3D aplicados para classificar nódulos em imagens de TC

Autores: Hélio Rocha Vieira de Couto Júnior; Romuere Rodrigues Veloso e Silva.

Resumo: O câncer de pulmão é a principal causa de morte relacionada ao câncer em todo o mundo. O diagnóstico precoce de nódulos pulmonares em tomografia computadorizada no tórax proporciona uma oportunidade de projetar um tratamento eficaz. Dessa forma, propõe-se a aplicação de mapas de textura diretamente aos nódulos 3D como pré-processamento de extração de características. Para isso, os padrões binários locais (LBP), com ramificações, como o uso de vizinhos com bordas (LBP-6), dimensões médias (LBP-M), e uma vizinhança de $3 \times 3 \times 3$ (LBP- 3×3), para destacar a textura do nódulo. Redes neurais convolucionais, como DenseNet, ResNet e LeNet, foram usadas como extratores de atributos usando os mapas de textura 3D calculados. Em seguida, esses atributos profundos são usados como entrada para treinar um classificador Random Forest. Nos experimentos, é utilizada a base de imagens LIDC-IDRI. Dois processos de segmentação foram utilizados, um realizado por radiologistas, presente na própria base (B1), e outro realizado automaticamente por um método da literatura (B2). Em B1, os melhores resultados foram os atributos dos nódulos originais extraídos com a arquitetura DenseNet atingindo uma acurácia de 0,8371, especificidade de 0,9130, sensibilidade de 0,7328 e Kappa de 0,6591. Em B2, os melhores resultados foram uma combinação de atributos do nódulo original combinado com o LBP-6 extraído com arquitetura LeNet que atingiu uma acurácia de 0,9037, uma especificidade de 0,8453, sensibilidade 0,9266 e Kappa de 0,7641. Em conclusão, é possível melhorar a precisão da classificação incluindo um cálculo de mapa de textura como parte do processo.

Keywords: Mapas de Textura. CNN. TC. Descritor de Textura.

2.1 Introdução

De acordo com a American Cancer Society, cólon e reto, mama, próstata, pulmão e brônquios são os quatro principais tipos de câncer de morte ([SIEGEL et al., 2017](#)). O câncer de pulmão é um dos principais tipos de

câncer (YANG; CHEN, 2015), e seu número estimado de mortes foi de 155.870 em 2017 (representando cerca de 26% das mortes de quatro cânceres primários) (SIEGEL et al., 2017). A maioria dos pacientes apresenta mau prognóstico devido ao diagnóstico feito em estágio avançado da doença. Os pacientes diagnosticados em um estágio inicial têm significativamente mais chance de sobrevivência (cerca de 70% em 5 anos); no entanto, os sintomas clínicos do câncer de pulmão geralmente se apresentam em estágio avançado.

Com o avanço da tecnologia de tomografia computadorizada (TC), o espaço entre as camadas digitalizadas está se tornando menor. Com esse avanço o número de camadas teve um aumento de imagens de TC. Com isso os médicos precisam de muito mais tempo para detectar e diagnosticar a doença. A tarefa de encontrar nódulos e classificá-los entre benignos e malignos é árdua para o olho humano, tornando-se uma oportunidade de usar sistemas de computador para auxiliar nessas tarefas.

Os métodos de diagnóstico auxiliado por computador (CAD) são geralmente baseados em dois métodos existentes: modelos de classificação baseados em atributos aritméticos e/ou com atributos extraídos de forma manual (KRISHNAMURTHY; NARASIMHAN; RENGASAMY, 2016; SHEWAYE; MEKONNEN, 2016); e usando rede neural profunda com extração automática de atributos (SHEN et al., 2015; NIBALI; HE; WOLLERSHEIM, 2017; LIU; KANG, 2017). Os métodos da primeira categoria geralmente medem características radiológicas, como tamanho do nódulo, localização, forma e textura. Na segunda categoria, os modelos baseados em redes neurais profundas podem aprender automaticamente os atributos de diagnóstico de imagens de TC de pulmão. No entanto, a maioria dos modelos de aprendizado profundo existentes têm problemas na classificação de nódulos pulmonares usando redes neurais convolucionais 2D (CNNs) (SHEN et al., 2015) ou com múltiplas visualizações para imitar volumes de imagens 3D (NIBALI; HE; WOLLERSHEIM, 2017; LIU; KANG, 2017).

Neste trabalho, é comparado o desempenho dos métodos do mapa de textura binária local (LBP) (OJALA; PIETIKÄINEN; MÄENPÄÄ, 2002) para imagens 3D usadas como entrada para três arquiteturas de aprendizado profundo 3D: DenseNet, ResNet e LeNet.

2.1.1 Objetivos

Desenvolver um método de caracterização de imagens para classificação de nódulos pulmonares entre benigno e maligno baseado na aplicação de mapas de textura.

Os objetivos específicos deste trabalho são:

1. Avaliar diferentes métodos para a caracterização da textura dos nódulos pulmonares;
2. Analisar o desempenho de diferentes arquiteturas de CNNs para extração de atributos;
3. Classificar as imagens de nódulos utilizando algoritmos de aprendizagem de máquina.

2.1.2 Organização do Trabalho

Na Seção 2.2 explanamos sobre a base de dados, métodos LBP, conceitos e arquiteturas CNN e um classificador utilizado no trabalho. Na Seção 2.3 têm-se uma revisão da literatura para uma comparação dos métodos propostos com o apresentado. Apresentamos o método na Seção 2.4 com as métricas selecionadas para a avaliação. Na Seção 2.5 foram analisados os resultados obtidos a partir da execução da metodologia proposta. Por fim, na Seção 2.6 foram apresentadas as observações finais sobre o método proposto.

2.2 Referencial Teórico

Para melhor compreensão do assunto a ser tratado neste trabalho, é necessário detalhar a base de dados LIDC-IDRI, métodos de processamento de imagens, o LBP, e Rede Neural Convolucional (CNN).

2.2.1 LIDC-IDRI

A coleção de imagens do LIDC Image Database Consortium (LIDC-IDRI) consiste em exames de tomografia computadorizada (TC) de tórax para diagnosticar e rastrear câncer de pulmão com nódulos segmentados.

Este conjunto de dados de imagem contém 1.018 casos. Cada caso contém imagens de uma TC e um arquivo XML associado que registra os resultados de um processo de anotação de imagem realizado por quatro radiologistas experientes. Na fase inicial da leitura às cegas, cada radiologista revisou independentemente cada TC e marcou lesões pertencentes a uma das três categorias (nódulo ≥ 3 mm, nódulo < 3 mm e não nódulo ≥ 3 mm) (III et al., 2011a). Na outra etapa, cada radiologista revisou independentemente suas marcas, juntamente com as marcas anônimas dos outros três radiologistas, para dar uma opinião final. Uma imagem do conjunto de dados foi mostrada na Figura 12.

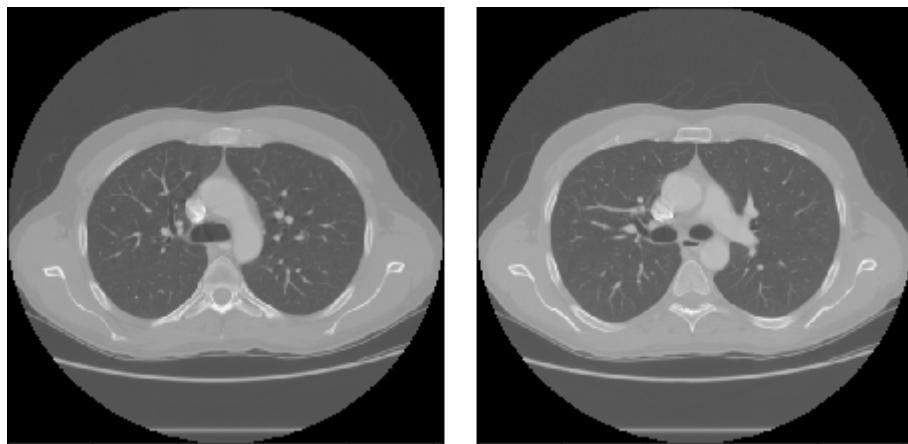


Figura 12 – Exemplo de imagem de TC da LIDC-IDRI.

Neste trabalho, são utilizadas imagens de nódulos 3D da base de dados LIDC-IDRI. Nesta base de dados, o grau de malignidade dos nódulos foi classificado na faixa de 1 a 5, representando o nível de probabilidade do nódulo ser maligno. É considerado 1 e 2 benigno e 4 e 5 como maligno (B1). Como os nódulos com grau 3 são classificados pelos radiologistas como indefinido, não foram utilizados neste trabalho. Para a segmentação foi utilizado 50% de concordância dos radiologistas como mostrado na Figura 13.

A segmentação do trabalho (FILHO et al., 2014) também foi usada para ser testada por uma segmentação feita automaticamente (B2), essa segmentação é baseada em técnicas de processamento de imagens e reconhecimento de padrões e pode ser resumida em três etapas. Na primeira etapa, é realizada a extração e reconstrução do parênquima pulmonar e posteriormente realçada para evidenciar suas estruturas. No segundo estágio, os candidatos a nódulos são segmentados. E terceiro estágio, os recursos de forma e textura são extraídos, selecionados e então classificados usando uma máquina de vetores de suporte. O experimento B1 resultou em 333 nódulos benignos e 462 nódulos malignos e o B2 em 956 nódulos benignos e 390 nódulos malignos.

2.2.2 LBP

Primeiramente, é apresentado o LBP 2D; a transformação é usada com base nos padrões binários locais (LBP) (OJALA; PIETIKÄINEN; HARWOOD, 1996). Em geral, o LBP mede estruturas locais em um determinado pixel usando P amostras em um círculo de raio R ao redor do pixel e resulta em informações com um código para estrutura doméstica ou padrão local. O operador detecta microestruturas na imagem em diferentes resoluções reguladas pelo parâmetro R , por exemplo, pontos, arestas, cantos. Para obter o valor do LBP, um limiar na vizinhança é feito usando o pixel central e, em seguida, multiplicando pela ponderação binária. Como exemplo dos pontos de amostragem $P = 8$ e raio $R = 1$, o cálculo do valor LBP é ilustrado na Figura 14.

Para este trabalho, são utilizados e comparados três métodos LBP para imagens 3D. No primeiro método ($LBP - 6$), são utilizados apenas os vizinhos que fazem fronteira com o pixel central, resultando em um código binário de 6 dígitos. Isso permite novos valores entre 0 e 63; um exemplo está na Figura 15a, onde o pixel central é usado como uma referência para pixels vizinhos para binarizar e calcular um novo valor para o centro.

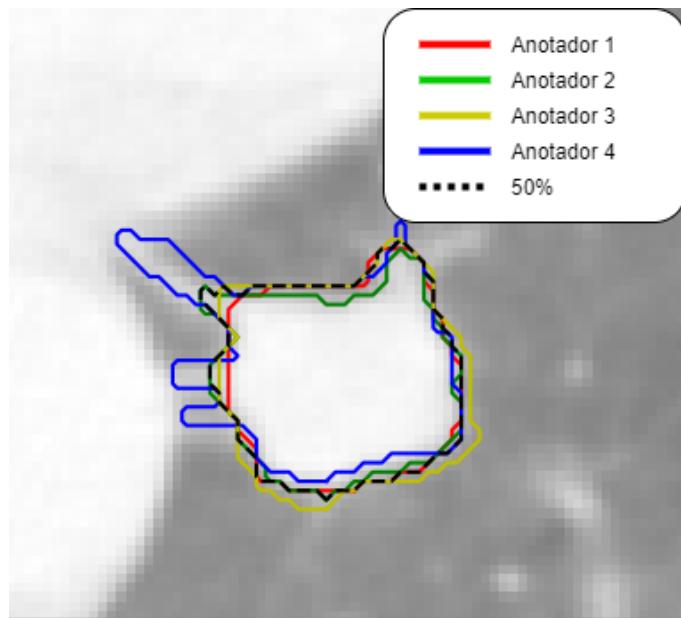


Figura 13 – Representação gráfica das anotações juntamente com 50% de concordância das anotações.

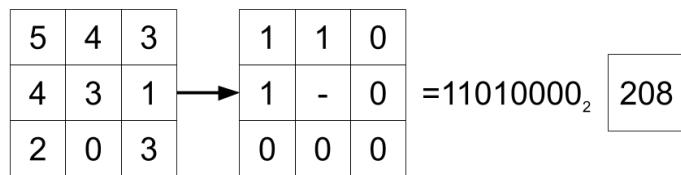


Figura 14 – Transformação LBP com $P = 8$ e $R = 1$.

O segundo método ($LBP - M$) é semelhante ao apresentado no trabalho de [Arai, Herdiyeni e Okumura \(2012\)](#). A imagem 3D é transformada em três imagens 2D dos eixos X, Y e Z da imagem e aplicado o LBP em cada uma delas, os três valores obtidos foram retirados da média e referidos ao pixel central, conforme mostrado na Figura 15b.

O terceiro método ($LBP - 3\times$) é baseado no trabalho de [Zhao e Pietikainen \(2007\)](#). É usado como um cubo com as dimensões $3\times3\times3$ ou os 26 pixels vizinhos e calcula o LBP. Resulta em grandes números muito distintos uns dos outros, conforme mostrado na Figura 15c, tendo uma variação de 0 e 2^{26} para cada pixel.

2.2.3 CNN

Uma CNN é um tipo particular de rede neural de várias camadas projetada para reconhecer padrões visuais diretamente de imagens. Inicialmente introduzida por [LeCun et al. \(1998\)](#), a arquitetura CNN facilita o processamento da percepção humana utilizando campos receptores locais. Muitos modelos de classificação de imagens foram introduzidos para ajudar a resolver o problema de reconhecimento. A classificação de imagens é um dos principais problemas da Visão Computacional, com uma grande variedade de aplicações práticas. O objetivo da arquitetura da CNN é dimensionar a imagem para um tipo que possa ser processado facilmente sem comprometer as características vitais para obter uma previsão correta. O método CNN funciona passando a imagem de entrada para o conjunto de várias camadas, como camada de convolução, camada de agrupamento e camada totalmente conectada para fornecer um resultado correto. Neste trabalho, são criadas CNNs 3D baseadas em CNNs 2D que serão utilizadas para treinamento e extração de atributos de imagens.

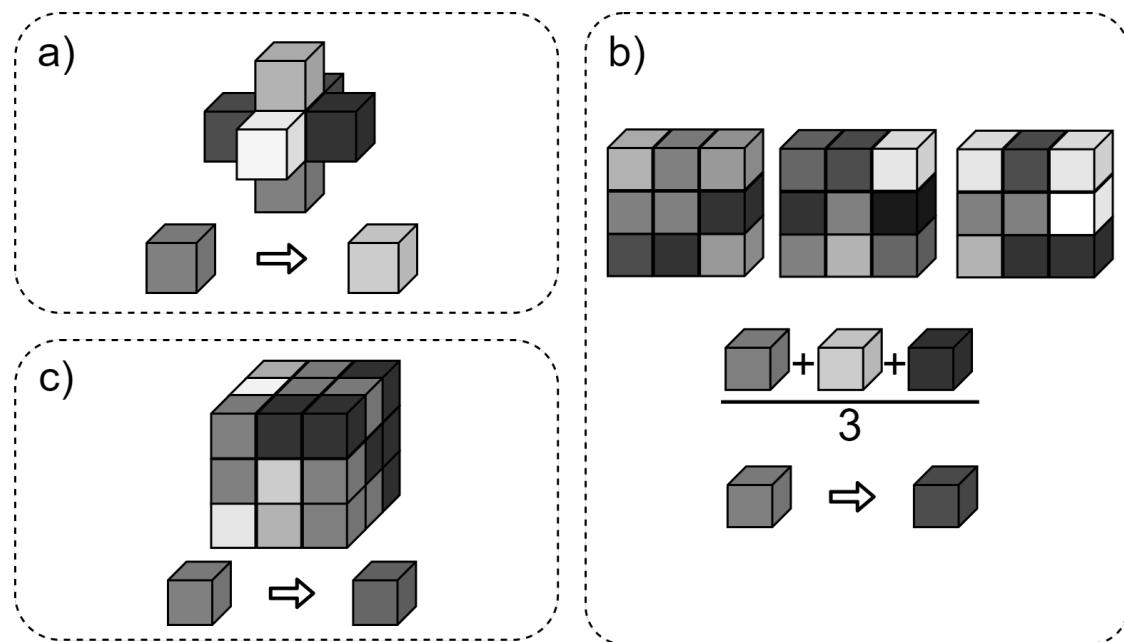


Figura 15 – Representação gráfica dos métodos LBP 3D. a) LBP-6; b) LBP-M; c) LBP – 3x

2.2.3.1 Camada de convolução

A função da camada de convolução é coletar atributos válidos das imagens de entrada e transferi-los para a camada seguinte. Esta camada mantém uma relação espacial de pixel observando as características da imagem. O filtro escolhido é aplicado à imagem inserida e, assim, obtém um recurso desejado. Os filtros enxergam pequenos quadrados e vão “escorregando” por toda a imagem captando os traços mais marcantes. Este processo também é denominado “mapa de atributos” ou “mapa de ativação”. A Figura 16 mostra um filtro 3×3 aplicado a uma imagem 5×5 que fornece um resultado para o mapa da convolução com 3×3 .

Filtro	Imagen	Resultado
1 0 1 0 1 0 1 0 1	1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 1 1 0 0 1 1 0 0	4 3 4 2 4 3 2 3

Figura 16 – Representação de uma convolução. Com o filtro percorrendo a imagem e gerando um resultado.

A CNN executa diferentes filtros com várias entradas de convolução e, portanto, resultam em vários mapas resultantes. Todos esses mapas são combinados para formar o desempenho final das camadas de convolução.

2.2.3.2 Pooling

A intenção desta camada é isolar os mapas de feições somando ou calculando a média dos valores em todos os mapas de atributos. A camada de pooling concentra-se na produção de atributos versáteis, minimizando a dimensionalidade espacial. Duas técnicas de agrupamento amplamente utilizadas são por média e máxima. O pooling médio (MediaPooling) funciona calculando a média de cada bloco de atributos. O pooling máximo

(MaxPooling) funciona calculando o valor máximo de cada bloco de atributos. Podemos ver o resultado dessas técnicas na Figura 17.

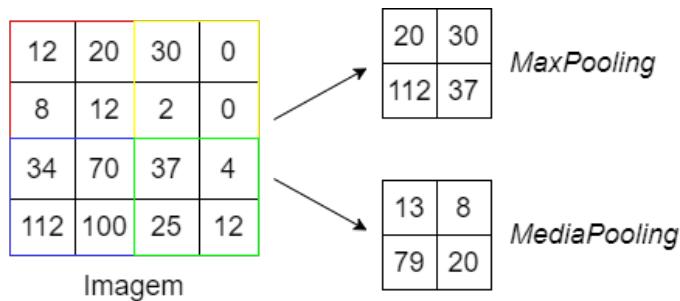


Figura 17 – Representação gráfica do *Pooling*. Cada bloco de cor representa onde será aplicado cada *pooling* na imagem.

2.2.3.3 Totalmente conectada

No final da CNN depois das Convoluçãoções de *poolings* tem-se uma rede neural. A rede neural tem entradas 1D, por esse motivo é preciso agrupar os mapas de textura 2D para o 1D. Para isso a camada *Flatten* é utilizada. Na Figura 18 temos um exemplo de como o *Flatten* realiza esta operação.

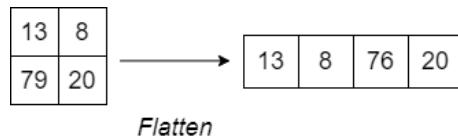


Figura 18 – Representação gráfica do *Flatten*.

A função da camada totalmente conectada depende integralmente dos resultados da camada de convolução e *pooling*. Esta é a camada final na qual todos os mapas de características são usados e preparados para a parte de classificação.

2.2.3.4 Arquiteturas CNNs

A CNN é um tipo especial de Redes Neurais, que tem mostrado desempenho exemplar em várias competições relacionadas à Visão Computacional e ao Processamento de Imagens (KHAN et al., 2020). Algumas das áreas de aplicação da CNN incluem classificação e segmentação de imagens, detecção de objetos, processamento de vídeo, processamento de linguagem natural e reconhecimento de fala. A poderosa capacidade de aprendizado da CNN profunda se deve principalmente ao uso de vários estágios de extração de atributos que podem aprender automaticamente as representações dos dados. Várias ideias inspiradoras para trazer avanços nas CNNs foram exploradas, como o uso de diferentes funções de ativação e perda, otimização de parâmetros, regularização e inovações arquitetônicas. No entanto, a melhoria significativa na capacidade de representação da CNN profunda é alcançada por meio de inovações nas arquiteturas.

Para este trabalho será utilizado três arquiteturas CNNs: DenseNet, ResNet e LeNet, sendo que a DenseNet e a ResNet são CNNs profundas tendo varias camadas de convoluções, algumas arquiteturas CNNs quando há um grande número de convoluções alguns dos atributos das imagens vão desaparecendo. Para evitar esse problema essas arquiteturas “enviam” o resultado das suas convoluções para frente e para as convoluções mais a frente, fazendo com que os atributos obtidos com as convoluções iniciais não desapareça. A LeNet é uma CNN básica com poucas convoluções.

2.2.3.5 LeNet

LeNet proposta por (LECUN et al., 1998), é uma arquitetura CNN simples contendo camadas essenciais, como a camada de convolução, camada de *pooling* e camada densa, estabelecendo uma base para o desenvolvimento de outras CNNs.

A arquitetura é composta por um empilhamento de camadas de convolução, onde 3×3 filtros são aplicados para o processo de convolução, que tem o passo fixado em 1 pixel. O agrupamento espacial é feito por camadas de *pooling*, que seguem as camadas de convolução. O MaxPooling realizado em uma janela de 2×2 pixels. A Figura 359 ilustra a organização das camadas na arquitetura da LeNet, onde é possível observar que a rede contém três blocos diferentes sendo dois de convolução (a) seguidos de *Pooling*. (b) Um bloco de uma camada *Flatten*, onde os mapas de textura 3D convertidos em uma lista 1D de atributos que representam a imagem de entrada, seguida por três camadas densas (c), ou totalmente conectadas, onde a imagem de entrada será classificada.

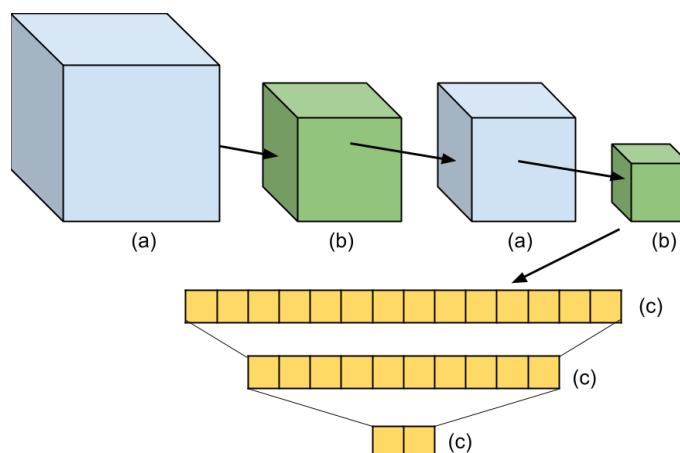


Figura 19 – Arquitetura LeNet. (a) camada de convolução 3D, (b) camada de agrupamento 3D, (c) camada totalmente conectada.

2.2.3.6 ResNet

A ResNet é uma arquitetura CNN apresentada por (HE et al., 2015a). Redes residuais são uma rede neural clássica usada para muitas tarefas de visão computacional. O avanço fundamental com ResNet permitiu treinar redes neurais intensas com mais de 150 camadas. Antes da ResNet, redes neurais profundas eram difíceis de treinar devido ao problema de gradientes de desaparecimento. É baseado em redes simples, que inserem “atalhos” fazendo com que a rede se torne residual, as saídas de uma camada são entradas de camadas mais abaixo, como ilustrado na Figura 20. Atalhos podem ser aplicados quando as entradas e saídas têm as mesmas dimensões, mesmo que não tenham as mesmas dimensões.

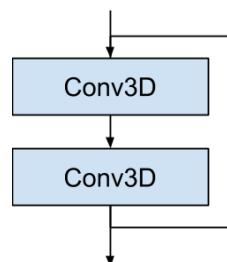


Figura 20 – ResNet *residual block*.

A Figura 21 mostra a arquitetura ResNet, contendo blocos residuais (ResBlock) já representados na Figura 20. Cada ResBlock pode ter uma quantidade diferente de camadas convolucionais e tendo no início uma camada de *Pooling*. A primeira camada convolucional da arquitetura possui um filtro de 7×7 , seguida dos blocos ResBlock e, por último, a última camada de *Pooling* e *Dense*, onde será classificada a imagem de entrada.

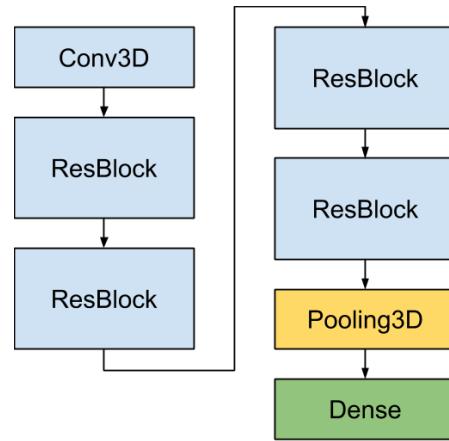


Figura 21 – Ilustração da arquitetura ResNet.

2.2.3.7 DenseNet

No trabalho desenvolvido por (HUANG et al., 2017a), é proposta a arquitetura DenseNet. Essa CNN tem como principal característica conectar as camadas com todas as suas camadas frontais. DenseNet é composta de blocos densos, conforme mostrado na Figura 22. Nestes blocos, as camadas estão densamente conectadas. Cada camada recebe na entrada todos os mapas de atributos de saída das camadas anteriores. Esse uso excessivo de resíduos cria uma supervisão em profundidade, pois cada camada recebe mais supervisão da função de perda por causa das conexões mais curtas. Um bloco denso é um grupo de camadas conectadas a todas as suas camadas anteriores (HUANG et al., 2017a).

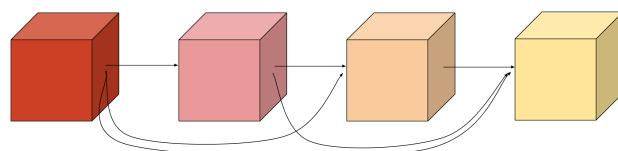


Figura 22 – Exemplo de bloco denso.

Uma única camada possui a normalização e as convoluções. Em vez de somar o residual como na ResNet, a DenseNet concatena todos os mapas de atributos. A DenseNet apresenta várias vantagens claras, como alívio ou problema de desaparecimento, como as camadas da camada de convolução utilizadas nas camadas após perda de dados. Esta propriedade reforça a propagação de atributos, estimula a reutilização de atributos e reduz substancialmente o número de parâmetros. A Figura 23 mostra a arquitetura do DenseNet semelhante à arquitetura da ResNet 2.2.3.6, onde no final de cada bloco denso, representado na Figura 22 tem uma camada de convolução seguida por um *Pooling* para aplicar normalização. A imagem resultante é diminuída para o próximo bloco denso.

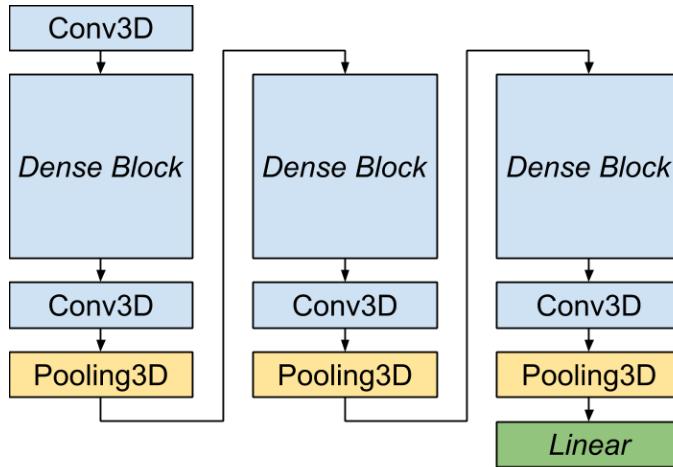


Figura 23 – Arquitetura DenseNet.

2.2.4 Random Forest

A *Random Forest* (RF) (??) é um classificador baseado em árvores de decisão. A popularidade dos modelos de árvore de decisão na classificação se deve à sua simplificação no algoritmo e flexibilidade no manuseio de diferentes tipos de atributos de dados. No entanto, o modelo de árvore única é possivelmente sensível a dados de treinamento específicos e fácil de ajustar (BHATTACHARYYA et al., 2011). RF é uma combinação de vários preditores de árvore de forma que cada árvore dependa de um conjunto de dados independente aleatório e todas as árvores na floresta tenham a mesma distribuição. A capacidade da RF não depende apenas da força da árvore individual, mas também da correlação entre as diferentes árvores. Quanto mais forte a resistência de uma única árvore e menor a correlação de diferentes árvores, melhor será o desempenho da RF. Embora possivelmente existam algumas instâncias com classificação incorreta em nosso conjunto de dados, a floresta aleatória ainda é robusta a ruído.

2.3 Trabalhos Relacionados

No trabalho de Al-Shabi et al. (2019), um novo método foi proposto para prever a malignidade de nódulos que tem a capacidade de analisar a forma e o tamanho de um nódulo usando um extrator de característica global, bem como a densidade e estrutura do nódulo usando um extrator de característica local. A proposta é o uso de blocos residuais com um tamanho de bloco 3×3 para extração de atributos locais e blocos não locais para extraer os atributos globais. O Bloco Não Local tem a capacidade de extrair atributos globais sem usar um grande número de parâmetros. A ideia principal por trás do Bloco Não Local é aplicar multiplicações de matrizes entre atributos nos mesmos mapas de atributos. O conjunto de dados utilizado foi o LIDC-IDRI. O método proposto alcançou acurácia de 0,8846, uma especificidade de 0,8738 e sensibilidade de 0,8866.

No artigo de Zhai et al. (2020), é proposto uma nova estrutura de rede neural convolucional multireta (MT-CNN) para identificar nódulos malignos de nódulos benignos em TC. O MT-CNN aprende as características do nódulo pulmonar tridimensional (3D) a partir de nove visualizações bidimensionais (2D), que são decompostas em diferentes ângulos de cada nódulo. Cada um dos modelos 2D MT-CNN consiste em duas divisões, uma é a divisão de classificação do nódulo e a outra é a reconstrução da imagem. Onde a motivação da reconstrução é preservar mais informações microscópicas na estrutura hierárquica do CNN, o que é benéfico para a identificação de nódulos malignos. O resultado final da classificação é obtido integrando nove modelos 2D. Testaram o método nos conjuntos de imagens LUNA-16 e LIDC-IDRI. A MT-CNN atinge uma taxa de sensibilidade de 0,8774 e uma especificidade de 0,8887.

Dey, Lu e Hong (2018), consideram o problema da classificação entre nódulos pulmonares benignos e malignos em imagens de TC, que visa aprender um mapeamento direto de imagens 3D para rótulos de classe. Para isso quatro redes neurais convolucionais (CNN), incluindo uma CNN 3D básica, uma nova rede

de múltiplas saídas, uma DenseNet 3D e uma DenseNet 3D com múltiplas saídas. Essas quatro redes foram avaliadas no conjunto de imagens LIDC-IDRI. A DenseNet 3D de múltiplas saídas (MoDenseNet) atingiu uma acurácia de 0,8545 para o diagnóstico de nódulo pulmonar.

A estrutura proposta por [Afshar et al. \(2020\)](#), conhecida como 3D *Multi-scale Capsule Network* (3D-MCN), foi projetada exclusivamente para se beneficiar de: (i) entradas 3D, fornecendo informações sobre o nódulo em 3D; (ii) entrada em escala múltipla, capturando as características locais do nódulo, bem como as características dos tecidos circundantes, e; (iii) arquitetura baseado em CapsNet, sendo capaz de lidar com um pequeno número de amostras de treinamento. A arquitetura 3D-MCN proposta previu malignidade do nódulo pulmonar com acurácia de 0,9312, sensibilidade de 0,9494 e especificidade de 0,9000 quando testada no conjunto de dados LIDC-IDRI.

Dante dos trabalhos relacionados, pode-se observar que suas propostas são referentes a arquitetura CNN. Porém, não foram encontradas pesquisas com a aplicação de mapas de textura antes de passar pela CNN. Assim, este trabalho tem como objetivo a utilização do mapa de textura nas imagens originais do nódulo para serem posteriormente utilizadas na CNN.

2.4 Metodologia Proposta

Neste trabalho, é proposto o uso de mapas de textura LBP como entrada em CNNs 3D. Desta forma, as arquiteturas CNN conhecidas são treinadas com o objetivo de extrair os atributos do mapa de textura para classificar os nódulos pulmonares em malignos e benignos. Figura 24 mostra as principais etapas da nossa metodologia.

Para validar nossa metodologia, é aplicado o banco de dados LIDC-IDRI de duas formas. O primeiro usa a segmentação do especialista da base de dados LIDC-IDRI; esta versão é chamada B1. Nesse caso, o desempenho da nossa metodologia é avaliado usando a melhor segmentação disponível para este banco de dados. No entanto, percebe-se nos trabalhos relacionados que a segmentação dos nódulos pulmonares é uma tarefa desafiadora. Além disso, está sujeito a erros e limites incertos. Por este motivo, também são utilizados nódulos obtidos pelo método de segmentação B2 ([FILHO et al., 2014](#)).

O próximo passo da metodologia proposta é o cálculo do mapa de textura. Este procedimento visa melhorar a entrada das CNNs. Como mostrado em trabalhos de visualização, as características de textura são a base da classificação do nódulo. A hipótese deste trabalho é que melhorar a entrada de CNNs usando mapas de textura (LBP) em vez de dados brutos aumentará as taxas de classificação. São usadas três versões de um LBP 3D: 1) LBP-6; 2) LBP-M; 3) LBP-3×. Em seguida, são utilizadas três arquiteturas CNN: 1) LeNet; 2) ResNet; e 3) DenseNet. Essas arquiteturas produziram atributos para cada nódulo e estes foram utilizados para treinar um classificador Random Forest (RF).

As CNNs foram implementadas com base nos modelos 3D de cada uma. As entradas da CNN foram definidas de acordo com o tamanho do maior nódulo. No treinamento foi usado 100 épocas com o otimizador Adam a uma taxa de aprendizagem de 1e-5. Para o treinamento da CNN, são usados 60% das imagens para treinamento e 20% para validação. Para o treinamento do classificador RF, utiliza-se os mesmos 60% das imagens de treinamento das CNNs, portanto não houve viés nos resultados e 20% para os testes. Este processo foi repetido 5 vezes para uma avaliação mais concreta.

2.4.1 Métricas de avaliação

Este trabalho visa diferenciar nódulos em benignos e malignos, o que significa um problema de classificação binária. Como forma de avaliar os resultados da classificação, pretende-se utilizar, Acurácia (Ac), Especificidade (Espe), Sensibilidade (Sens) e *Kappa* (*K*). Para isso será utilizado a matriz de confusão ou tabela de confusão, que tem como objetivo calcular a quantidade de Falso Positivo (FP), quando um nódulo benigno é classificado como maligno; Falso Negativo (FN), quando um nódulo maligno é classificado como benigno; Verdadeiro Positivo (VP,) quando um nódulo maligno é classificado corretamente e Verdadeiro Negativo (VN) quando classifica um nódulo benigno.

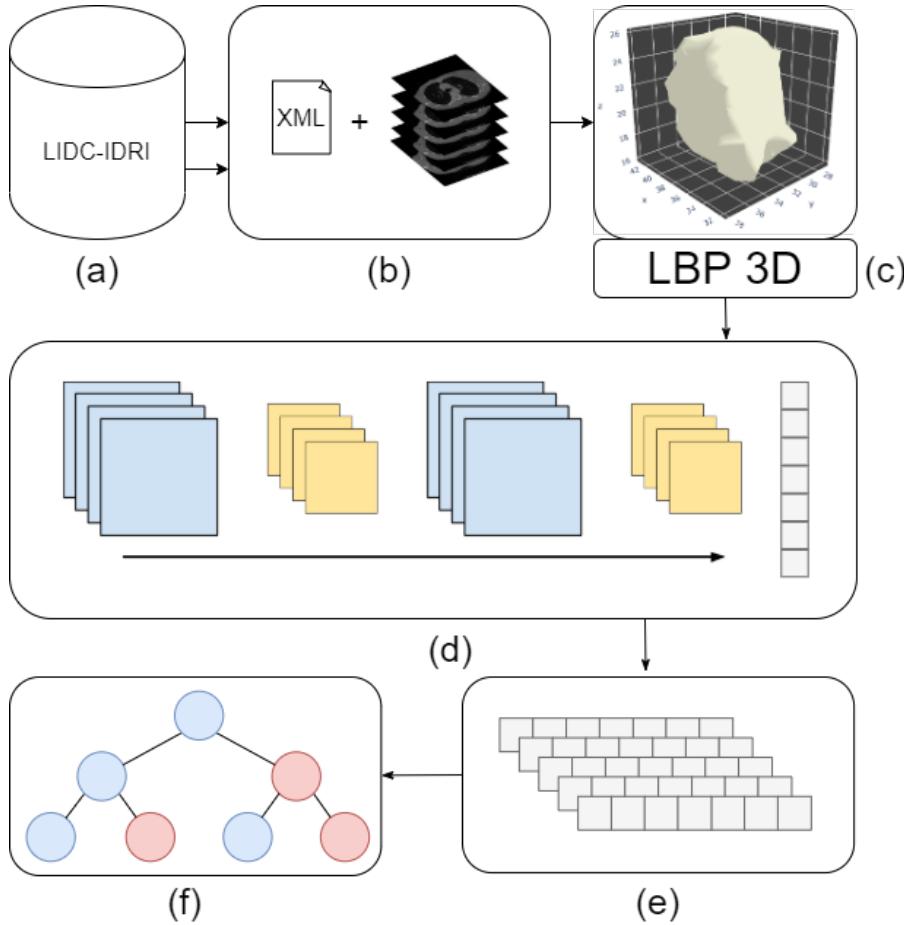


Figura 24 – Metodologia proposta. (a) a base de dados LIDC-IDRI. (b) Informações de especialistas em diagnóstico e segmentação de nódulos. (c) Aplicação de métodos LBP-3D. (d) Extração dos atributos de cada nódulo usando arquiteturas CNN. (e) Agrupar as características para treinamento / teste do classificador. (f) classificador RF para diferenciar nódulos de malignos e benignos.

A Sens, representada pela Equação 2.1, é a taxa de nódulos malignos classificados corretamente.

$$Sens = \frac{VP}{(VP + FN)}. \quad (2.1)$$

A Espe, representada pela Equação 2.2, é a taxa de nódulos benignos classificados corretamente.

$$Espe = \frac{VN}{(VN + FP)}. \quad (2.2)$$

A Ac, representada pela Equação 2.3, é a proporção de nódulos calculados corretamente.

$$Ac = \frac{(VP + VN)}{(VP + FP + FN + VN)}. \quad (2.3)$$

O κ , representado pela Equação 2.4, é uma pontuação que expressa a concordância entre dois anotadores (classe real e a previsão) em um problema de classificação. Para o cálculo usa-se a Ac e Acurácia Aleatória (AA). O cálculo do AA está na Equação 2.5.

$$\kappa = \frac{Ac - AA}{1 - AA}, \quad (2.4)$$

$$AA = \frac{(VN + FP) * (VN + FN) + (FN + VP) * (FP + VP)}{(VP + FP + FN + VN)^2} \quad (2.5)$$

O coeficiente κ é interpretado usando as diretrizes descritas por (LANDIS; KOCH, 1977b), onde a força do coeficiente κ é interpretada da seguinte maneira: 0, 01 – 0, 20, leve; 0, 21 – 0, 40, justo; 0, 41 – 0, 60, moderado; 0, 61 – 0, 80, substancial; e 0, 81 – 1, 00, quase perfeito.

2.5 Resultados e Discussões

Para realizar os experimentos, são utilizadas sete entradas diferentes para as CNNs: 1) o volume do nódulo original; 2) LBP – 6; 3) LBP – M; 4) LBP – 3 \times ; 5) concatenação da característica de 1) e 2); 6) concatenação do recurso de 1) e 3); 7) concatenação do recurso de 1) e 4). Além disso, os experimentos foram realizados usando a segmentação disponível no LIDC-IDRI (B1) e usando a segmentação de (FILHO et al., 2014) (B2). Os resultados experimentais são apresentados nas Tabelas 8 e 9.

Tabela 8 – Análise do desempenho da nossa metodologia em B1.

	<i>Sens</i>	<i>Espe</i>	<i>Ac</i>	κ
Original				
DenseNet	0,7328±0,01	0,9130±0,00	0,8371±0,00	0,6591±0,01
ResNet	0,7433±0,01	0,8772±0,00	0,8208±0,00	0,6280±0,00
LeNet	0,8179±0,01	0,8402±0,01	0,8308±0,01	0,6548±0,02
LBP-6				
DenseNet	0,7418±0,01	0,8315±0,00	0,7937±0,00	0,5754±0,01
ResNet	0,7478±0,01	0,8402±0,00	0,8013±0,00	0,5906±0,01
LeNet	0,7701±0,01	0,7989±0,00	0,7868±0,00	0,5655±0,01
LBP-M				
DenseNet	0,7090±0,01	0,8543±0,01	0,7931±0,00	0,5704±0,01
ResNet	0,7343±0,01	0,8217±0,00	0,7849±0,00	0,5576±0,01
LeNet	0,7687±0,02	0,7989±0,01	0,7862±0,00	0,5641±0,01
LBP-3 \times				
DenseNet	0,7000±0,02	0,8576±0,00	0,7912±0,00	0,5656±0,02
ResNet	0,7313±0,01	0,8630±0,00	0,8075±0,00	0,6010±0,02
LeNet	0,7582±0,01	0,8196±0,01	0,7937±0,00	0,5773±0,01
Original+LBP-6				
DenseNet	0,7343±0,01	0,8457±0,00	0,7987±0,00	0,5842±0,01
ResNet	0,7493±0,01	0,8478±0,00	0,8063±0,00	0,6005±0,01
LeNet	0,8194±0,01	0,8185±0,01	0,8189±0,00	0,6320±0,01
Original+LBP-M				
DenseNet	0,7403±0,01	0,8772±0,00	0,8195±0,00	0,6252±0,01
ResNet	0,7209±0,00	0,8304±0,01	0,7843±0,00	0,5549±0,01
LeNet	0,8224±0,01	0,8337±0,01	0,8289±0,01	0,6516±0,02
Original+LBP-3 \times				
DenseNet	0,6970±0,01	0,8533±0,00	0,7874±0,00	0,5579±0,0
ResNet	0,7373±0,01	0,8717±0,00	0,8151±0,00	0,6163±0,01
LeNet	0,8164±0,02	0,8370±0,00	0,8283±0,01	0,6498±0,02

Em negrito são os melhores resultados.

Analizando os resultados do experimento usando B1, é possível ver que os melhores resultados foram obtidos sem mapas de textura em termos de *Spec*, *Acc* e κ . Em relação a *Sens*, as características de concatenação de volume de nódulo LBP-M alcançam o melhor resultado. Porém, os nódulos deste experimento possuem a melhor segmentação possível facilitando o processo de classificação. Em relação aos experimentos com B2, uma pequena melhora nos resultados foram obtidos utilizando o volume original com o LBP-M. Neste experimento, os nódulos foram segmentados por meio de um método computacional, o que significa que esses resultados estão mais próximos de um ambiente real. Porém, o volume original obteve a melhor Especificidade (*Spec* = 0, 9040).

Tabela 9 – Análise do desempenho da nossa metodologia em B2.

	<i>Sens</i>	<i>Espe</i>	<i>Ac</i>	κ
Original				
DenseNet	0,8906±0,00	0,7987±0,01	0,8648±0,00	0,6731±0,01
ResNet	0,8922±0,00	0,9040±0,00	0,8955±0,00	0,7548±0,00
LeNet	0,9203±0,01	0,8547±0,01	0,9019±0,00	0,7615±0,01
LBP-6				
DenseNet	0,9057±0,00	0,8013±0,00	0,8764±0,00	0,698±0,00
ResNet	0,9099±0,00	0,8427±0,00	0,891±0,00	0,7361±0,00
LeNet	0,9224±0,00	0,8333±0,02	0,8974±0,00	0,7484±0,02
LBP-M				
DenseNet	0,8958±0,00	0,7640±0,00	0,8588±0,00	0,6538±0,01
ResNet	0,9005±0,00	0,8600±0,01	0,8891±0,00	0,7348±0,01
LeNet	0,9193±0,00	0,8320±0,01	0,8948±0,00	0,7425±0,01
LBP-3x				
DenseNet	0,9141±0,00	0,8227±0,00	0,8884±0,00	0,7273±0,00
ResNet	0,9057±0,00	0,8547±0,00	0,8914±0,00	0,7388±0,00
LeNet	0,9198±0,00	0,8427±0,01	0,8981±0,00	0,7515±0,01
Original+LBP-6				
DenseNet	0,9073±0,00	0,8067±0,01	0,8790±0,00	0,7045±0,01
ResNet	0,9021±0,00	0,8547±0,00	0,8888±0,00	0,7332±0,01
LeNet	0,9266±0,00	0,8453±0,01	0,9037±0,00	0,7641±0,01
Original+LBP-M				
DenseNet	0,9068±0,00	0,7693±0,00	0,8682±0,00	0,6745±0,00
ResNet	0,9005±0,00	0,8573±0,00	0,8884±0,00	0,7328±0,00
LeNet	0,9130±0,00	0,8520±0,02	0,8959±0,00	0,7479±0,02
Original+LBP-3x				
DenseNet	0,9214±0,00	0,8107±0,00	0,8903±0,00	0,7293±0,00
ResNet	0,9052±0,00	0,8573±0,00	0,8918±0,00	0,7400±0,00
LeNet	0,9203±0,00	0,8493±0,01	0,9004±0,00	0,7573±0,01

Em negrito são os melhores resultados.

Nas CNNs, é possível destacar os atributos extraídos com LeNet, que alcançou, em geral, os melhores resultados quando comparado com outras arquiteturas. Isso pode ser explicado pelo menor número de camadas em comparação com DenseNet e ResNet.

A Tabela 2.5 fornece uma comparação de nosso método com métodos de última geração baseados em CNN e extração de atributos. Al-Shabi et al. (2019) foi utilizada uma ResNet modificada, onde extraia atributos locais e não locais. No trabalho de Zhai et al. (2020) utilizou uma arquitetura CNN para classificar e reconstruir o nódulo para definir os atributos destacados pela CNN. Tanto a modificação na DenseNet proposta por Dey, Lu e Hong (2018) e o método proposto por Afshar et al. (2020), que utiliza três imagens do nódulo com escalas diferentes para classificar o nódulo, embora as acurácia de classificação alcançadas fossem promissoras, necessita de um poder computacional maior que os demais métodos.

Trabalhos	Base	Método	Ac	Sens	Espe
Al-Shabi et al. (2019)	LIDC-IDRI	MoCNN	0,8846	0,8866	0,8738
Zhai et al. (2020)	LIDC-IDRI	MT-CNN		0,8774	0,8887
Dey, Lu e Hong (2018)	LIDC-IDRI	MoDenseNet	0,9040	0,9047	0,9030
Afshar et al. (2020)	LIDC-IDRI	3D-MCN	0,9312	0,9494	0,9000
Método proposto	LIDC-IDRI	LBP3D + CNN3D	0,9037	0,9266	0,8453

2.6 Conclusão

Neste trabalho, é proposta uma metodologia utilizando LBP3D como entrada para CNN3Ds aplicada na classificação de nódulos pulmonares entre malignos e benignos. Os resultados mostram que não houve uma melhora significativa nos resultados da CNN aplicando mapas de textura na entrada em relação aos volumes originais. Nossos resultados experimentais demonstram um desempenho decente de classificação no diagnóstico de câncer de pulmão. Desta forma, nosso objetivo foi apresentar outras formas de usar CNNs, que podem ser aplicadas a outros métodos ou problemas.

Para trabalhos futuros, será avaliado o desempenho de outros mapas de textura como *wavelets*, matriz de cocorrência em níveis de cinza e histograma de gradientes orientados para serem usados também em outros problemas de classificação.

Segmentação do Disco Óptico em Imagens Retinoscópicas Utilizando a Rede Neural Convolucional U-Net

Autores: Walef Alves da Silva; Romuere Rodrigues Veloso e Silva.

Resumo: A Retinopatia Diabética (RD) é uma complicação comum e potencialmente prejudicial à visão do diabético e é uma das principais doenças causadoras de cegueira. Um diagnóstico precoce da RD para localizar lesões em seu estado inicial pode fazer com que alguns tratamentos reduzam o risco da perda da visão. O diagnóstico da RD pode se dar por métodos computacionais automatizados através de imagens retinoscópicas como também pode ser feita de maneira manual através da análise de um especialista. Quando o processo de identificação de lesões (microaneurismas, exsudatos duros, moles e disco óptico (DO)) é executado de maneira automática, a análise destas torna-se mais concisa e precisa, auxiliando a elaboração do diagnóstico do estágio da doença. Este trabalho analisa o desempenho da Rede Neural Convolucional (RNC) U-Net baseado no modelo original da arquitetura com o ajuste na quantidade de camadas para melhorar o seu desempenho no problema da segmentação do disco óptico em imagens retinoscópicas contendo ou não retinopatia diabética. Os resultados obtidos utilizando as bases *I-DRID*, *RIM-ONE* e *DRIONS-DB* foram de dice superior a 85% e acurácia superior a 87% mostrando ser uma metodologia promissora para a segmentação do disco óptico.

Keywords: Retinopatia Diabética, Segmentação, U-Net.

3.1 Introdução

A retinopatia diabética (RD) proliferativa é uma complicação comum e potencialmente prejudicial à visão do diabético, indivíduos incapazes de controlar a glicemia tendem a sofrer com a retinopatia (B. et al., 2019).

A RD é uma das principais causas de cegueira. Apesar dos tratamentos disponíveis, alguns pacientes só descobrem a doença tarde, dificultando o tratamento. Se a retinopatia for detectada no início, alguns fatores de risco modificáveis (glicose, pressão arterial) podem retardar a progressão e reduzir o risco da cegueira (SCANLON, 2019).

A detecção da RD se da pela presença de lesões na retina. A segmentação de micro-aneurismas, ex-sudatos duros e moles e do disco óptico é crucial no diagnóstico da RD. No entanto, tamanhos variados, características semelhantes (intensidade e cor), variação da localização e formas tornam este trabalho desafiador. A delineação manual feita por oftalmologistas em imagens retinoscópica é demorada e exaustiva, o que leva a uma alta demanda para desenvolver um método de segmentação automática eficaz e preciso (XUE et al., 2019).

Com base na necessidade e a importância, apontadas acima, da segmentação como processo de análise e diagnóstico da retinopatia diabética em imagens retinoscópicas. Este trabalho emprega a Rede Neural Convolucional (RNC) U-Net no problema da segmentação do disco óptico em imagens que contém ou não a RD. O desempenho deste trabalho foi avaliado utilizando as métricas: Acurácia (PROVOST; DOMINGOS, 2000) e o dice (GINSBERG; YOUNG, 1992)

3.1.1 Objetivos

3.1.1.1 Objetivo Geral

A partir do contexto apresentado, o presente trabalho possui como objetivo principal aplicar uma RNC U-Net para a segmentação do disco óptico em imagens doentes ou saudáveis.

3.1.1.2 Objetivos Específicos

1. Avaliar o desempenho da segmentação utilizando a RNC U-NET.
2. Gerar uma segmentação coerente a máscara disponibilizada pelo especialista.
3. Construir uma arquitetura que forneça ao especialista uma nova proposta na segmentação do disco óptico.
4. Testar o desempenho do método em diferentes bases.

3.1.2 Organização do Trabalho

Além da introdução, este trabalho está dividido na seguinte ordem: no Capítulo 2, apresenta-se o referencial teórico; no Capítulo 3, são apresentados os trabalhos relacionados; no Capítulo 4, descreve-se a metodologia proposta; no Capítulo 5, são descritos os resultados e discussões da execução do projeto; e por fim, são apresentadas as conclusões e sugestões para trabalhos futuro no Capítulo 6.

3.2 Referencial Teórico

Este capítulo apresenta a fundamentação teórica utilizada no desenvolvimento deste trabalho contendo conceitos, conhecimentos e informações pertinentes ao trabalho e necessária para compreensão das técnicas utilizadas para alcançar os objetivos esperados pelo mesmo.

3.2.1 Retinopatia diabética

Segundo a Organização Mundial de Saúde (OMS), a RD é uma doença que pode se desenvolver em pacientes diabéticos que estão em uma faixa etária de idade denominada ativa, casos que ocorrem em pacientes com menos de 20 anos e com mais de 70 anos não são comuns. Esta doença é normalmente progressiva e possui 5 estágios, o primeiro é caracterizado pela ausência da doença em pacientes que possuem diabetes, no segundo estágio o paciente apresenta pequenas lesões acompanhadas de hemorragia, mas de forma discreta. O terceiro estágio é denominado moderado, e já possui diversos tipos de aneurismas e hemorragias. O quarto estágio já está em um nível avançado da doença, neste caso o tratamento já se faz necessário devido a diversas lesões como hemorragia, aneurismas e má formação de vasos vasculares, entre outros. O último estágio da doença já traz complicações maiores e pode levar a cegueira do paciente, desta maneira já existe a necessidade de cirurgias, injeções, etc.

A RD tem como a principal causa de problemas visuais o edema macular, o mesmo pode estar presente desde o início da retinopatia até em estágios mais avançados da doença, atacando 30% dos pacientes com mais de 20 anos de diabetes. O estágio mais avançado da doença é o proliferativo que é aquele que se relaciona mais frequente com a perda visual grave, devido a eventos oculares possivelmente causadores de cegueira irreversível incluindo o edema macular e o descolamento tracional de retina. Estima-se que em olhos com RD proliferativa não tratada a taxa de evolução para cegueira seja de 50%, em 5 anos(Oliveira Renan Magalhães Montenegro Júnior, 2018).

"A RD é uma das principais complicações relacionadas ao diabetes mellitus (DM) é a principal causa de cegueira em pessoas com idade entre 20 e 74 anos. Aproximadamente 12% dos novos casos de cegueira legal, isto é, a diminuição da acuidade visual a um nível que impõe o exercício de atividades laborais, são causados pela RD. Após 20 anos de doença, mais de 90% dos diabéticos tipo 1 e 60% daqueles com o tipo 2 apresentarão algum grau de retinopatia."(Oliveira Renan Magalhães Montenegro Júnior, 2018).

Shu, Chui e FRANZCO (2016) citam em seu trabalho uma estimativa feita pela OMS onde a RD é responsável por cerca de 37 milhões de casos de cegueira registrados em todo o mundo, essa quantidade é equivalente a 4,8% dos casos. Ainda nesse trabalho, foi feita uma análise conjunta de 22.896 pessoas com diabetes que residem nos EUA, Austrália, Europa e Ásia, onde se concluiu que a prevalência geral de qualquer DR é de 34,6%.

Graças a duas importantes séries de estudos é que foi possível se basear para elaborar definições como a de tratamento com fotocoagulação a laser e os critérios necessários para o diagnóstico e caracterização da RD. Os dois estudos em questão são: Diabetic Retinopathy Study (DRS – 1976 a 1979) e Early Treatment Diabetic Retinopathy (ETDRS – 1985 a 1997), o primeiro foi responsável pela definição de critérios diagnósticos para a RD proliferativa, enquanto o segundo definiu critérios utilizados referentes à RD não proliferativa e ao edema macular, assim como a utilização da fotocoagulação a laser nesses pacientes. Buscando menor fragmentação e maior padronização na classificação da RD ETDRS (Fong et al., 2016).

3.2.2 Redes Neurais Convolucionais

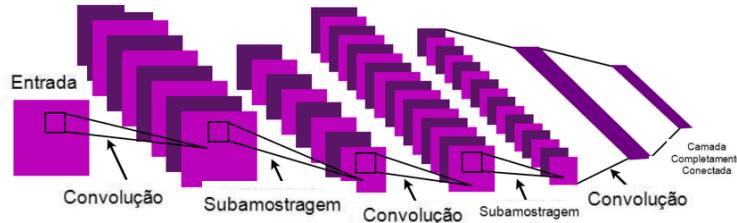
As redes neurais convolucionais (CNN - do inglês *Convolutional Neural Network*) são modelos biologicamente inspirados que podem aprender características de forma hierárquica (Lecun; Bengio; Hinton, 2015). Este tipo de rede neural vem sendo amplamente utilizada em aplicações de segmentação, classificação, detecção e reconhecimento em imagens e vídeos. As mesmas são especialmente projetadas para lidar com a variabilidade em dados bidimensionais (2D), como as imagens em formato matricial. Contudo, as RNCs têm como característica sua versatilidade, e possuem a capacidade de processar em diversos formatos de matrizes: 1D para sinais e sequências (linguagem); 2D para imagens ou espectrogramas de áudio; e 3D para imagens em vídeos ou volumétricas. As RNCs possuem quatro princípios que a aproximam de sinais naturais: conexões locais, pesos compartilhados, agrupamento e o uso de muitas camadas (Almousli, 2014; Lecun; Bengio; Hinton, 2015; Perona; Fink, 2013).

De forma geral uma RNC tem sua arquitetura composta de três tipos de camadas: convolução, subamostragem e completamente conectada (Kang; Wang, 2014). As camadas convolucionais são responsáveis pela extração de características e a camada de subamostragem é responsável pela redução de dimensionalidade dos dados a serem classificados. As características são extraídas de uma camada para outra da rede, onde um neurônio na camada atual conecta-se a um campo receptivo local na camada anterior. Dessa forma, as características são combinadas sequencialmente para, assim, serem extraídas características de mais alto nível (Fernandes, 2013). Essas camadas são seguidas por camadas completamente conectadas, análogas à percepção de multicamada (MLP), responsável por fazer a classificação com base nas características extraídas nas camadas anteriores.

A Figura 25 ilustra a arquitetura de uma RNC. Onde a entrada da rede é normalmente uma matriz tridimensional com altura e largura (de acordo com as dimensões da imagem) e profundidade, determinada pela quantidade de canais de cores. Geralmente as imagens utilizam três canais, RGB, com os valores de cada pixel. As convoluções funcionam como filtros que enxergam pequenos quadrados e vão percorrendo por toda a imagem aprendendo os traços mais marcantes como representado na Figura 26. As funções de

ativação servem para trazer a não-linearidades ao sistema, para que a rede consiga aprender qualquer tipo de funcionalidade. Há muitas funções, como sigmoid, tanh e softmax, mas a mais indicada para redes convolucionais é a Relu por ser mais eficiente computacionalmente sem grandes diferenças de acurácia quando comparada a outras funções. A camada de pooling serve para simplificar a informação da camada anterior. Assim como na convolução. A unidade é responsável por resumir a informação daquela área em um único valor. Ao final da rede é colocada uma camada completamente conectada, onde sua entrada é a saída da camada anterior e sua saída são N neurônios, com N sendo a quantidade de classes do seu modelo para finalizar a classificação.

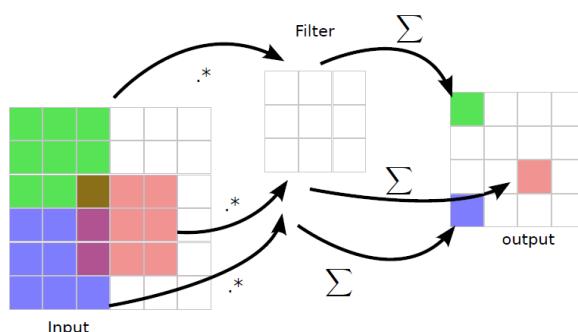
Figura 25 – Representação da Arquitetura das Redes Convolucionais.



Fonte: Adaptado de ([LECUN et al., 2010](#))

As camadas de convolução são compostas por filtros treináveis que são aplicados em toda a imagem de entrada, gerando assim, vários mapas de características ([LECUN et al., 2010](#)). Os filtros determinam uma pequena área (3×3 , 5×5 , 7×7 pixels) e cada neurônio é ligado somente aos neurônios nas proximidades da camada anterior. Os neurônios compartilham os pesos entre si, levando os filtros a aprender os padrões frequentes que ocorrem em qualquer parte da imagem. Posteriormente ao treinamento da rede, os filtros tornam-se responsáveis pela detecção de características particulares que ocorrem na imagem ([HAFEMANN, 2014a](#)). Uma vez que a característica é detectada, sua localização exata na imagem de entrada torna-se irrelevante (??). A Figura 26 ilustra como ocorre a convolução em uma imagem.

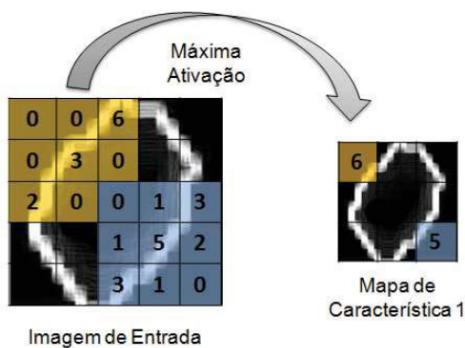
Figura 26 – Representação da Camada de Convolução.



Fonte: ([HAFEMANN, 2014b](#))

De acordo com [Hafemann \(2014a\)](#), as camadas de subamostragem implementam uma função de redução da solução não-linear, a fim de reduzir a dimensionalidade selecionando características invariantes à deslocamentos e distorções. A Figura 27 apresenta o processo da camada de subamostragem do tipo máxima ativação, em que apenas o pixel de maior intensidade do campo receptivo anterior é mantido. Existem outras formas de ativação na camada de subamostragem, como por exemplo: média, mediana, mínima e entre outras.

Figura 27 – Representação da Camada de Subamostragem.



Fonte: ([SILVA et al., 2017](#))

Após as operações realizadas nas camadas de convolução e subamostragem que são encarregadas de extrair atributos da imagem, os *pixels* de todos os mapas de características da camada anterior são dados como entrada às camadas completamente conectadas, que por sua vez, são responsáveis pela classificação dos padrões de entrada, semelhante à MLP.

Para o problema de classificação de imagens, as arquiteturas baseadas no modelo mostrado na [Figura 25](#) determinam o estado da arte. [Long, Shelhamer e Darrell \(2015\)](#) propôs uma arquitetura sem camadas completamente conectadas: a rede completamente convolucional (FCN do inglês *Fully convolutional networks*). Esta rede gera uma saída nas mesmas dimensões da imagem de entrada, realizando uma segmentação *pixel-a-pixel* da imagem de entrada.

3.2.2.1 Rede Completamente Convolucional

Uma Rede completamente convolucional(RNC) usa uma rede neural convolucional para realizar a transformação dos pixels de imagem em categorias de pixel. Diferentemente das redes neurais convolucionais citada na Subseção [3.2.2](#), uma FCN converte a altura e a largura do mapa de recursos da camada intermediária de volta ao tamanho da imagem de entrada através da camada de uma convolução transposta, de modo que as previsões tenham uma correspondência individual com a imagem de entrada em dimensão espacial (altura e largura). Dada uma posição na dimensão espacial, a saída da dimensão do canal será uma previsão de categoria do pixel correspondente ao local ([Long; Shelhamer; Darrell, 2015](#)).

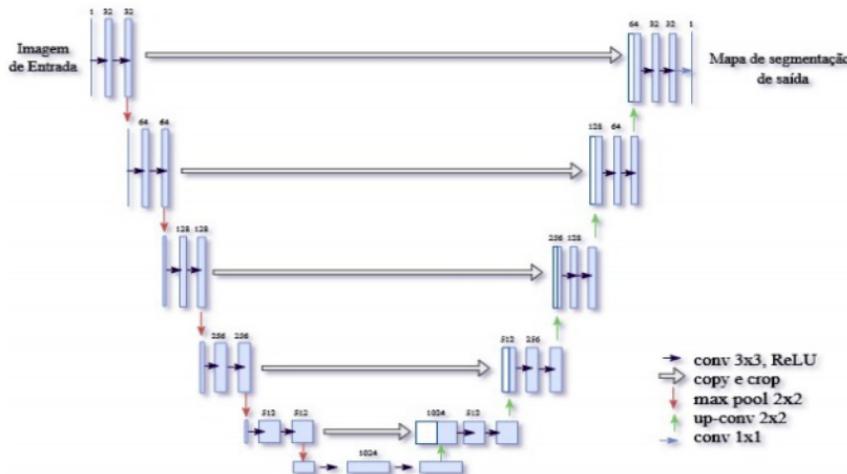
Uma FCN indica que a rede neural é composta de camadas convolucionais sem nenhuma camada totalmente conectada ou MLP normalmente encontrada no final da rede, elas possuem em sua estrutura apenas convoluções e suas funções de ativação (sigmoid, relu e etc). As redes convolucionais, são treinadas de ponta a ponta, pixels a pixels, que tenta regenerar tanto a entrada quanto a saída, verifica-se que camadas convolucionais em diferentes níveis caracterizam o contexto sob diferentes perspectivas. Uma camada superior realiza a propagação de informações, enquanto uma camada inferior carrega as informações e pode separar melhor os pixels com aparência semelhante. Ambas as camadas são usadas em conjunto para realizar a segmentação as FCNs são comumente utilizadas como auto-codificadores e também para tarefas supervisionadas, como classificação e segmentação. Dentre as arquiteturas FCNs famosas podemos ressaltar as utilizadas para segmentação como a UNet e a ResNet ([WANG et al., 2015](#)).

3.2.3 U-Net

[Ronneberger, Fischer e Brox \(2015\)](#) apresentaram a U-Net, que é uma RNA (Rede Neural Artificial) completamente convolucional. A mesma teve em sua concepção, o objetivo de realizar a segmentação de imagens biomédicas. A U-Net é uma rede totalmente convolucional, cuja a estrutura consiste em um caminho de contração para registrar o contexto, e um caminho de expansão simétrico que possibilita uma segmentação com exatidão.

A U-Net distingui-se das demais arquiteturas FCN (Rede Completamente Convolucional) pela junção entre os mapas de características do estágio de contração e seus equivalentes simétricos no estágio de expansão, permitindo a propagação de informações de contexto para os mapas de atributos de alta resolução. Em diversos desafios biomédicos de segmentação de imagens o presente modelo vem se destacando entre os mais eficientes métodos existentes (RONNEBERGER; FISCHER; BROX, 2015). A Figura 28 ilustra a arquitetura e as camadas que constituem a rede.

Figura 28 – Representação da Arquitetura U-Net.



Fonte: (RONNEBERGER; FISCHER; BROX, 2015)

A arquitetura consiste em um caminho de contração (lado esquerdo) para capturar o contexto e um caminho de expansão (lado direito) proporcional que permite uma localização precisa. O caminho de contração é uma arquitetura comum de uma CNN, utilizando de inúmeras aplicações de duas convoluções 3×3 , cada uma acompanhada por uma função de ativação Unidade Linear Retificada (ReLU) e uma operação de subamostragem com ativação máxima de 2×2 reduzindo a dimensão da imagem, com passo 2 para subamostragem. Em cada uma dessas etapas, o número de canais de características é duplicado.

O caminho expansivo baseia-se em um levantamento do mapa de características seguido de um sobre-amostragem 2×2 que faz a metade do número de canais de características, que é uma união com o mapa de características correspondentemente cortado do caminho de contratação e duas convoluções 3×3 , cada uma seguida por uma função de ativação ReLU. Essa etapa do caminho expansivo é importante devido à perda de *pixels* de borda em cada convolução no caminho de contratação. Na última camada é utilizada uma convolução 1×1 para mapear cada vetor de características para o número desejado de classes.

Por último, essa rede é capaz de ser treinada do início ao fim a partir de poucas imagens, onde a U-Net simplesmente correlaciona os mapas de características do codificador para mapear mapas de características do decodificador em todas as etapas para gerar uma estrutura como escada. Essa arquitetura por suas conexões de concatenação permite que o decodificador em cada etapa aprenda as características pertinentes que são perdidas quando agrupadas no codificador. No presente trabalho, a U-Net foi empregada para gerar a segmentação da região do Disco Óptico em imagens de RD.

3.3 Trabalhos Relacionados

Neste capítulo expomos trabalhos que abordam problemas semelhantes ao nosso, segmentação do disco óptico em imagens retinoscópicas que contem ou não RD. A Tabela 10 apresenta um resumo dos trabalhos relacionados.

Tabela 10 – Resumo dos trabalhos relacionados.

Trabalho	Método	Dataset	Amostras	Acurácia
(XUE et al., 2019)	Segmentação Automática	IDRID	81	-
(WEN-BO et al., 2018)	CNN	DRIONS	100	98,90%
(KOWSALYA et al., 2018)	Kapur e K-means	RIM-ONE	169	-
(XUE et al., 2017)	Saliência baseada em agrupamento	MESSIDOR	1200	94,00%
Método proposto	U-Net	IDRID	54	99,52%

Xue *et al.* (2019) em seu trabalho propõe um novo método automático de segmentação de microaneurismas, exsudatos duros e DO para a avaliação diagnóstica da RD. Este método de segmentação multitarefa é baseado num novo sistema de membrana denominada de sistema de membrana dinâmica com estruturas híbridas, os resultados obtidos para a segmentação do DO foram: 96,7% para o dice e 98,1% para sensitividade.

Wen-Bo *et al.* (2018) propôs um método de detecção de DO baseado em Convolutional Neural Network (CNN) e o Campo Aleatório Condicional (CAC). As imagens retinoscópicas foram pré-classificadas pela CNN e uma função de potência de primeira ordem foi construída do CAC. Em seguida, a combinação linear das funções do kernel gaussiano foram usadas para construir a função de segunda ordem CAC. Ao fim, aplicou-se o método de restrições regionais para à análise da consistência dos rótulos e corrigir o rótulo de cada pixel calculando a média de probabilidade posterior de super *pixels*, onde foi obtido 98,90% de acurácia.

Kowsalsya *et al.* (2018) em seu trabalho, implementou uma técnica de exame para extraer o DO para avaliar sua condição. Para minerar a região do DO a partir de imagens RGB da retina foi aplicado uma abordagem baseada na combinação da entropia de Kapur e agrupamento K-means. Após a extração do DO, análises foram realizadas comparando com as verificações de fundo do especialista, os resultados obtidos foram de 93,93% de dice e 96,73% de sensibilidade.

Xue *et al.* (2017) em seu estudo, desenvolveu um novo método de detecção do disco óptico usando modelo de saliência baseado em agrupamento, para simular o mecanismo de filtragem humana de sistema visual para detecção do DO de imagens do fundo do olho. Primeiro, candidatos a DO são extraídos com K-means. Segundo, duas saliências da imagem são selecionadas como a região DO. Terceiro, o contorno do DO original é extraído por elipse. Último, o contorno do DO pode ser segmentado com precisão por contorno ativo, o resultado obtido foi de 94% de acurácia.

A abordagem aqui proposta utiliza de uma rede específica para segmentação de imagens biomédicas que pode ser treinada de ponta a ponto com um conjunto pequeno, por isso o tempo de execução é diminuído onde todos os testes realizados foram rodados no Collab. É importante ressaltar que alguns dos trabalhos relacionados realizam a segmentação do disco óptico e também de microaneurismas.

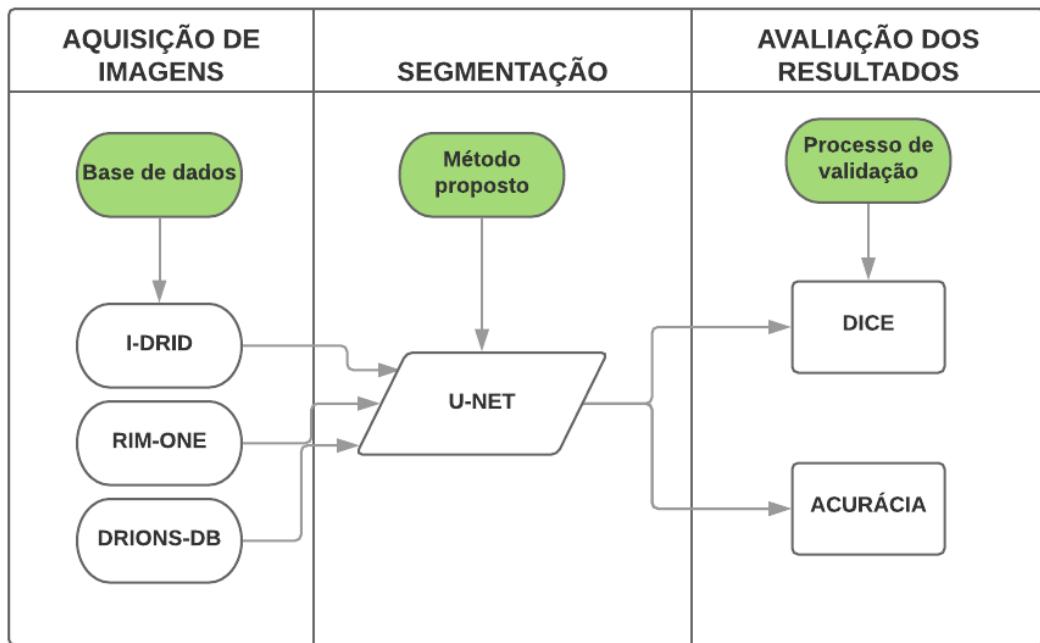
3.4 Materiais e Métodos

Esta seção apresenta os métodos propostos para segmentação do disco óptico, a partir de imagens médicas do fundo do olho. Esse método é composto por três etapas, como pode ser observado na Figura 29. Em síntese, a primeira etapa descreve os materiais utilizados, como as bases de imagens de RD. Na segunda etapa é realizado a segmentação. Na terceira e última etapa, o resultado da segmentação é avaliado.

3.4.1 Base de Imagens

Foram 3 os conjuntos de dados utilizados neste trabalho, onde as imagens foram obtidas através da retinografia digital. No exame de retinografia ocular as imagens obtidas estão as cores se encontram em alta resolução, permitindo desta forma ver com detalhe todo o fundo ocular (retina), as imagens são capturadas pelo retinógrafo, por mais que sejam capturadas pelo mesmo exame possuem dimensões diferentes para cada base.

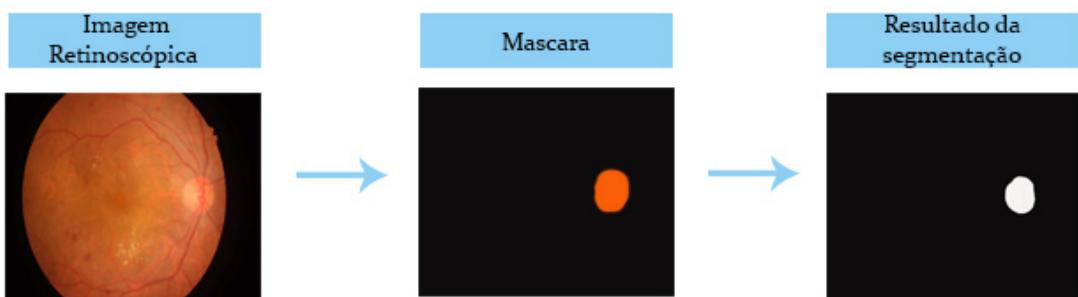
Figura 29 – Etapas do método proposto.



3.4.1.1 I-DRID

O dataset IDRID disponível como parte do desafio “IDRID - Diabetic Retinopathy”, organizado em conjunto com o IEEE International Symposium on Biomedical Imaging ([IEEE, 2018](#)). O dataset além das imagens fornece a máscara binária para avaliação e desempenho técnico de lesões e do disco óptico como pode ser observado na Figura 29. Este conjunto de imagens consiste em 81 imagens de fundo de cor com sinais de RD e 164 sem sinais de RD ([PORWAL et al., 2018](#)), porém apenas 54 imagens possuem marcação do disco óptico feita por especialistas e elas possuem dimensões de 4288x2848 pixels.

Figura 30 – Representação do resultado da segmentação com a base I-DRID.

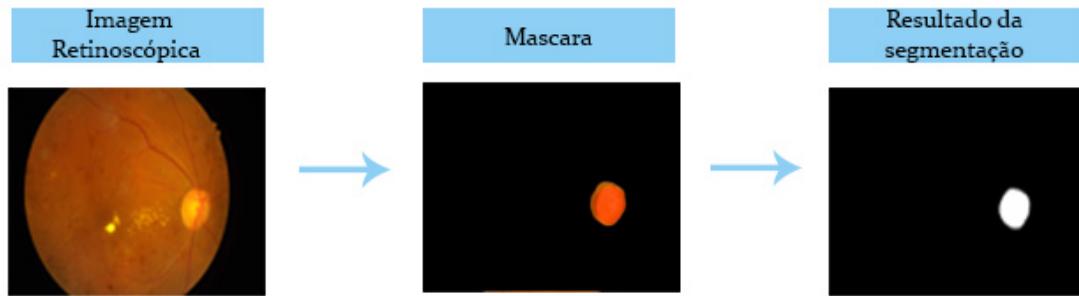


3.4.1.2 RIM-ONE

A RIM-ONE é uma base pública que possui um conjunto de 50 imagens biomédicas de fundo do olho com suas referentes máscaras binárias ([FUMERO et al., 2011a](#)). As imagens possuem dimensões de 600x400

pixels. A Figura 31 mostra exemplo dessa base de imagens.

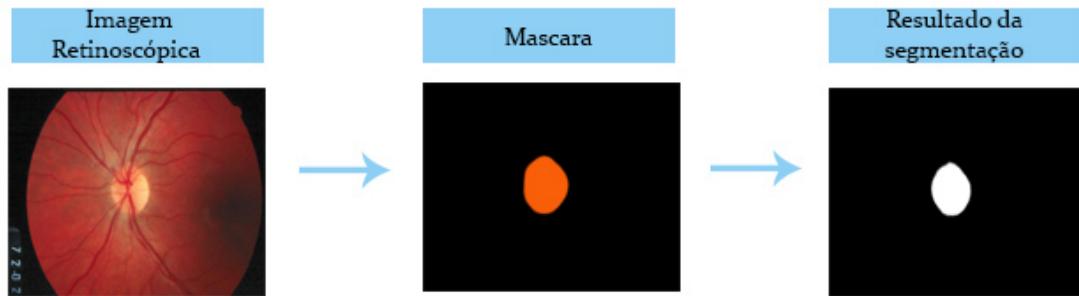
Figura 31 – Representação do resultado da segmentação com a base RIM-ONE.



3.4.1.3 DRIONS-DB

DRIONS-DB possui em seus dados um conjunto de 37 imagens e suas respectivas máscaras ([FEIJOO et al., 2014](#)). As imagens possuem dimensões de 530x462 pixels. A Figura 32 mostra exemplo dessa base de imagens.

Figura 32 – Representação do resultado da segmentação com a base DRIONS-DB.



3.4.2 Segmentação

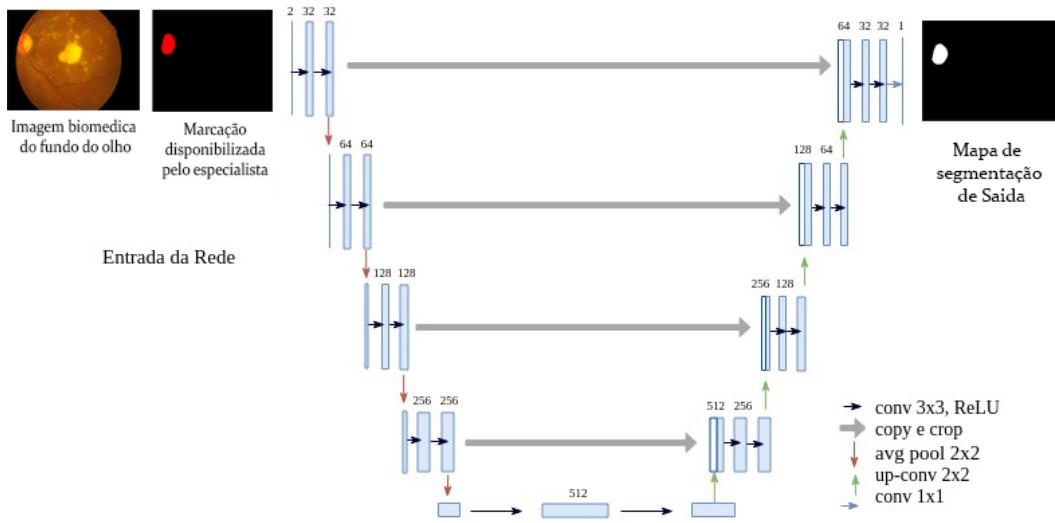
A segmentação do Disco Óptico é necessária pois em alguns estágios a doença pode apresentar uma lesão que se assemelha com o mesmo dificultando assim o diagnóstico. Esse trabalho apresenta um método para segmentação do DO, utilizando uma rede convolucional baseada na arquitetura U-Net. O método possui a entrada composta pela imagem de RD e pelas marcações dos especialistas dos exames de treino.

3.4.2.1 U-Net

A estratégia utilizada para a segmentação é um modelo de aprendizagem profunda baseado na U-Net proposta por [Ronneberger, Fischer e Brox \(2015\)](#) descrita na Subseção 3.2.3. A segmentação do disco óptico é realizada em cada pixel da imagem, sendo assim uma segmentação 2D. A entrada da rede consiste em uma imagem do fundo do olho acompanhada pela marcação específica do especialista.

A rede utilizada para segmentação difere da U-Net original na quantidade de camadas. No método proposto a rede possui apenas quatro camadas de subamostragem com função de ativação do tipo média, onde a média de intensidade dos pixels da janela é mantida. Para o caminho de expansão é utilizada a

Figura 33 – U-Net arquitetura. Cada caixa azul corresponde a um mapa de recursos multicanal. O número de canais é indicado na parte superior da caixa. O tamanho X-Y é fornecido no canto inferior esquerdo da caixa, caixas brancas representam mapas de recursos copiados. As setas denotam as diferentes operações (RONNEBERGER PHILIPP FISCHER, 2015).



convolução transposta. Cada convolução é seguida por uma camada de normalização de *batch* e uma camada de *Dropout*, com uma taxa de 20% para desativação de neurônios. Por fim, a segmentação é feita por meio de uma convolução (1 x 1), com função de ativação sigmóide. A rede é treinada com a função de perda *Dice Loss*, representada pela Equação 3.1.

$$S = 1 - \frac{2VP}{2VP + FP + FN}. \quad (3.1)$$

3.4.3 Métricas de desempenho

As métricas utilizadas para validação dos resultados foram acurácia (PROVOST; DOMINGOS, 2000) e o dice (GINSBERG; YOUNG, 1992).

O índice Dice é bastante aplicado na literatura como medida para desempenho da segmentação e indica a proporção de *pixels* compartilhadas entre duas amostras em relação ao total de *pixels*.

Já acurácia, calcula o total de acertos baseado em todas as instâncias segmentadas, calculado a partir de todos os casos (segmentados corretamente e incorretamente) (PROVOST; DOMINGOS, 2000).

A equação Dice (DSC) é dada pela dobrado de verdadeiros positivos (VP) comuns em duas amostras dividido pela soma do numero total de amostra somado aos falsos positivos (FV) e falsos negativos (FN), Equação Equação 3.2, onde a faixa de valores compreendidos por esta medida vão de 0% (nenhuma correspondência entre os volumes) a 100% (total correspondência entre os volumes).

$$DSC = \frac{2VP}{2VP + FP + FN}. \quad (3.2)$$

A equação da Acurácia (A) é expressa através da razão do acerto total (dado por VP + VN) pelo total de casos da base. A equação para calculo da acurácia pode ser observado na Equação 3.3.

$$A = \frac{VP + VN}{VP + VN + FP + FN}. \quad (3.3)$$

3.4.3.1 Validação

Para que seja realizada a análise dos resultados obtidos pelo presente trabalho, foram definidas algumas métricas para a validação da eficiência do mesmo. Este trabalho utiliza métricas comumente empregadas

na literatura para análise de desempenho de sistemas baseados em processamento de imagens e reconhecimento de padrões para segmentação de imagens. Estas métricas são Acurácia (A), Dice (Subseção 3.4.3). Tais métricas têm como objetivo medir o desempenho do modelo proposto como aceitável ou não, além de auxiliar na identificação de pontos positivos e negativos para melhoria futura deste trabalho nas fases de treinamento, validação e teste.

3.5 Resultados e Discussões

Esta seção apresenta e discute os resultados obtidos pela metodologia proposta para segmentação do disco óptico em imagens retinoscópicas. Todos os testes foram implementados na linguagem de programação Python com o auxílio da biblioteca de aprendizagem profunda Keras ([CHOLLET et al., 2018](#)),

Para a realização destes experimentos, foi utilizada a plataforma Google Colaboratory ([CARNEIRO et al., 2018](#)). A mesma trata-se de um serviço baseado em *cloud* que utiliza Jupyter Notebooks para disseminar a pesquisa em aprendizado de máquina. No Colab, como também é conhecido, é fornecido um ambiente totalmente configurado e pronto para ser utilizado, com acesso gratuito a uma GPU robusta.

Para os testes da metodologia foram utilizadas todas as 54 imagens da base *I-DRID*, 50 imagens da base *RIM-ONE* e 37 imagens da base *DRIONS-DB*. As bases foram divididas aleatoriamente em 3 partes: treinamento, validação e teste. Da base *I-DRID* foram utilizadas 37 imagens para treino, 8 para validação e 9 para teste, a base *RIM-ONE* foram utilizadas 35 imagens para treino, 7 para validação e 8 para teste e a base *DRIONS-DB* foram utilizadas 24 imagens para treino, 6 para validação e 7 para teste.

Inicialmente foi realizada a quantização não-uniforme dos exames da base. Onde os valores de intensidade dos *voxels* são normalizados entre 0 e 255, tornando a região do disco óptico mais uniforme e realizando uma representação da imagem com uma quantidade menor de bits.

Logo após, se inicia a etapa de segmentação. Essa etapa foi executada separadamente para cada base e em seguida, executada juntando as bases de duas em duas em uma única base e fazendo variações de treino e teste.

Tabela 11 – Resultados obtidos no método proposto com a U-Net treinada e testada apenas como uma única base.

Base de dados	Acurácia	Dice
<i>I-DRID</i>	99,52%	96,38%
<i>RIM-ONE</i>	98,90%	91,29%
<i>DRIONS-DB</i>	97,10%	81,53%

Diante dos resultados exibidos na Tabela 11 podemos observar resultados satisfatórios, na base *I-DRID* e na base *RIM-ONE* apresentam resultados semelhantes podemos atribuir isso porque as mesmas possuem quantidades semelhantes de imagens, já a base da *DRIONS-DB* apresenta pouca discrepância na acurácia porem o Dice apresenta resultados não tão apropriados, podemos atribuir isso ao menor numero de imagens existentes na base.

Tabela 12 – Resultados obtidos no método proposto com a U-Net com variações na base de teste.

Base de treinamento	Base de teste	Acurácia	Dice
<i>I-DRID</i>	<i>DRIONS-DB+</i> <i>RIM-ONE</i>	87,50%	83,01%
<i>RIM-ONE</i>	<i>DRIONS-DB+</i> <i>I-DRID</i>	88,15%	85,70%
<i>DRIONS-DB</i>	<i>I-DDRID+</i> <i>RIM-ONE</i>	85,40%	86,00%

Diante dos resultados exibidos na Tabela 12 podemos observar a variação na base de teste, onde as demais bases foram agrupadas de duas em duas. Com isso o conjunto de teste ficou bem maior que o de

treino onde causou interferência nos resultados mostrando uma queda nos mesmos.

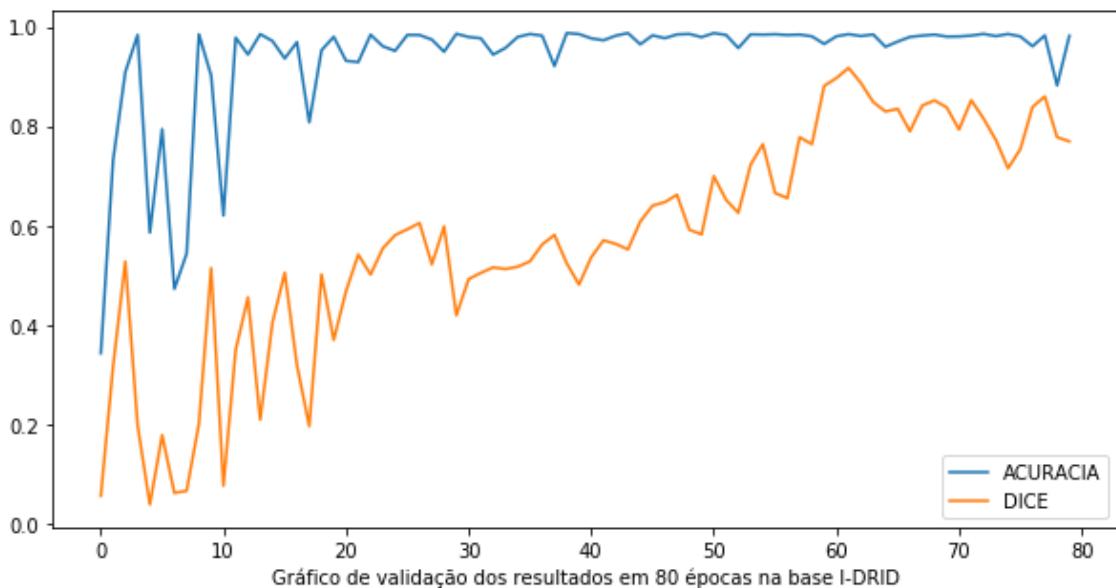
Tabela 13 – Resultados obtidos no método proposto com a U-Net com variações na base de treino.

Base de treinamento	Base de teste	Acurácia	Dice
DRIONS-DB+ I-DRID	RIM-ONE	98,54%	97,23%
DRIONS-DB+ RIM-ONE	I-DRID	99,54%	95,39%
I-DRID+ RIM-ONE	DRIONS-DB	98,85%	97,43%

Com a variação exibida na Tabela 13 podemos observar que com um conjunto de treinamento maior interfere significativamente nos resultados da rede, mostrando uma melhora nos mesmos.

A Imagem 34 mostra a validação dos resultados na base I-DRID ao longo de 80 épocas executada na plataforma Google Colaboratory.

Figura 34 – Gráfico do resultado da validação



A comparação com outros trabalhos dentro da literatura é uma tarefa um pouco complicada, já que há algumas diferenças nas maneiras de abordagem do problema e também nas ferramentas e técnicas utilizadas para solucioná-los. A Tabela 14 exibe um estudo comparativo entre o trabalho proposto e outros quatro da literatura exibindo o método utilizado, dataset e quantidade de amostras, onde podemos observar que os resultados obtidos pela metodologia proposta são semelhantes e por isso podemos considerá-los satisfatórios.

3.6 Conclusão e Trabalhos Futuros

A retinopatia diabética segundo a OMS (Organização Mundial de Saúde) está entre as cinco doenças que mais causam cegueira. A detecção precoce da retinopatia é uma estratégia para encontrar microaneurismas, assim, possibilitar melhor chance de tratamento. Para uma detecção precisa, é necessário imagens retinoscópicas do fundo do olho. Isso evidencia a importância do desenvolvimento de pesquisas com o objetivo

Tabela 14 – Comparação do método proposto com alguns trabalhos relacionados.

Trabalho	Método	Dataset	Amostras	Sensibilidade	Acurácia	Dice
(XUE et al., 2019)	Segmentação Automática	IDRID	81	98,1%	-	96,7%
(WEN-BO et al., 2018)	CNN	DRIONS	100	-	98,90%	-
(KOWSALYA et al., 2018)	Kapur e K-means	RIM-ONE	169	96,73%	-	93,93%
(XUE et al., 2017)	Saliência baseada em agrupamento	MESSIDOR	1200	-	94,00%	-
Método proposto	U-Net	IDRID	54	-	99,52%	96,38%

de segmentar automaticamente o disco óptico, pois a doença em estágio avançado pode apresentar lesão semelhante ao disco óptico dificultando assim a segmentação do mesmo.

Desta forma, este trabalho propõe um método para segmentação do disco óptico em exames de RD, baseados em técnicas de aprendizado profundo. O método utiliza uma rede completamente convolucional com arquitetura baseada na U-Net.

Os resultados obtidos através de experimentos utilizando as bases de imagens públicas de exames de RD do olho, *I-IDRID*, *RIM-ONE* e *DRION-DB*, mostram quanto promissor é o método proposto nesse trabalho. O método produz resultados com o Dice acima de 85% utilizando as três bases, comparáveis aos trabalhos existentes na literatura.

O objetivo da metodologia não é substituir a função do especialista, e sim oferecer a ele uma segunda opinião, corroborando para o aumento da produtividade. Dessa forma, tornando mais ágil e menos exaustiva a análise de exames pelos especialistas. Os resultados apresentados mostraram-se bastante promissores por apresentarem valores satisfatórios para as métricas de avaliação.

Como trabalhos futuros, destacamos:

- Realizar novos testes com uma quantidade maior de imagens / outros datasets;
- Realizar o ajuste fino para obter melhores resultados;
- Realizar testes com a segmentação de microaneurismas.

Determinação automática do glaucoma utilizando índices de diversidade taxonômica

Autores: Antônio Oseas de Carvalho Filho; Tomaz Ribeiro Viana Bisneto.

Resumo: O glaucoma pode ser definido como uma doença neurodegenerativa, que possui uma etiologia multifatorial, que ocasiona na lesão progressiva do nervo óptico. Por ser uma doença crônica e causar cegueira permanente, caso não tratada, seu diagnóstico precoce é necessário, garantindo melhor qualidade de vida e um prognóstico mais favorável para o paciente. O Processamento Digital de Imagens em conjunto com técnicas computacionais de aprendizado de máquina possibilitam a criação de métodos para detecção automática do glaucoma. Assim, esse trabalho apresenta a proposta que busca a criação de uma abordagem para viabilizar o diagnóstico do glaucoma, com o uso de características de textura a partir de índices de diversidade taxonômica. Onde, os resultados indicam que o método é promissor, alcançando um resultado de 98,7% de acurácia com o classificador *Random Forest*.

Keywords: Glaucoma, Image Processing, Taxonomic Diversity Indexes, Pattern Recognition.

4.1 Introdução

O glaucoma pode ser definido como uma doença neurodegenerativa, possuindo uma etiologia multifatorial, que ocasiona a lesão progressiva do nervo óptico, com perda de campo visual correspondente (MELLO; SUSANNA; ALMEIDA, 2016). Por ser uma doença crônica e assintomática, a princípio, é necessário seu diagnóstico precoce para que assim sejam aplicados os procedimentos para amenização de prováveis complicações, evitando a perda parcial ou total da visão.

Mesmo as causas da cegueira variando de acordo com as questões socioeconômicas e geográficas, o glaucoma, independente da população e nível social, se mantém como uma das principais causas de cegueira irreversível (QUIGLEY; BROMAN, 2006). Como visto em Kingman (2004a), o glaucoma é a segunda maior causa de cegueira no mundo, está atrás apenas da catarata. Contudo, a perda de visão causada pela catarata pode ser reversível através de cirurgia, enquanto que o glaucoma causa cegueira permanente.

Mediante o quadro clínico apresentado pelo o glaucoma, a detecção nos estágios iniciais aumenta a possibilidade de obter um melhor tratamento. Quando o glaucoma é identificado na fase inicial facilita-se o

tratamento por não apresentar grande relevância das lesões, favorecendo ao profissional responsável no tratamento e ao paciente diminuindo custo terapêutico (JUNIOR et al., 2018a).

Como visto em Nunes (2006), os sistemas de diagnóstico auxiliado por computador (ou simplesmente sistemas CAD, de “computer-aided diagnosis”) são sistemas computacionais, muitas vezes acoplados a equipamentos médicos, com a finalidade de auxiliar na tomada de decisão a respeito de um diagnóstico. Aplicações com finalidades diversas têm sido desenvolvidas por vários grupos de pesquisas, visando auxiliar na composição de diagnósticos como uma forma de contribuir para a detecção precoce de doenças.

Com intuito de aumentar a precisão na detecção de lesões, a utilização de técnicas de Visão Computacional e Reconhecimento de Padrões, tem se mostrado efetivas e apresentam bons resultados (LEDERLIN et al., 2013a; FILHO et al., 2017a; MUNIA et al., 2017). O desenvolvimento de métodos automáticos para detecção e diagnóstico de doenças estão sendo amplamente exploradas. O que possibilita a detecção de anormalidades em possíveis regiões suspeitas, resultando no auxílio ao diagnóstico de forma antecipada, sendo esta, uma segunda opinião para os especialistas.

Os sistemas CAD geralmente se utilizam das informações extraídas de imagens médicas, que podem ser provenientes de diversos tipos de modalidades de exames. Comumente esses sistemas fazem uso de um conjunto de técnicas computacionais, tais como inteligência artificial, técnicas de extração de características e reconhecimento de padrões, que possibilitam extrair e analisar informações úteis ao diagnóstico de imagens clínicas.

4.1.1 Objetivos

Em vista do contexto apresentado, este trabalho tem como objetivo desenvolver uma abordagem que viabiliza o diagnóstico automático do glaucoma, utilizando Índice de Diversidade Taxonômico para caracterização da textura da imagem da retina. Realizando a extração de características em imagens do Disco Óptico (DO), tendo como base a diversidade taxonômica dos pixels, a fim de identificar padrões nas texturas, tornando possível a quantificação do glaucoma.

Para a concretização do objetivo geral, tem-se os seguintes objetivos específicos:

1. Desenvolver e adaptar técnicas para caracterizar propriedades de textura;
2. Classificar as regiões de interesse extraídas em glaucoma e normal
3. Avaliar a viabilidade do uso de Índices de diversidade taxonômicos como forma de extração de textura em imagens do DO; e,
4. Construir uma metodologia capaz de sugerir ao especialista uma segunda opinião na detecção e no diagnóstico do Glaucoma.

4.1.2 Organização do capítulo

Este trabalho está organizado em seis seções. A Seção 2 apresenta os fundamentos teóricos para o entendimento da metodologia abordada no trabalho. Na Seção 3 são detalhados os principais trabalhos relacionados. Na Seção 4 são abordadas as etapas de desenvolvimento da pesquisa, iniciando pela aquisição das imagens, extração de características de textura utilizando os índices de diversidade taxonômica, seguida pela classificação e finalizando com a validação da metodologia proposta. O Seção 5 discute os resultados obtidos com a metodologia proposta. E por fim, na Seção 6 são apresentadas as conclusões e trabalhos futuros.

4.2 Referencial Teórico

De modo a obter entendimento claro da metodologia e resultados, aqui apresentados, faz-se necessário conhecer alguns conceitos que darão suporte ao entendimento do tema. Nesta seção objetiva-se o esclarecimento da teoria para o entendimento deste trabalho, fornecendo o embasamento teórico necessário para a compreensão do mesmo. Desta forma, serão abordados os seguintes tópicos: Glaucoma, processamento

de imagens, índices de diversidade taxonômica, classificação e reconhecimento de padrões e Validação dos Resultados.

4.2.1 Glaucoma

O glaucoma é classificado como neuropatia óptica crônica, progressiva, caracterizada por alterações do DO e da camada de fibras nervosas da retina. Possuindo como anatomia patológica o surgimento de lesões progressivas dos nervos ópticos, compreendendo um conjunto de afecções oculares, ocasionando a perda, irreversível, total ou parcial da visão (MELLO; SUSANNA; ALMEIDA, 2016).

Dentre os fatores de desenvolvimento do glaucoma, a elevação da Pressão Intraocular (PIO) é a mais comum e principal causa da doença. Esta elevação acontece devido a obstrução do canal de drenagem do líquido humor aquoso, resultando em inúmeras afecções oculares em células da retina e do nervo óptico (MELLO; SUSANNA; ALMEIDA, 2016). A fim de detectar essas anomalias causadas pelo glaucoma, são utilizados exames de avaliação de imagens, tais como a Tomografia de Coerência Óptica, Tomografia Retiniana Heidelberg ou por meio de métodos computacionais com análises de imagens da retina.

Por ser assintomático, a princípio, e de etiologia multifatorial o glaucoma acaba apresentando certo grau de dificuldade para sua detecção. Esse cenário contribuiu para o glaucoma se tornar a segunda maior causa de cegueira mundial, como visto em Nugroho et al. (2016), atingindo o número de 60 milhões de casos no ano de 2010, e com estimativas de alcançar 80 milhões de casos até o ano de 2020.

O processo de detecção do glaucoma por meio de exames oftalmológicos se dá principalmente pela avaliação do aumento do PIO. No entanto, o glaucoma pode ocorrer sem aumento da pressão ocular. Assim, outra forma de detecção, é pela análise de danos ao nervo óptico, sendo necessário um especialista treinado, demandando um alto custo, além de ser um processo demorado (ZILLY; BUHMANN; MAHAPATRA, 2017).

Segundo Foster et al. (2002) a principal característica que diferencia o glaucoma de outras causas de morbidade visual e do estado normal da retina é um padrão característico de dano à cabeça do nervo óptico. Assim, é possível a identificação do glaucoma analisando mudanças de formas, cores ou profundidade em imagens do DO, é possível analisar essas alterações comparando as Figuras 35 e 36. Nesse cenário, a utilização de sistemas CAD auxiliam a detecção precoce de lesões, contrímbuindo com tratamento e garantindo um prognóstico mais favorável para o paciente.



Figura 35 – Imagem de retina sem glaucoma

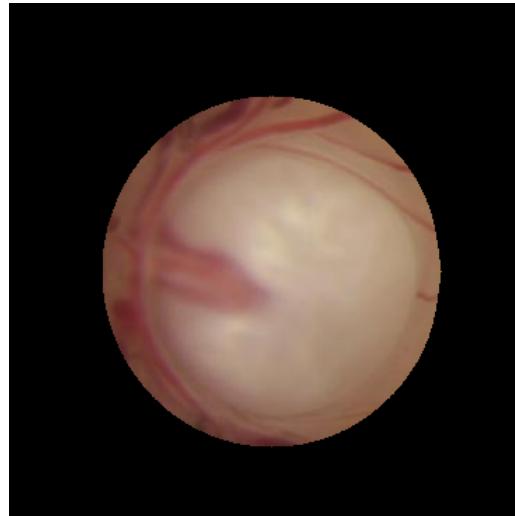


Figura 36 – Imagem de retina com glaucoma

4.2.2 Processamento de Imagens

A história do processamento de imagens digitais como ciência data de 1964, quando durante a corrida espacial, imagens fornecida por sondas espaciais eram processadas por computador para corrigir vários tipos de distorção inerentes as câmeras utilizadas ([FILHO; NETO, 1999](#)). Desde o ano de 1964 aos dias atuais, a área de processamento de imagens vem crescendo expressivamente e o contexto de suas aplicações permeiam quase todos os ramos da atividade humana.

Segundo [Foster et al. \(2002\)](#), o processamento de imagens digitais compreende processos cujas entradas e saídas são imagens e, além disso, engloba os processos de extração de atributos a partir de imagens, incluindo o reconhecimento de objetos individuais. Tendo como objetivos, a melhora da informação visual para interpretação humana e representação dos dados para percepção automática através de máquinas.

De maneira geral, como visto na bibliografia ([FOSTER et al., 2002](#); [UMBAUGH, 1997](#); [FILHO; NETO, 1999](#)), o processamento de imagens segue um conjunto de 5 passos principais utilizados para demonstrar as etapas do ciclo de vida do processamento de imagem (Figura37). Essas etapas consistem em: aquisição de imagens, pré-processamento, segmentação, representação e descrição, reconhecimento e interpretação.

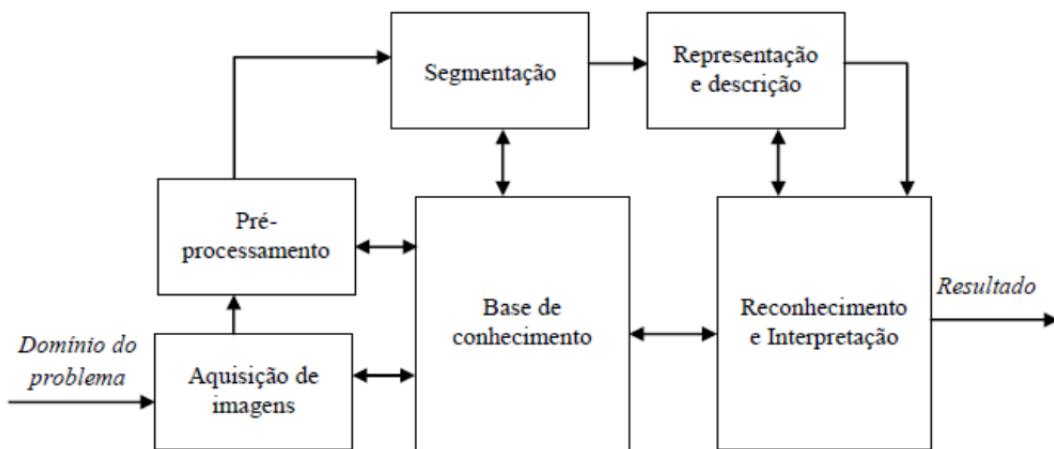


Figura 37 – Etapas do processamento digital de imagens. Adaptado de([GONZALEZ; WOODS, 2002](#)).

Ao final de cada uma das etapas do processamento de imagens os resultados gerados funcionam como entrada para a próxima etapa, dessa forma cada um dos passos irá resultar em informações para alimentar a etapa seguinte. Sendo que ao fim de todas as etapas, o resultado pode ser ou não representado por uma imagem digital.

A primeira etapa desse processo é a aquisição de imagens, onde um digitalizador converte a imagem analógica para digital. Assim, as imagens são capturadas e representadas de forma computacional para serem interpretadas em etapas posteriores.

O pré-processamento consiste na segunda etapa do processo. Essa etapa busca a melhoria dos aspectos visuais e estruturais da imagem, objetivando aumentar as chances de sucesso dos processos seguintes. Nessa etapa podem ser aplicadas técnicas de realce e melhoramento de imagem, por exemplo: remoção de ruído, filtros morfológicos, dentre outras.

A segmentação, por sua vez, trata de extrair da imagem apenas partes que realmente interessam para o processamento, dividir a imagem em regiões disjuntas, permitindo o isolamento de regiões de interesses (ROI). A segmentação é o processo de particionamento de regiões onde todos os elementos de uma região devem ser o mais homogêneo possível, enquanto regiões diferentes devem ser mais heterogêneo ([SOUZA et al., 2011](#)). O processo de segmentação pode ser abordado de três formas: manual, semi-automática e automática.

A quarta etapa é a representação e descrição, também conhecida como extração de características. Tendo como objetivo representar, descrever, através de valores, uma imagem ou regiões da mesma. Cada medida extraída de uma determinada área compõe um vetor de características que define um padrão para a determinada área.

E como última etapa, tem-se o reconhecimento de padrões, onde é atribuído um rótulo baseado nas características extraídas na etapa anterior no intuito de diferenciar possíveis padrões contidos em um grupo de imagens, possibilitando a classificação da imagem ou partes dela.

4.2.2.1 Textura

A extração de características de uma imagem objetiva determinar aspectos que resultem em alguma informação quantitativa de interesse ou que sejam básicas para discriminação entre classes distintas. Como afirma Pedrini e Schwartz (2008a), a complexidade na análise de imagens está na definição de um conjunto de características que possa descrever de forma real cada região contida na imagem, de modo a serem utilizadas em processos de alto nível.

Como visto em Gonzalez e Woods (2002), a extração de características em imagens digitais são compreendidas por duas categorias de análises: textura e forma. Uma análise por textura busca descrever suavidade, rugosidade e regularidade da região de uma imagem. Enquanto uma análise por forma mensura informações da configuração e da aparência externa da região avaliada (FILHO et al., 2016a).

Segundo Haralick et al. (1973), textura é definida como a informação de uma região relacionada a coeficientes de uniformidade, densidade, aspereza, regularidade, intensidade, dentre outras características da imagem. A análise de textura é relevante em imagens digitais, uma vez que possibilita distinguir regiões da imagem que apresentam características semelhantes (AZEVEDO; CONCI; VASCONCELOS, 2018).

Nas próximas seções serão apresentadas as fundamentações dos descritores de textura baseados em Índices Taxonômicos, utilizados neste trabalho para a extração de características de textura.

4.2.2.2 Índices de Diversidade

Uma comunidade pode ser definida como um conjunto de espécies que ocorrem em um determinado lugar e tempo (MAGURRAN, 2013). Diversidade, por sua vez, caracteriza a variedade de espécies presentes na comunidade. A diversidade é um termo muito utilizado na área da ecologia, podendo ser mensurada a partir de duas componentes: riqueza de espécies e a abundância relativa.

A riqueza de espécies consiste no número de espécies encontradas em determinada região. Já a abundância relativa avalia o número de indivíduos de uma determinada espécie existentes numa dada área (PIANKA, 1994). As medidas de diversidade de espécies são geralmente úteis para comparar padrões em diferentes áreas.

Porém, como visto em Clarke e Warwick (1998a), diversidade não pode ser medida apenas com a utilização de dados como a abundância e a riqueza de espécies, carecendo de medidas que levem em conta questões evolutivas e de parentesco. Nesse aspecto, as medidas de diversidade filogenéticas, incorporam as relações de parentesco e proximidade entre espécies, aumentando a descoberta de padrões sobre a diversidades em diferentes comunidades (MAGURRAN, 2013).

Dentro do contexto das relações filogenéticas, o conceito de árvores filogenética, ou simplesmente filogenia, é tido com uma árvore onde as folhas representam os organismos, ou espécies, e os nós internos representam supostos ancestrais (Figura 38). As arestas da árvore denotam as relações evolutivas (ARAÚJO, 2003).

A combinação de abundância das espécies com a proximidade filogenética para gerar um índice de diversidade é denominada diversidade taxonômica (SILVA; BATALHA, 2006). A taxonomia, como ciência, lida com a classificação, identificação e nomenclatura das espécies, baseando-se no conjunto das distâncias entre pares de espécies acumuladas a partir das árvores filogenéticas (VANDAMME et al., 1996).

A diversidade taxonômica é muito sensível a perturbações ambientais e apropriado para avaliar as diferenças entre comunidades (SILVA; BATALHA, 2006). Uma comunidade em que as espécies estão distribuídas em muitos gêneros deve apresentar uma diversidade maior que uma comunidade em que a

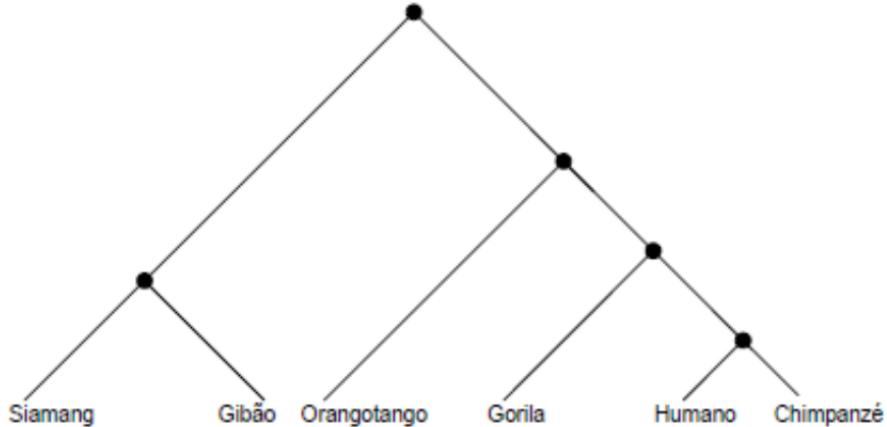


Figura 38 – Exemplo de árvore filogenética de primatas. Disponível em ([ARAÚJO, 2003](#)).

maioria das espécies pertence a um mesmo gênero, dessa forma a análise da diversidade taxonômica proporciona o entendimento de padrões sobre determinada comunidade ([RICOTTA, 2004](#)).

A aplicação dos índices de diversidade ao processamento de imagens, resulta na discriminação de características em uma imagem. Onde, a imagem ou região da mesma representa a “comunidade” e as “espécies”, nesse contexto, seriam os pixels, como visto em [Sousa et al. \(2011\)](#). A Figura 39 mostra um exemplo de uma região da imagem onde há diversidade de “espécies” em relação aos pixels. Assim, é possível a identificação de padrões em imagens, utilizando as relações taxonômicas.

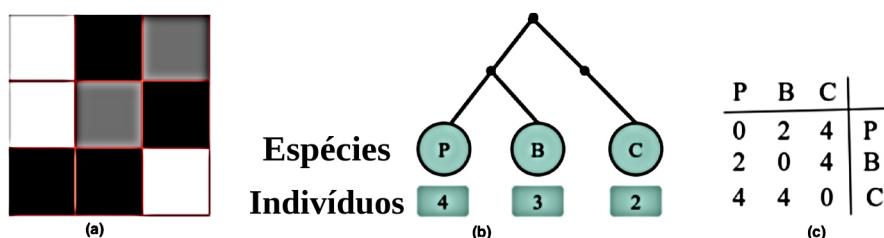


Figura 39 – (a) Representação da região analisada (b) Árvore filogenética extraída (c) Matriz com distâncias levando em consideração a quantidade de arestas da árvore filogenética que as separa as espécies analisadas. Disponível em ([CARVALHO et al., 2018a](#)).

4.2.2.2.1 Índices de Diversidade Taxonômica

O cálculo entre dois organismos escolhidos aleatoriamente em uma filogenia existente em uma comunidade é apresentado por índices de Diversidade Taxonômica (Δ) e Distinção Taxonômica (Δ^*) ([CLARKE; WARWICK, 1998a](#)). Como afirma [Filho et al. \(2017b\)](#) Estes índices possuem três fatores essenciais para aplicação: número de espécies, número de indivíduos e estrutura de conexão das espécies (número de arestas). Neste trabalho são utilizados esses dois índices para descrever a textura de imagens como retinas glaucomatosas e retinas normais.

O valor de Δ considera a abundância das espécies e a relação taxonômica entre elas. Portanto, seu valor expressa a distância taxonômica média entre dois indivíduos, escolhidos aleatoriamente de uma amostra ([GORENSTEIN, 2009](#)). Podendo ser definido pela Equação 4.1, onde X_i é a abundância da i -ésima espécie,

N é o número total de indivíduos e ω_{ij} é a distância da espécie i à espécie j na classificação taxonômica.

$$\Delta = \frac{\sum \sum_{i < j} \omega_{ij} X_i X_j}{[N(N - 1)/2]} . \quad (4.1)$$

Já o Índice de Distinção Taxonômica (Δ^*) representa a distância taxonômica média entre dois indivíduos, com a restrição de que sejam de espécies diferentes (GORENSTEIN, 2009). Este índice é definido por 4.2 onde X_i ($i = 0, \dots, s$) é a abundância da espécie i, X_j ($j = 0, \dots, s$) representa a abundância da espécie j e ω_{ij} é a distância da espécie i à espécie j na classificação taxonômica.

$$\Delta^* = \frac{\sum \sum_{i < j} \omega_{ij} X_i X_j}{\sum \sum_{i < j} X_i X_j} . \quad (4.2)$$

4.2.2.2.2 Classificação e Reconhecimento de Padrões

Em paralelo ao crescimento das tecnologias da informação, a complexidade dos problemas que essas tecnologias atendem também evoluem de forma igual, de modo que soluções estáticas não oferecem mais suporte a todos as necessidades inerentes a algumas situações, geralmente atreladas a problemas com um alto nível de abstração.

Para Kruse et al. (2016), a inteligência computacional se trata de “compreender conceitos, paradigmas, algoritmos e implementações para desenvolver sistemas que exibem comportamentos inteligentes em ambientes complexos”. Dessa forma, essa metodologia visa fornecer características, inerentes ao comportamento do aprendizado humano, dando às máquinas a capacidade de resolução de problemas muito abstratos, caracterizando as mesmas como “sistemas inteligentes”.

A técnica de reconhecimento de padrões é uma sub-área da inteligência computacional cujo objetivo é classificar objetos tendo base um conhecimento prévio ou informações estatísticas extraídas dos padrões (OLIVEIRA et al., 2013). Segundo Looney (1997), padrão é tudo aquilo para o qual existe uma entidade nomeável representante, entendido a partir de alguma forma de conhecimento.

O reconhecimento de padrões é um processo que procura a maior aproximação da classificação, buscando a melhor probabilidade de se estar certo. Assim, associando os conceitos de reconhecimento de padrões a resolução de problemas ligados à área de saúde, como afirma Netto et al. (2010), é possível analisar informações úteis ao diagnóstico, determinando a presença ou ausência de uma doença, dessa forma, aprimorando resultados dos diagnósticos.

4.2.2.2.3 Classificadores

Para a análise de padrões, na metodologia, foram escolhidas classificadores levando em consideração sua popularidade, eficácia, facilidade de adaptação nos experimentos e sua integração com a ferramenta de extração de características em imagens. Dentro o conjunto de candidatos, três se destacaram por apresentar boa aceitação em todos os aspectos analisados, sendo eles: *Multi Layer Perceptron* (MLP) *Sequential Minimal Optimization* (SMO) e *Random Forest*.

O MLP consiste em uma rede neural perceptron, formada por um conjunto de camadas. Este classificador tem em sua arquitetura uma camada de entrada, uma ou mais camadas ocultas, e uma camada de saída. A camada de entrada consiste em várias unidades que recebem entradas do mundo real, enquanto a camada de saída retorna os resultados para o mundo real, já as camadas ocultas são responsáveis por extrair padrões subjacentes das entradas (MONIKA; VENKATESAN, 2015).

O classificador SMO é um algoritmo para resolver o problema de programação quadrática que surge durante o treinamento de uma SVM. O SMO divide o problema de programação quadrática avaliado em uma série de menores problemas. Esses pequenos problemas são resolvidos analiticamente, o que permite que o SMO manipule conjuntos de treinamento muito grandes e possa distinguir padrões dentre os conjuntos de dados avaliados (PLATT, 1998).

Por fim, o Random Forest é uma combinação de preditores de árvores de decisão, de modo que cada árvore depende dos valores de um vetor aleatório amostrado de forma independente e com a mesma distribuição

para todas as árvores na “floresta”. Cada árvore dá um voto indicando sua decisão sobre a classe do objeto. Assim, a classe com o maior número de votos é escolhida para o objeto (MA et al., 2018).

4.2.2.3 Validação

Para avaliar o desempenho de modelos de classificação, geralmente são utilizadas as seguintes estatísticas: Sensibilidade (S), Especificidade (E) e Acurácia (A). Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em quatro classes, sendo elas: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

A Sensibilidade (S), é a capacidade de um teste diagnóstico identificar os verdadeiros positivos nos indivíduos que apresentam doença. Tem a função de indicar a eficácia do método, conforme a Equação 4.3.

$$S = \frac{VP}{VP + FN}. \quad (4.3)$$

Especificidade (E), trata-se da capacidade de um teste de diagnóstico identificar os verdadeiros negativos nos indivíduos verdadeiramente saudáveis (Equação 4.4).

$$E = \frac{VN}{VN + FP}. \quad (4.4)$$

A Acurácia (A) calcula a proporção de acertos, ou seja, o total de verdadeiramente positivos e verdadeiramente negativos da amostra. Como demonstra a Equação 4.5.

$$A = \frac{VP + VN}{VP + VN + FP + FN}. \quad (4.5)$$

Além da métricas citadas anteriormente, uma maneira bastante utilizada em avaliação do desempenho quantitativo de uma determinada técnica ou modelo proposto é a análise da curva Receiver Operationg Characteristic (ROC) (ERKEL; PETER, 1998). Como visto em Braga (2001), a área sob curva ROC (AUC) caracteriza-se como uma boa forma de avaliar um sistema de classificação binária, pois o mesmo implica em uma descrição empírica da capacidade do sistema de diagnóstico poder discriminar entre dois estados num universo, verificando-se o melhor desempenho, alcançado pelo método avaliado, quando o valor de AUC se aproxima de 1 (equivalente a 100%).

Por fim, como forma de avaliar o nível de concordância do experimento é usado o método estatístico de cálculo do Coeficiente Kappa (K) (COHEN, 1968c). Por ser um dado conservador, o Coeficiente Kappa, é utilizado para avaliar a reprodutibilidade dos resultados em fase de validação, onde, quanto mais próximo 1 for o resultado maior será a concordância.

A Equação 4.6 demonstra o cálculo de K, onde P_o é a taxa de aceitação relativa P_e é a taxa hipotética de aceitação.

$$K = \frac{P_o - P_e}{1 - P_e}. \quad (4.6)$$

A finalidade da utilização destas métricas é medir o desempenho da metodologia utilizada neste trabalho como satisfatório ou não. Com o intuito de apresentar pontos positivos e negativos que permita melhorias futuras deste trabalho.

4.2.3 Trabalhos Relacionados

A literatura disponível oferece estudos conhecidos sobre o problema da classificação automática do Glaucoma, bem como a aplicação de técnicas computacionais para extração de características. Nesta seção serão apresentados resumos desses trabalhos.

No sistema desenvolvido por Yu et al. (2005), são usadas técnicas de análise estatísticas method of moments para derivar características definidoras de imagens, distinguindo automaticamente imagens CSLT de nervos ópticos saudáveis e doentes. Nessa abordagem foi alcançado valores de acurácia de 86%, utilizando máquina de vetores de suporte (SVM) como classificador.

Foi proposto por [Bock et al. \(2007\)](#), um sistema automatizado de detecção de glaucoma usando imagens. Onde foram analisadas característica de intensidade de pixel usando *Linear Discriminant Analysis* (LDA) , de texturas com Filtro de Gabor, características espetrais com *Fast Fourier Transform* (FFT) e comparações com parâmetros de modelos de histograma, visando capturar estruturas glaucomatosas e avaliar os resultados. O sistema proposto alcançou 86% de taxa de sucesso em um conjunto de dados contendo uma mistura de 200 imagens reais de olhos saudáveis e glaucomatosos.

No CAD proposto por [Hatanaka et al. \(2010\)](#), são utilizadas imagem do fundo da retina, calculando a relação entre o Copo e o Disco Óptico, com objetivo de detectar o glaucoma. Nessa abordagem foi atingida uma acurácia total de 84,5% de acurácia, com a sensibilidade de 80% e especificidade de 85%.

Em [Acharya et al. \(2011\)](#), foi apresentado um método para detecção de glaucoma usando uma combinação de textura e espectros de ordem superior (HOS) de imagens de fundo digital. Nesse metodologia de classificação foram usadas imagens fornecidas pela Faculdade de Medicina Kasturba Medical College, e os resultados obtidos apontam uma precisão de 91% de acurácia para a metodologia proposta.

No trabalho de [Yadav, Sarathi e Dutta \(2014\)](#) é proposto uma metodologia de classificação automática de Glaucoma, utilizando *Gray Level Co-occurrence Matrix* (GLCM) e aplicação de *Principal Component Analysis* (PCA). O método de classificação fornece um alto nível de acurácia, que tem como valor de acertos 90% de acurácia.

Na metodologia de [Maheshwari, Pachori e Acharya \(2017\)](#) e apresentado um método de diagnóstico automático do glaucoma. Para aplicação da metodologia foram usadas duas bases de imagens, a primeira privada com 60 imagens. A segunda uma base de domínio público, que conta com 500, adquiridas com o Grupo de análises de imagens médicas (MIAG). Durante a realização da extração de características, nessa metodologia, foram utilizadas as transformadas de Wavelet Empírico e Features de Correntropia, resultando em uma classificação com sensibilidade de 100%, especificidade de 96,67% e acurácia em 98,33%.

[Junior et al. \(2018b\)](#) propõe a detecção do glaucoma através de descritores de textura baseados na diversidade filogenética. Nessa proposta é feita a segmentação do disco óptico de maneira automática, usando Otsu e k-means, como metodologias. No estudo foram utilizadas a bases públicas de imagens Drishti-GS e RIM-ONE, e como resultados foi constatado o valor de 99,6% de acurácia durante a etapa de classificação das imagens.

Os trabalhos relatados acima são modelos de metodologias baseadas em características de textura e reconhecimento de padrões para auxílio ao diagnóstico do glaucoma. No intuito de aumentar a acurácia e melhorar os valores de sensibilidade e especificidade, este trabalho pretende explorar um potencial referente a medidas de descrição de textura baseada em índices de Diversidade Taxonômicas.

4.3 Método proposto

A metodologia proposta neste trabalho teve como objetivo realizar a classificação automática do glaucoma, consistindo em quatro etapas distintas, a primeira, a aquisição de imagens dos bancos de imagens Drishti-GS ([SIVASWAMY et al., 2014a](#)) e RIM-ONE ([TRUCCO et al., 2013](#)); na segunda etapa é realizada a extração de características de textura utilizando os índices Taxonômicos, implementados neste trabalho com a biblioteca de manipulação de imagens OpenCV ([TEAM, 2017](#)); na terceira etapa, usa-se algoritmos disponíveis na ferramenta WEKA¹ para realizar a classificação das características extraídas; e por fim, são apresentadas as formas de validação dos resultados obtidos. A Figura 40 demonstra de maneira resumida as etapas seguidas na metodologia que será utilizada nesta proposta.

4.3.1 Aquisição de Imagens

Durante a etapa de aquisição das imagens, na metodologia proposta, foram usadas duas bases de imagens públicas, sendo elas a Drishti-GS e RIM-ONE v2. Ambas as bases são formadas por um conjunto de imagens da retinas, contendo imagens com DO saudáveis e glaucomatosos, apresentando grande aceitação no meio científico e sendo utilizadas em diversas metodologias de análises e detecção do glaucoma.

¹ <http://www.cs.waikato.ac.nz/ml/weka/>

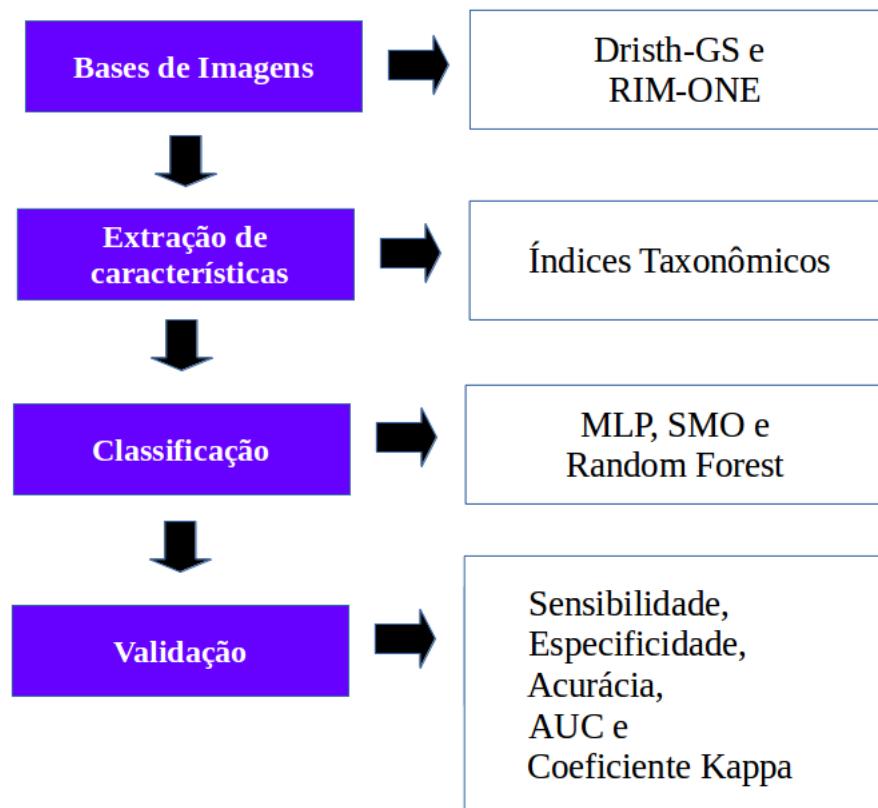


Figura 40 – Metodologia proposta.

O conjunto de dados Drirth-GS consiste em um total de 101 imagens, onde se verifica a presença de 70 imagens de olhos glaucomatosos e 31 imagem de olhos saudáveis. Todas as imagens, da base, foram coletadas no hospital de olhos Aravind, Madurai India. No qual, a seleção de pacientes com glaucoma foi feita por pesquisadores clínicos e os pacientes selecionados tinham entre 40 e 80 anos de idade, com um número aproximadamente igual de homens e mulheres. Cada imagem da base possui marcações manuais, obtidas de quatro especialistas, discriminatórias da região do DO e do Copo Ótico ([SIVASWAMY et al., 2014a](#)).

A base de domínio público RIM-ONE v2, foi disponibilizada pelos hospitais Universitários de Canaria, a Clínica San Carlos e o hospital Universitário Miguel Servet. A base é composta por um total de 455 imagens de retina analisadas por especialistas. Sendo que, são 200 imagens de olhos com glaucoma em vários níveis e 255 sem presença do glaucoma. Cada imagem possui uma máscara correspondente, essa máscara foi utilizada para que o método proposto fosse aplicado apenas sob a região do disco óptico ([TRUCCO et al., 2013](#)). As Figuras 41 e 42 apresentam exemplos das imagens contidas nas bases de imagens utilizadas neste trabalho.

4.3.2 Extração de Características

Embora as redes de aprendizagem profunda tenham mostrado-se promissoras na representação de informações em visão computacional, tais modelos de aprendizagem profunda supervisionados, exigem grandes quantidades de dados rotulados e recursos de computação, que são caros ([DU et al., 2017a](#)). Assim nessa proposta, empregamos uma metodologia tradicional de extração de características, aplicada a um conjunto de imagens de pequeno tamanho, visando demonstrar a eficiência da metodologia, aqui apresentada, na caracterização do glaucoma.



Figura 41 – Imagem de retina sem glaucoma disponível na base DRISHTI-GS

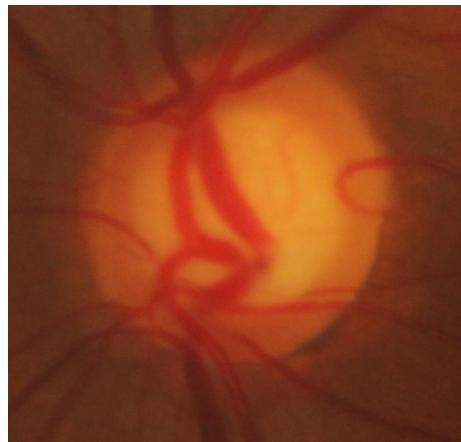


Figura 42 – Imagem de retina com glaucoma disponível na base RIM-ONE v2

Após a etapa de aquisição de imagens e aplicação das marcações do DO feitas por especialistas médicos, dando continuidade às etapas presentes na metodologia, as imagens foram encaminhadas à fase de extração de características de textura. Para descrição da textura dos objetos foram utilizados os Índices de Diversidade Taxonômica e Distinção Taxonômica, como apresentados na Seção 2.3.1.

Os índices taxonômicos se baseiam na distância filogenética, calculada a partir do número de aresta entre duas folhas em uma determinada árvore. Em nossa proposta, para representar as imagens, utilizamos o modelo de árvore enraizada na forma de cladograma inclinado. A Figura 43 traz um exemplo dessa arquitetura de árvore, onde as espécies são os valores de intensidade dos pixels de um determinado canal e os indivíduos são pixels da imagem.

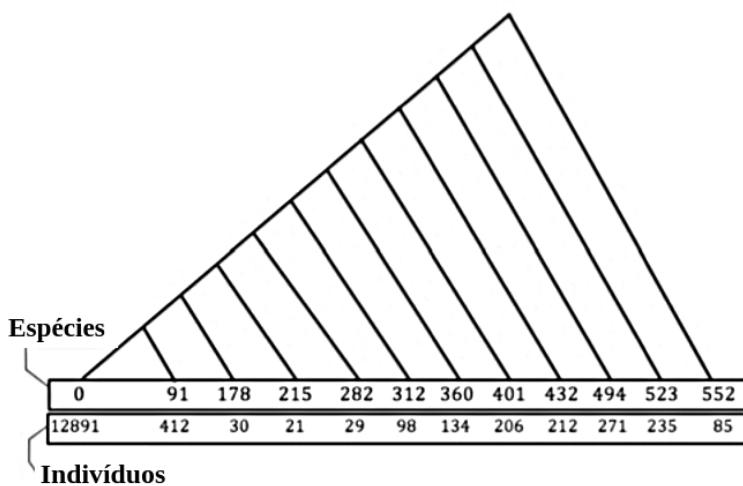


Figura 43 – Exemplo de árvore enraizada na forma de cladograma inclinado. Disponível em ([FILHO et al., 2017b](#)).

Em cada ROI analisada, foram extraídos os 2 índices de diversidade taxonômica, porém como diferencial da presente metodologia foram aplicados os cálculo de Diversidade Taxonômica e Distinção Taxonômica para os pixels em níveis de cinza, como no trabalho de [Filho et al. \(2017b\)](#), e para cada um dos 3 canais RGB resultando em um total de 8 características por imagem. Ao final do processo, todos os índices aplicados

às imagens caracterizam a textura das mesmas e os atributos extraídos buscam fornecer medidas de homogeneidade ou heterogeneidade de cada ROI.

4.3.3 Classificação

O processo de classificação, na presente metodologia, foi realizado com classificadores presentes no pacote de software WEKA. Essa ferramenta, agrupa um conjunto de algoritmos provenientes de diferentes abordagens voltadas para aprendizagem de máquina e mineração de dados. Também esse pacote de software permite o desenvolvimento de novos esquemas de aprendizado de máquina(HALL et al., 2009)

Dentre os classificadores presentes no pacote WEKA, foram selecionados: MLP, SMO e *Random Forest*. Esses classificadores foram escolhidos por apresentarem bons resultados, como visto na literatura, sendo rigorosos em relação a presença de ruídos no processo de classificação. O método utilizado para classificação foi o k-fold, com $k = 10$.

4.3.4 Validação

Após a finalização da etapa de reconhecimento de padrões, é necessário validar os resultados e discutir prováveis melhorias. Essa metodologia utiliza métricas comumente aplicadas a sistemas baseados em processamento de imagem. Para a validação dos resultados, foram utilizadas as métricas de avaliação baseadas em estatísticas: Sensibilidade (S), Especificidade (E), Acurácia (A), Área sob a curva ROC (AUC) e o Coeficiente Kappa (K).

4.4 Resultados e Discussões

Esta seção mostra e discute os resultados obtidos com a metodologia proposta por esse trabalho, obtendo assim, a segmentação das ROI em imagens do retina e classificação em glaucomatosas e normais. Este trabalho apresenta os testes realizados a partir das análises dos resultados provenientes da extração e características de cada canal RGB e níveis de cinza, além da combinação das características extraídas de todos os canais resultando em um vetor de 8 características.

Tabela 15 – Testes com níveis de cinza

Classificador	A (%)	E (%)	S (%)	AUC	K
MLP	93,3	89,1	97,7	0,91	0,86
SMO	66,9	90,2	42,2	0,66	0,32
<i>Random Forest</i>	94,2	94	94,4	0,98	0,88

Tabela 16 – Testes com o canal vermelho (Red)

Classificador	A (%)	E (%)	S (%)	AUC	K
MLP	93,5	88,4	98,8	0,92	0,87
SMO	83,8	89,1	78,1	0,83	0,67
<i>Random Forest</i>	96,7	97,2	96,20	0,99	0,93

Tabela 17 – Testes com o canal verde (Green)

Classificador	A (%)	E (%)	S (%)	AUC	K
MLP	80	68,1	92,5	0,83	0,60
SMO	64,3	90,9	36,3	0,63	0,27
<i>Random Forest</i>	86,1	84,9	87,4	0,94	0,72

Os valores presentes nas Tabelas 15, 16, 17, 18 e 19 apresentam um resumo dos principais testes realizados, demonstrando o desempenho da metodologia utilizada para descrever características de textura utilizando três classificadores, MLP, SMO e *Random Forest*.

Tabela 18 – Testes com o canal azul (Blue)

Classificador	A (%)	E (%)	S (%)	AUC	K
MLP	94,2	93,3	95,1	0,96	0,88
SMO	94,4	92,3	96,6	0,94	0,88
<i>Random Forest</i>	96,6	93	94,4	0,99	0,87

Tabela 19 – Testes com todos os canais e nível de Cinza

Classificador	A (%)	E (%)	S (%)	AUC	K
MLP	98,5	99,5	97,4	0,98	0,97
SMO	96	93,7	98,5	0,96	0,92
<i>Random Forest</i>	98,7	98,9	98,5	0,99	0,97

Nos experimentos descritos na Tabela 5 foi constatado que os melhores resultados apresentam valores acima de 98 % na taxa média de acertos, 98.5% para sensibilidade média e 98.9% para especificidade média. Sendo que o maior valor alcançado dentre todos os experimentos foi de 98,7% para acurácia, também apresentando os bons resultados de 0,99 para AUC e 0,97 para o coeficiente kappa, utilizando o classificador *Random Forest* juntamente com o índice de diversidade(Δ) e distinção taxonômica (Δ^*), aplicados a todos os canais de cores da imagem RGB.

Com base nos resultados obtidos, constata-se que o classificador *Random Forest* conseguiu os resultados mais promissores e equilibrados em todos os casos de testes na metodologia. Já o classificador SMO apresentou o menor resultado levando em consideração apenas o canal verde (Tabela 17), tendo o como pior valor para a A 64.3%.

A Tabela 20 apresenta uma breve comparação entre os resultados encontrados utilizando a metodologia propostas e alguns trabalhos citados no Seção 3, que realizam a classificação de regiões extraídas do DO em Glaucomatosas e Normais. Porém, algumas bases são diferentes da usada no presente trabalho. Portanto, os dados apresentados na Tabela 6, são apenas para ilustrar/comparar os resultados obtidos com os trabalhos da área, sem desmerecer nenhum dos métodos citados.

Tabela 20 – Trabalhos relacionados

Trabalho	Bases de Imagens	A (%)	E (%)	S (%)	AUC
Yadav, Sarathi e Dutta (2014)	Base Privada	90	-	-	-
Maheshwari, Pachori e Acharya (2017)	Base Privada e Base Pública MIAG	98,3	96,6	100	-
Junior et al. (2018b)	Drith-GS e RIM-ONE	99,6	99,3	100	0,9
Vinícius dos Santos Ferreira et al. (2018)	Drith-GS, DRIONS-DB e RIM-ONE	100	100	100	1
Método Proposto	Drith-GS e RIM-ONE	100	100	100	1

No que diz respeito às métricas de desempenho utilizadas neste trabalho, o ideal para um sistema CAD é ter um bom equilíbrio entre as três métricas de avaliação (acurácia, sensibilidade e especificidade), pois, um bom método deve ser capaz de classificar com sucesso, tanto os casos de positivos (que possuem a doença) como os casos de negativos (que não possuem a doença). Fica claro que na presente metodologia além dos valores de validação apresentarem equilíbrio, demonstram resultados robustos, motivando sua aplicação.

4.5 Conclusão

Este trabalho apresentou um método automático, com o uso dos Índices de Diversidade (Δ) e Distinção (Δ^*) Taxonômicos, junto com reconhecimentos de padrões, capaz de discriminar e classificar, em imagens, anormalidades em possíveis regiões suspeitas, resultando no auxílio ao diagnóstico de forma antecipada do glaucoma, sendo esta, uma segunda opinião para os especialistas.

Os índices de diversidade taxonômicos propostos neste trabalho, mostraram- se eficientes na tarefa de caracterização das regiões do Disco Óptico. Pois, foi possível alcançar resultados promissores, tendo

valores expressivos em todas as métricas de avaliação dos resultados, com a taxa de sucesso acima de 98%.

Os resultados obtidos demonstraram o desempenho próspero das técnicas de extração de textura pelos índices taxonômicos em união com o classificador *Random Forest*. Os bons resultados na avaliação da acurácia, sensibilidade, especificidade, AUC e coeficiente de concordância kappa, fornecem bases sólidas para aplicação da técnica em ambientes reais.

Desta forma, este trabalho apresenta contribuições em duas áreas de estudo. Na área da saúde, oferecendo uma metodologia automática para auxílio no diagnóstico do glaucoma. Já na computação, as contribuições consistem na adaptação de técnicas de outras áreas do conhecimento, adequando-as no contexto da computação, especificamente, na área do processamento de imagens propondo novos descritores de textura baseados na diversidade taxonômica.

Assim, acredita-se que a metodologia apresentada neste trabalho poderá integrar uma ferramenta CAD, sendo aplicada em casos reais e atuais na detecção do glaucoma. Assim, trazendo benefícios ao especialista, que poderá contar com uma segunda opinião durante a detecção do glaucoma, bem como garantindo ao paciente melhor qualidade de vida e um prognóstico mais favorável quanto ao tratamento do Glaucoma.

Como trabalhos futuros, pretende-se:

- Aumentar a quantidade e variabilidade das amostras de lesões utilizando mais bases de imagens;
- Desenvolver um método automático para a segmentação do DO; e
- Aplicar índices taxonômicos para investigar outras doenças oculares.

Análise de Textura de Nódulos Pulmonares para Sistemas CAD Utilizando Índice de Diversidade Funcional

Autores: Antônio Oseas de Carvalho Filho; William de Oliveira Torres.

Resumo: O câncer de pulmão é uma das principais causas de mortes por câncer no mundo. As chances de sobrevivência aumentam ao detectar a doença em seus estágios iniciais. Uma das melhores ferramentas usadas para diagnosticar é a tomografia computadorizada. O problema desse método é que, seu resultado depende diretamente do especialista que está analisando a imagem; esse processo é cansativo e pode levar ao erro. Sistemas de Detecção Auxiliado por Computação (*Computed-Aided Detection Systems - CAD*) vem se mostrando uma boa forma de auxílio aos especialistas. Esse trabalho apresenta o uso de descritores baseados no Índice de Diversidade Funcional (IDF) na etapa de redução de falsos positivos (FP) em tomografia computadorizadas. Para validar a eficiência da metodologia, foram usadas as métricas: acurácia, sensibilidade, especificidade, ROC e Kappa, alcançando 97.73%, 98.40%, 95.87%, 0.941, e 0.996, respectivamente. A metodologia proposta apresentou resultados satisfatórios e superiores ao estado da arte, com isso pode-se concluir que, o IDF é robusto e capaz de participar de um Sistema de Detecção Auxiliado por Computação real.

Keywords: Câncer de Pulmão, Imagens Médicas, Análise de Texturas, Diversidade Funcional.

5.1 Introdução

No ano de 2018, o câncer de pulmão teve um total de 2.093,876 novos casos em todo o mundo, levando a 1.761,007 mortes, representando 11,6% e 18,4% dos casos e mortes de câncer, respectivamente. O câncer de pulmão, quando comparado com outros 32 tipos, é o mais comumente diagnosticado e também o mais letal (BRAY et al., 2018a).

A detecção de nódulos pulmonares é crucial pois há uma grande chance de evoluir para câncer (Huang et al., 2013). A detecção de nódulos por tomografia computadorizada (TC) é uma tarefa complexa, tendo em vista que eles podem ter um contraste semelhante a outras estruturas de baixa densidade e tamanho

pequeno em uma área complexa (por exemplo, conectada a vasos sanguíneos ou nas bordas do pulmão) (LEE; KOUZANI; HU, 2010). Outro fator que dificulta a detecção é que os especialistas têm um grande número de TCs para analisar. Esse processo exaustivo e repetitivo pode levar a distrações e resultar em um mal-entendido da análise, especialmente quando as imagens exibem simultaneamente outras anomalias. Portanto, esse tipo de análise é frequentemente vulnerável a erros (OST; FEIN; FEINSILVER, 2003).

Devido a complexidade da tarefa de detecção de nódulos pulmonares, Sistemas de Detecção Auxiliadas por Computação (*Computer Aided Detection Systems - CAD*) começaram a ser desenvolvidos. CADs usam técnicas computacionais com a finalidade auxiliar os especialistas no momento da análise dos exames. O CAD usado como base por esse trabalho possui 5 etapas em seu processo, são elas: 1) Aquisição de imagens: etapa onde as imagens médicas são obtidas; 2) Segmentação: nessa etapa obtemos as regiões de interesse na imagem, no caso de TC do pulmão são obtidos os nódulos pulmonares; 3) Redução de Falsos Positivos (FP): analisa todas as áreas de interesse obtidas na etapa anterior e então válida se o que foi segmentado realmente é de interesse (Contribuição deste trabalho); 4) Extração de características: responsável pela obtenção de atributos para caracterização da imagem; 5) Classificação: trata-se da última etapa do sistema, nela as características extraídas são utilizadas como entrada para um algoritmo de classificação, onde a saída é, usualmente, o diagnóstico da imagem ou da área de interesse.

O presente trabalho trata a utilização de atributos baseados em índice de diversidade funcional na etapa de redução de falso positivos em imagens de TC do pulmão. O objetivo é reduzir FPs na detecção dos nódulos pulmonares, ou seja, após a segmentação dos nódulos muitas regiões são falso positivos (não-nódulos) e isso acaba prejudicando as etapas posteriores de um sistema CAD.

Para a tarefa de redução de FPs entre nódulos e não-nódulos em imagens de TCs é proposto um descritor que trabalha a caracterização apenas em texturas contidas em regiões segmentadas em exames de TC. Para essa tarefa, os conceitos de índice de diversidade funcional, provindos da biologia, foram empregados para extração de atributos de texturas. Optou-se por analisar exclusivamente a textura das imagens em decorrência de certas situações, como por exemplo: a) mesmo para os mais experientes especialistas, há regiões em que a diferenciação de contraste e textura entre tipos de tecidos pode passar despercebida; b) devido a natureza das imagens do exame, uma parte expressiva dos sistemas CAD fornece regiões arredondadas, abrindo assim uma maior chance de erros na classificação quando se utiliza à análise de forma para a descrição.

A principal contribuição desse trabalho é uma nova forma para se descrever texturas baseados em um índice ecológico, o índice de diversidade funcional para a redução de FPs em imagens TCs pulmonares. O uso desse descritor provê a possibilidade estimar o comportamento das texturas sobre cada imagem.

5.2 Trabalhos Relacionados

A segmentação pode gerar candidatos a nódulos incorretos, conhecidos como FP. O desafio enfrentado é reduzir o número de FPs sem perder os nódulos reais. Há trabalhos reconhecidos na literatura que abordam o mesmo problema enfrentado pelo método proposto. Para efeito de comparação, iremos listar alguns deles logo abaixo.

Em (Boroczky; Zhao; Lee, 2005) propõem um método de seleção de subconjuntos de características baseado em Algoritmos Genéticos para realiza a redução de FPs de nódulos pulmonares, ela utiliza 189 imagens, que a partir de cada imagem são extraídas 23 características, em seguida são selecionados os melhores atributos. O melhor resultado obtido foi utilizando 9 características, tendo o resultado de 92,4% de sensibilidade e 85,4% de especificidade usando a validação cruzada *leave-one-out*.

A proposta de (Setio et al., 2016) é utilizar a técnica *multi-view convolutional networks* (ConvNets) para a redução de FPs. Foram utilizadas um total de 888 imagens oriundas da base LIDC-IDRI. Atingindo uma sensibilidade de 87,9% da tarefa de redução de FPs.

No trabalho de (El-Regaily et al., 2017) os autores propõem o uso de um conjunto de redes neurais convolucionais (*Ensemble of Convolutional Neural Networks E-CNNs*) para a redução de FPs. São utilizadas 3 CNNs, cada uma diferenciando na quantidade de camadas e tamanho das imagens de entrada. Foram

utilizadas um total de 400 imagens de CTs. Esse método obteve uma sensibilidade de 77,77%, especificidade de 69,5% e acurácia de 70,53%.

A proposta de (FILHO et al., 2017) realiza a classificação de nódulos pulmonares, para isso, ele usa medidas de forma, analisando sua forma usando diagramas de forma, medidas de proporção e uma análise baseada em cilindro. Além disso, usa o classificador de máquina de vetores de suporte (Support Vector Machine - SVM). O método obteve uma acurácia de 95.33%.

Para a redução de FPs em TCs pulmonares, (NAQI; SHARIF; JAFFAR, 2018) utilizam *deep learning* baseada em *autoencoder* (LIOU et al., 2014) e *softmax*. Um total de 1018 imagens originadas da base LIDC-IDRI. Na etapa de redução de FP, foi obtida uma sensibilidade de 95.6%.

Por fim, o trabalho de (Qin et al., 2018) aplica um modelo de CNN 3D baseado na *DenseNet* para redução de FPs. São utilizadas 888 imagens vindas da base do desafio LUNA16. Esse método obteve uma sensibilidade de 98.30% na tarefa de redução de FPs.

Para facilitar a visualização dos trabalhos relacionados, a Tabela 21 mostra de forma mais direta o trabalho, a base de imagens utilizada, a quantidade de exames na amostra e a técnica utilizada para a tarefa.

Tabela 21 – Resumo dos trabalhos relacionados

Trabalho	Base	Amostra	Técnica
Boroczky, Zhao e Lee (2005)	Privada	189	Seleção de subconjuntos de características baseado em Algoritmos Genéticos.
Setio et al. (2016)	LIDIC-IDRI	888	<i>Multi-view Convolutional Networks</i> (ConvNets).
El-Regaily et al. (2017)	LIDIC-IDRI	400	Conjunto de redes neurais convolucionais (<i>Ensemble of Convolutional Neural Networks</i> E-CNNs).
Filho et al. (2017)	LIDIC-IDRI	833	Análise de forma usando diagramas de forma, medidas de proporção e uma análise baseada em cilindro.
Naqi, Sharif e Jaffar (2018)	LIDIC-IDRI	1018	Usam <i>deep learning</i> baseada em <i>autoencoder</i> e <i>softmax</i> .
Qin et al. (2018)	LUNA16	888	CNN 3D baseado na <i>DenseNet</i> .

Alguns trabalhos apresentados anteriormente tratam exclusivamente da redução de FPs, já outros, a redução é apenas uma das etapas presentes no trabalho. Apesar de obterem bons resultados, devemos levar em consideração que os trabalhos possuem algum tipo de problema como: 1) quantidade reduzida de imagens, 2) devido ao comportamento da doença, são geradas imagens com formas semelhantes entre nódulos e não-nódulos após a segmentação; 3) alto custo de processamento quando são utilizados algoritmos de *deep learning*. Dessa forma, utilizamos a base de imagens LIDC-IDRI(III et al., 2011b) para contornar o problema da quantidade de imagens, e atributos de textura para que o método seja menos sensível à segmentação e, ao mesmo tempo, necessite de um poder de processamento menor que um algoritmo de *deep learning*.

5.3 Referencial Teórico

Com o propósito de facilitar a compreensão do trabalho aqui proposto, serão abordados nesta seção conceitos e informações necessárias para a compreensão da proposta. O restante da seção está dividido nas seguintes sub-seções: 5.3.1 Câncer de Pulmão, 5.3.2 Tomografia computadorizada, 5.3.3 Processamento Digital de

Imagens, [5.3.4 Extração de características](#), [5.3.5 Classificação de Imagens](#) e por fim [5.3.6 Validação dos Resultados](#).

5.3.1 Câncer de Pulmão

O câncer de pulmão é o crescimento descontrolado das células no pulmão, levando então à formação de nódulos pulmonares. Segundo dados publicados em [Bray et al. \(2018a\)](#), dentre os tipos que podem atingir ambos os sexos, o câncer de pulmão é o mais comumente diagnosticado representando aproximadamente 1 a cada 10 dos casos de diagnósticos e a principal causa de óbito, representando aproximadamente 1 a cada 5 mortes por câncer no mundo.

De acordo com [INCA \(2019b\)](#), o tabagismo ou a exposição passiva ao tabaco são fatores importantes de risco que contribuem com o desenvolvimento do câncer de pulmão. Cerca de 85% dos casos diagnosticados feitos, o câncer está ligado ao consumo do tabaco ou de seus derivados.

O diagnóstico da doença pode ser feito pelos seguintes exames: Raio-X do tórax, complementado por TC e/ou broncoscopia. Uma vez obtida a confirmação da doença, é feito o estadiamento, que avalia o estágio de evolução, ou seja, verifica se a doença está restrita ao pulmão ou disseminada para outros órgãos. O estadiamento é feito por meio de vários exames, como biópsia pulmonar guiada por tomografia, biópsia por broncoscopia, tomografia de tórax, ressonância nuclear, PET-CT, cintilografia óssea, mediastinoscopia, ecobroncoscopia, entre outros ([INCA, 2019b](#)).

Para um tratamento adequado, é necessário o diagnóstico histológico e o estadiamento para definir se a doença está localizada no pulmão ou se existem focos em outros órgãos. Para os pacientes com doença localizada, e, particularmente, sem linfonodo (gânglio) aumentado (íngua) no mediastino (região entre os dois pulmões), o tratamento é cirúrgico, seguido ou não de quimioterapia e/ou radioterapia ([INCA, 2019b](#)).

5.3.2 Tomografia Computadorizada

Desde a década de 70, a TC vem sendo aclamada por ser uma ferramenta revolucionária na área de diagnósticos por imagens médicas. Muito disso se deve a forma com que a imagem é capturada na TC. A principal diferença da TC para a radiografia convencional, é possibilitar a visão transversal do corpo do paciente ao especialista ([SEERAM, 2010](#)).

Como explicada por [Seeram \(2010\)](#), a TC usa um computador para processar e reconstruir dados de transmissão de raios-X coletados do paciente. Para isso, é utilizado conhecimento multidisciplinar baseado em física, matemática, engenharia e ciência da computação. Sua evolução técnica continuada é marcada por diversos avanços recentes que fazem da TC uma das ferramentas mais importantes do século XXI.

Por sua natureza, a adoção dessa técnica, permite que a análise seja feita em qualquer parte do corpo do paciente, além de que, esse tipo de exame trata-se de uma exame não invasivo.

5.3.3 Processamento Digital de Imagens

Com a evolução tecnológica ocorrida nos últimos anos, o desenvolvimento de novos algoritmos para lidar com sinais bidimensionais permitem cada vez mais aplicações na área de processamento digital de imagens, com isso, despertasse o interesse para estudos nessa área ([ROCHA, 2014](#)).

No processamento digital de imagens, as entradas e saídas são imagens, cujos procedimentos englobam a extração de atributos e melhoram o aspecto visual de certas características, fornecendo informações que possam ser suficientes para a percepção e interpretação humana e seus dados possam ser interpretados de forma automática através de máquinas ([OLIVEIRA, 2013](#)).

Uma imagem pode ser definida como uma função bidimensional, onde x e y são coordenadas espaciais e a amplitude de f no par de coordenadas $f(x, y)$ é denominado intensidade ou nível de cinza. Quando x e y e a intensidade dos valores de f são finitos e discretos, tem – se então uma imagem digital clássica. Esses elementos possuem uma localização e um valor particular, chamados de *pixels* ([SOUSA, 2011](#)) e ([SILVA, 2016](#)).

As interpretações das informações podem ser feitas através de algoritmos computacionais, tomando como entrada uma imagem, que passa por técnicas de processamento digital de imagem para extração

de características, após esta etapa os dados extraídos são usados para o reconhecimento ou classificação, considerando um catálogo de padrões ou um banco de padrões (CONCI; AZEVEDO; LETA, 2008).

Para análise de imagens digitais é fundamental que as regiões de interesse sejam separadas das demais, com isso evita-se que haja perda de processamento por parte da máquina com dados irrelevantes, para o sistema em desenvolvimento (SOUZA, 2011).

5.3.4 Extração de Características

Na análise de imagens, visão computacional e reconhecimento de padrões, os sistemas necessitam-se reconhecer elementos ou objetos de interesse nas imagens, para isso, são estabelecidos parâmetros quantificáveis, que dependem de cor, posição, orientação, dimensões, textura, para poder ser realizado computacionalmente (CONCI; AZEVEDO; LETA, 2008).

A extração de características é uma forma utilizada para extrair dados em uma imagem que possa assim caracterizar a mesma, onde, a partir de um conjunto grande de características, seja extraída aquelas que são mais importantes para o problema, formando um conjunto mais reduzido de características por imagem digital (JR, 2014).

A extração de características tem como objetivo mensurar e quantificar padrões perceptíveis ou não visualmente, com a finalidade de categorizar objetos nas imagens, de acordo com aspectos morfológicos de interesse (JR, 2014).

Características de textura são determinadas como um conjunto de alguns padrões existentes em superfícies físicas, perceptíveis ao olho humano, trazendo grande quantidade de informação sobre a natureza da superfície, como suavidade, rugosidade e regularidade. A análise feita através de texturas é útil em aplicações, por se aproximar da avaliação feita pelo sistema visual humana (JR, 2014).

Após obter o conjunto de atributos que caracterizam a imagem, é a vez de verificar a eficácia dos mesmos. Para essa tarefa aplicamos os dados em um algoritmo de classificação e, baseado em sua saída, é possível verificar se as características utilizadas são capazes de descrever a imagem de forma eficaz.

5.3.5 Classificação de Imagens

O processo de classificação pode ser realizado considerando classes previamente definidas ou não definidas. Nas classes previamente definidas cada parâmetro que caracteriza cada classe é obtido através de treinamento, executados antes da aplicação do algoritmo de classificação. Com as classes não definidas, não se dispõe de parâmetros ou informações previamente coletadas a aplicação do algoritmo de classificação (JR, 2012).

A classificação consiste em reconhecer novos objetos, a partir de descritores gerados dos mesmos, tomando decisões a partir da extração de características do mundo real em imagens, isso é feito por meio de indagações simples a respeitos dos parâmetros extraídos, ou por meio de algoritmos de inteligência artificial (CONCI; AZEVEDO; LETA, 2008).

No processamento de imagens digitais, classificar uma imagem consiste em pegar um conjunto de características extraídas de um grupo de imagens do objeto de interesse e treinar um sistema para que possa reconhecer novos objetos a partir dessas características.

O reconhecimento de padrões ou características, usa alguns métodos ou técnicas, desde as mais simples, baseadas em vizinhos mais próximos, funções discriminantes, árvore de decisão, até as mais complexas, baseadas em redes *baysianas*, lógica *fuzzy* e redes neurais, para a classificação de imagens de acordo com os descritores extraídos (CONCI; AZEVEDO; LETA, 2008).

Para o desenvolvimento deste trabalho, a classificação será feita através da aplicação de algoritmos de *machine learning*, que implementa técnicas de reconhecimento de padrões redução de falsos positivos de nódulos pulmonares em imagens de tomografia computadorizada pulmonares.

5.3.6 Validação dos Resultados

Para validar o resultado obtido da etapa de classificação, foram utilizadas estatísticas/métricas que nos permitem descobrir a eficiência do método para classificar corretamente os VOIs que possuem lesões sobre

os VOIs que não possuem a lesão. As métricas foram, acurácia (Ac), sensibilidade (Sens), especificidade (Esp) (DUDA; HART, 1973), área sob a curva ROC (ROC) (MATOS, 2017), e índice Kappa (COHEN, 1968a).

Os cálculos de acurácia, sensibilidade e especificidade são feitos em cima de quatro valores que são fornecidos ao final da avaliação, são eles verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP) e falso negativo (FN). Nesse trabalho, o caso positivo é atribuído para os nódulos e os não-nódulos são dados como caso negativo, ou seja, o VP é o nódulo (caso positivo) que foi classificado como positivo, o FP é um não-nódulo (caso negativo) que foi classificado como positivo, o VN é um não-nódulo (caso negativo) que foi classificado negativo, por fim, o FN é quando um nódulo (caso positivo) é classificado como negativo.

Sensibilidade é uma métrica usada para medir a eficácia do método em identificar corretamente os casos VPs, ou seja, o qual eficaz é o método para identificar os nódulos. Para obter o valor da sensibilidade, é utilizada a Equação 5.1, onde a sensibilidade é igual ao verdadeiro positivo dividido pela soma dos verdadeiros positivos mais os falsos negativos.

$$Sens = \frac{VP}{VP + FN} \quad (5.1)$$

A especificidade é a métrica adotada para caracterizar o quanto eficaz o método é na identificação correta de VN, ou seja, o quanto eficaz é o método para identificação de não-nódulos. Para obter seu valor é usada a Equação 5.2, onde a especificidade é dada pela divisão dos verdadeiros negativos pela soma dos verdadeiros negativos mais os falsos positivos.

$$Esp = \frac{VN}{VN + FP} \quad (5.2)$$

A acurácia calcula a taxa de sucesso, ou seja, a porcentagem de VPs e VNs no teste, pode ser calculada de acordo com a Equação 5.3, onde a acurácia é dada pela soma dos verdadeiros positivos mais os verdadeiros negativos dividido pela soma dos verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos positivos.

$$Ac = \frac{VP + VN}{VP + VN + FN + FP} \quad (5.3)$$

A área sobre a curva ROC é uma das melhores formas assimilar a associação muitas vezes contraditória entre especificidade e sensibilidade. Ela traduz a probabilidade de que, diante de um caso positivo e um outro negativo, a regra do classificador tenda ao caso positivo. (MATOS, 2017)

Por fim, usamos também o índice Kappa (K) para avaliar os resultados (COHEN, 1968a). Quanto mais próximo de 1 for o valor Kappa obtido, maior será a concordância. De acordo com Landis e Koch (1977c), há uma categorização dos níveis de desempenho de classificação pelo índice Kappa que pode ser vista na Tabela 22.

Tabela 22 – Níveis de precisão da classificação, segundo o índice Kappa.

Índice Kappa (k)	Qualidade
$K < 0.2$	Pobre
$0.2 \leq K < 0.4$	Razoável
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito Bom
$K \geq 0.8$	Excelente

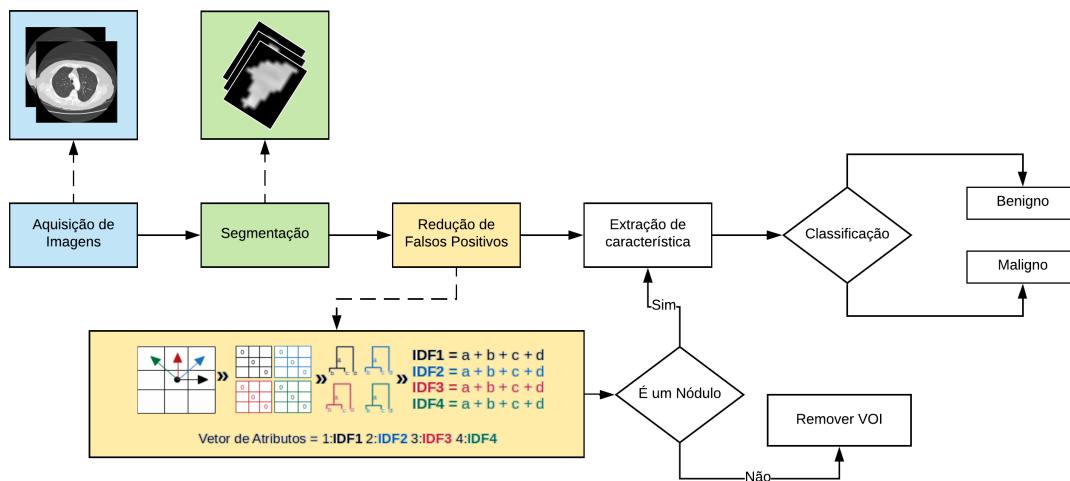
Fonte: Landis e Koch (1977c)

5.4 Metodologia

Nesta seção, será exibido a metodologia proposta para classificação de regiões suspeitas em nódulo e não nódulo, isto é, para redução de falsos positivos. A primeira etapa foi a aquisição de imagens, elas foram adquiridas a partir da base LIDC-IDRI (III et al., 2011b) usando o método de (FILHO et al., 2014a) para segmentação dos *Volumes of Interest* (VOI). Após esta etapa, aplicou-se a extração de características utilizando

os índices de diversidade funcional (Torres et al., 2018). Para finalizar, é realizada a classificação e a validação dos dados extraídos. A Figura 44 apresenta uma visão geral de um CAD completo, com ênfase na etapa de redução de falsos positivos que é foco desse trabalho.

Figura 44 – Fluxograma do CAD usado como base do trabalho focado em nosso método de redução de FPs (caixa amarela).



5.4.1 Aquisição das Imagens

A base de imagens usada para esse trabalho foi a LIDC-IDRI (III et al., 2011b), trata-se de um recurso de acesso gratuito e que foi projetado para treinar e avaliar com precisão os CADs para detecção de nódulos e diagnósticos de câncer de pulmão. Além disso, diversos trabalhos na literatura também usam esse conjunto de dados, como podemos ver em (FILHO et al., 2017c), (FILHO et al., 2017), e (Dey; Lu; Hong, 2018), permitindo uma melhor comparação dos resultados.

As TCs contidas nessa base têm origem em tomógrafos diferentes, isso proporciona uma dificuldade adicional na tarefa de classificação de nódulos pulmonares, pois cada equipamento produz as imagens de maneira diferente. Devido a isso, usá-los torna a generalização uma tarefa mais complexa para algoritmos. Com isso, se uma metodologia conseguir uma alta taxa de acertos, provavelmente terá bons resultados em protocolos diferentes.

As amostras utilizadas neste trabalho apresentam dois tipos de VOIs: nódulo e não-nódulo. Para extrair os VOIs dos nódulos, utilizou-se as marcações fornecidas pelos especialistas da *Lung Image Database Consortium* (LIDC). Para a segmentação dos não-nódulos foi utilizada a metodologia de (FILHO et al., 2014a). Assim, foi possível usar candidatos a não-nódulos que foram encontrados em um cenário real e nódulos fornecidas pelos especialistas.

O método de Filho et al. (2014a) é composta por três etapas. A primeira etapa, é realizada a segmentação e reconstrução do parênquima pulmonar, seguida por um realce para destacar suas estruturas. Na segunda etapa, os candidatos a nódulos são segmentados utilizando clusterização. Finalmente, no terceiro estágio, as características desproporção esférica, densidade esférica, esfericidade, distância radial ponderada, alongamento e índice de forma radial Boyce-Clark são extraídos, selecionados e classificados usando um *Support Vector Machine*.

Quando juntos, os VOIs não-nódulos vindos do método (FILHO et al., 2014a) e as VOIs nódulos vindas das marcações dos especialistas, foram obtidas 17741 VOIs não-nódulos e 6407 VOIs de nódulos formando assim as 24148 VOIs que foram utilizada nos testes presente trabalho.

5.4.2 Índice de Diversidade Funcional

Na ecologia, é comum o uso de índices de riqueza e abundância das espécies como uma forma de caracterização da biodiversidade. No entanto, estudos mais recentes focados em diversidade vêm empregando o conceito de diversidade funcional. A diversidade funcional é considerada uma das melhores formas de predizer a produtividade e a vulnerabilidade de um ecossistema do que outros índices (SCHLEUTER et al., 2010).

Conforme Tilman (2001), a diversidade funcional no funcionamento das comunidades expressa a influência que as espécies e suas características possuem. Sendo assim, ao se mensurar a diversidade funcional, se consegue ter ideia da pluralidade de características funcionais que compõem os fenótipos dos organismos que agem nos processos da comunidade.

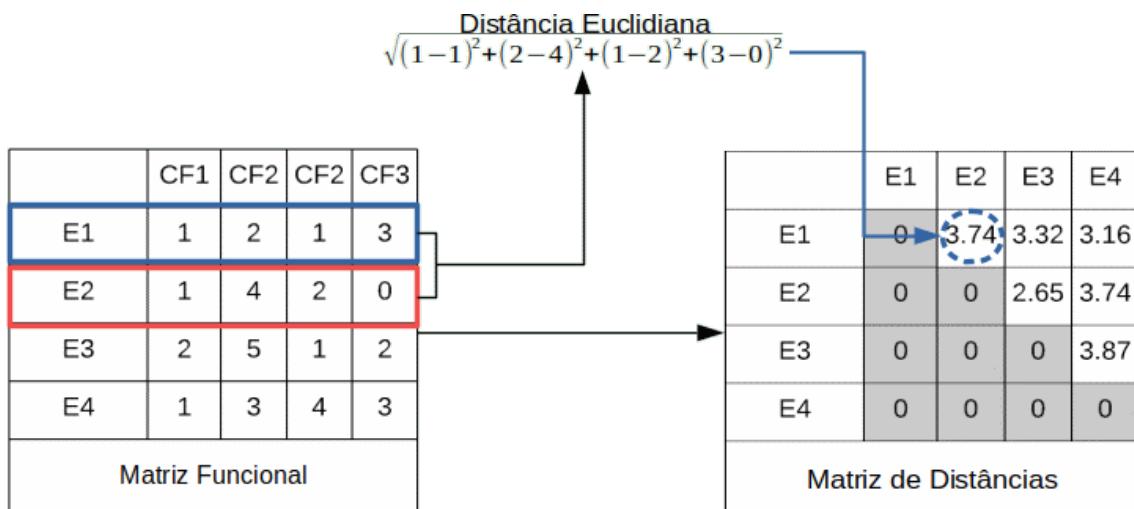
Para exemplificar, imaginemos duas comunidades ambas com a mesma quantidade de espécies, vamos chamar elas de A e B. Se A for composta apenas por aves e B for composta por lagartos, mamíferos e aves. B será mais diversificado pois há mais espécies funcionalmente distintas do que A (CIANCIARUSO; SILVA; BATALHA, 2009b).

Calcular a diversidade funcional é calcular a importância que cada espécie (E) desempenha em cada característica funcional (CF) na comunidade em que está presente. Diferentemente das características fenotípicas, a diversidade funcional é calculada independentemente da quantidade de membros da espécie na comunidade. Isso quer dizer que, considerando uma imagem como sendo uma comunidade, ao invés de, considerarmos unicamente as relações entre as espécies, consideramos as relações que uma espécie têm com seus vizinhos. Isso nos permite caracterizar e explorar texturas sobre certa região.

A Diversidade Funcional (IDF), consiste na soma dos comprimentos dos braços de um dendrograma funcional, ou seja, um dendrograma gerado a partir de uma matriz de *espécies x características funcionais* (PET-CHEY; GASTON, 2002).

Para o cálculo do IDF em uma imagem, primeiramente gera-se uma matriz de coocorrência de níveis de cinza (*Gray level Co-occurrence Matrix - GLCM*) (V.; UNNIKRISHNAN; BALAKRISHNAN, 2012), chamada matriz funcional em nosso método, como mostramos na Figura 45. Depois disso, criamos a matriz de distância (Figura 45) e logo após o dendrograma (Figura 46) é gerado pelo agrupamento da menor distância entre as espécies da matriz de distância e, finalmente, a soma de seus galhos está resultando em uma característica única.

Figura 45 – A Matriz Funcional exibe a relação entre espécies E e a importância delas em relação a uma Característica Funcional CF. Após a Matriz Funcional ser gerada, é criada então a Matriz de Distâncias onde, utilizando o cálculo de Distância Euclidiana, é obtido a distância entre uma espécie e outra.



Para uma melhor compreensão, a Tabela 23 exibe a correlação entre os termos utilizados pela biologia e os termos adotados no método proposto.

Figura 46 – Baseado na Matriz de Distâncias, um Dendrograma é gerado onde espécies são agrupadas de acordo com sua proximidade a outra. Por fim, é gerado o índice de diversidade funcional baseado na soma de todos os galhos do dendrograma.

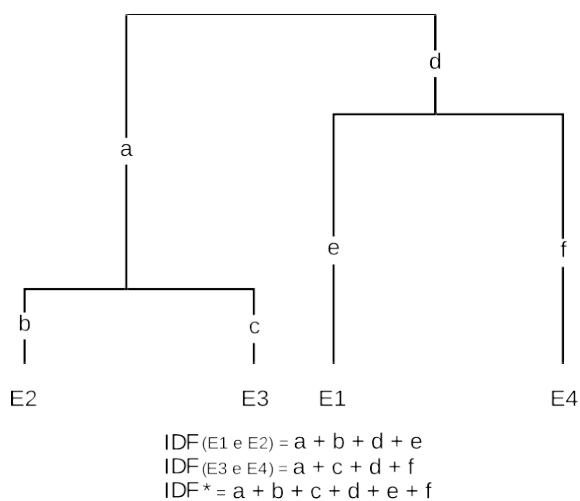


Tabela 23 – Correlação entre os termos utilizados pela biologia e os termos adotados no método proposto.

Termos de Biologia	Termos do método
Comunidade	VOI
Espécies	Valor de pixel da comunidade
Indivíduos	Número de pixels de uma espécie
Características fenotípicas das espécies	Relação de textura entre espécies

Para a extração de características de textura utilizando o IDF, reduziu-se a imagem que originalmente possuem 12 bits para 9, 7, e 5 bits utilizando a técnica de *quantization* (GERSHO, 1978), para reduzir o tamanho da matriz de coocorrência e reduzir o tempo de processamento. Foram calculadas as matrizes de coocorrência predefinidas para quatro angulações (0° , 45° , 90° e 135°) distância entre *pixels* D variando entre 1 a 3. Os índices gerados nas etapas subsequentes são armazenados em um vetor de características, para serem classificados posteriormente. Para facilitar a compreensão, a Figura 47 ilustra como funciona o algoritmo responsável pela etapa de extração de características.

5.4.3 Classificadores

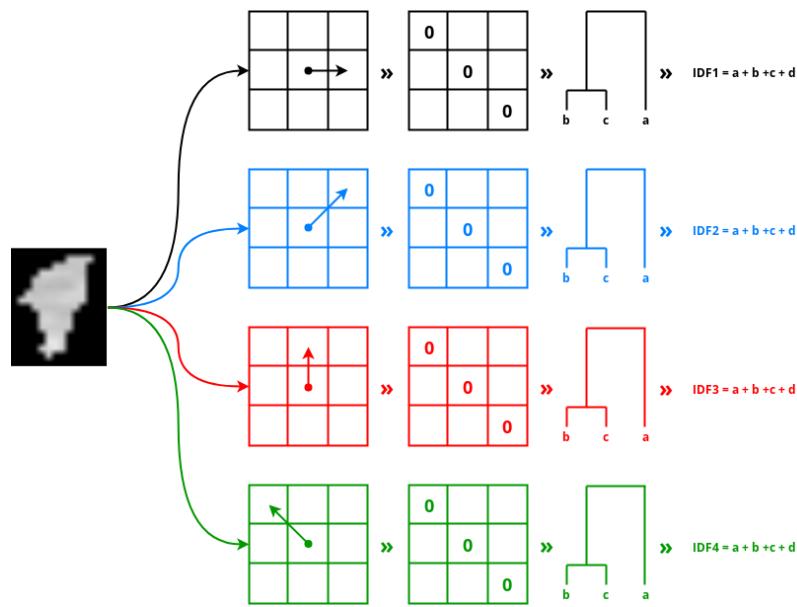
A classificação das características obtidas foram feitas pela suíte de algoritmos de aprendizagem de máquinas e mineração de dados WEKA (WITTEN et al., 2016), que contém ferramentas para pré-processamento de dados, classificação, regressão, agrupamento, regras de associação e visualização.

Os experimentos foram realizadas com a validação cruzada ($k = 10$). Este método dividiu os dados em k grupos; usa $k-1$ grupos para treinar o modelo e um para o teste. Repetimos esse processo k vezes, sempre mudando o grupo de teste. No final, todos os grupos são aplicados como um conjunto de testes e geramos uma pontuação média.

Para classificação, foi utilizado os classificadores: *Reduced Error Pruning Tree* (REPTree) (KALMEGH, 2015), *Instance-Bases learning with parameter K* (IBK) (AHA; KIBLER, 1991), *Random Forest* e *SVM* (CORTES; VAPNIK, 1995a). Esses algoritmos foram escolhidos a partir de um conjunto inicial de algoritmos candidatos devido à sua aplicação em trabalhos relacionados. Esses classificadores são robustos à presença de ruído, apresentando bons resultados na categorização de padrões de nódulos e não-nódulos em imagens de TC de pulmão. Os classificadores usados aqui tiveram seus parâmetros aprimorados através de tentativas e erros; no final, os parâmetros utilizados foram:

- **REPTree:** foi usado o número mínimo de instâncias igual a 5, a variação mínima para uma divisão

Figura 47 – Uma visão geral do método proposto para extrair recursos de um VOI segmentado. IDF1, IDF2, IDF3 e IDF4 são as quatro características calculadas.



em 0.001, o número de *folds* como 3, o *seed* para o embaralhamento aleatório de dados é 1 e *no pruning*.

- **IBK:** o número de vizinhos é 3, o número máximo de instâncias de treinamento mantidas é 0, foi habilitado a seleção do número de vizinhos mais próximos entre 1 e o valor *k* especificado usando a avaliação de espera única nos dados de treinamento, usando o algoritmo *Linear NN-Search* para o algoritmo de busca de vizinhos mais próximo usando a distância euclidiana.
- **Random Forest:** o tamanho de cada bolsa é 100, o número de iterações é 1000, o número de *slots* de execução é 3, o número de atributos a serem investigados aleatoriamente é 0, o número mínimo de instâncias por folha é 1.0, a proporção de variação numérica mínima de classe de a variação do trem para a divisão é 0.001 e a semente do gerador de números aleatórios é 1.
- **SVM:** utilizou-se o tipo de *kernel* SVM C-SVC ($C = 1.0$), o tipo de função do *kernel* é polinomial com grau 3, foi definido o gama na função do *kernel* igual a 0,0, definimos o parâmetro *nu* a 0,5, o *epsilon-SVR* é 0.001, a tolerância do critério de terminação é 0,001, o *epsilon* na função de perda do *epsilon-SVR* é 0,1.

5.5 Resultados

Para validar o método, foi usada a base de imagens LIDC-IDRI. A extração das características da textura foi realizada a partir dos descritores desenvolvidos utilizando o IDF e classificando as imagens em nódulos e não-nódulos, utilizando os classificadores com validação cruzada ($k = 10$).

Foram implementadas duas formas de testes. Na primeira forma, as imagens foram classificadas usando todas as distâncias em cada quantização separadamente (Tabelas 24, 25 e 26). Na segunda forma, os vetores de características de todas as distâncias de cada quantização foram combinados e depois classificados (Figura 48). As estatísticas apresentadas aqui servem para aumentar a compreensão dos resultados; no entanto, para escolher os melhores e os piores resultados, usamos o índice *Kappa* como métrica.

As tabelas onde estão os resultados dos testes, as siglas adotadas *D*, *A_c*, *Sens*, *Esp* e *ROC* representam Distância, Acurácia, Sensibilidade, Especificidade e área sob a curva ROC, respectivamente.

A Tabela 24 apresenta os resultados dos testes realizados com as imagens em quantização em 5 bits e cada distância separada. A melhor combinação foi o classificador *Random Forest* com a distância 1, tendo alcançado a acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 90.70%, 91.82%, 87.02%, 0.945 e 0.7518 respectivamente. Já a pior foi com classificador *SVM* com a distância 1, tendo atingido a

Tabela 24 – Resultados dos testes das imagens com a quantização igual a 5 Bits.

D	Classificador	Ac(%)	Sens(%)	Esp(%)	ROC	Kappa
1	Random Forest	90,70	91,82	87,02	0,9450	0,7518
	REPTree	89,11	90,51	84,34	0,9060	0,7074
	SVM	74,49	74,27	92,12	0,5200	0,0585
	IBK	82,88	83,29	80,61	0,7550	0,4927
2	Random Forest	88,01	91,16	76,85	0,9250	0,6562
	REPTree	86,57	91,02	74,45	0,8990	0,6405
	SVM	74,93	76,57	58,22	0,5720	0,1839
	IBK	80,51	83,79	69,48	0,7750	0,4830
3	Random Forest	52,39	92,46	23,11	0,9250	0,6562
	REPTree	84,71	92,55	67,77	0,8990	0,6305
	SVM	76,06	80,49	56,92	0,6460	0,322
	IBK	79,88	86,24	62,14	0,7750	0,4830

-Negrito é o melhor resultado.

acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 74.49%, 74.27%, 92.12%, 0.5200 e 0.0585 respectivamente.

Tabela 25 – Resultados dos testes das imagens com a quantização igual a 7 Bits.

D	Classificador	Ac(%)	Sens(%)	Esp(%)	ROC	Kappa
1	Random Forest	96,15	97,17	93,27	0,9910	0,9008
	REPTree	95,57	96,47	92,98	0,8852	0,8730
	SVM	74,42	74,22	92,62	0,5190	0,0548
	IBK	88,87	91,39	81,23	0,8950	0,7081
2	Random Forest	93,25	97,21	83,66	0,9810	0,9321
	REPTree	92,65	96,88	82,53	0,9690	0,8179
	SVM	74,88	76,50	58,05	0,5710	0,1804
	IBK	87,98	86,05	99,08	0,7750	0,6408
3	Random Forest	93,23	97,21	83,66	0,9810	0,8321
	REPTree	89,66	97,44	74,20	0,9540	0,7550
	SVM	75,82	80,22	56,36	0,6410	0,3129
	IBK	86,38	93,75	70,43	0,9000	0,6707

-Negrito é o melhor resultado.

A Tabela 25 apresenta os resultados dos testes realizados com as imagens em quantização em 7 bits e cada distância separada. A melhor combinação foi o classificador *Random Forest* com a distância 1, tendo alcançado a acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 96.15%, 97.17%, 93.27%, 0.991 e 0.9008 respectivamente. Já a pior foi com classificador SVM com a distância 1, tendo o atingido a acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 74.42%, 74.22%, 92.62%, 0.5190 e 0.0548 respectivamente.

A Tabela 26 apresenta os resultados com as imagens em quantização em 9 bits e cada distância. A melhor combinação foi o classificador *Random Forest* com a distância 2, tendo alcançado a acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 89.88%, 94.13%, 79.08%, 0.955 e 0.7457, respectivamente. Já o pior foi com classificador SVM com a distância 1, tendo o atingido a acurácia, sensibilidade, especificidade, área ROC e *Kappa* igual a 74.44%, 74.23%, 93.33%, 0.519 e 0.0552, respectivamente.

Por fim, foi concatenado os recursos em todas as distâncias para cada experimento de quantização. O classificador *Random Forest* também obteve o melhor resultado neste experimento. Na figura 48 mostramos um gráfico onde é possível ver a comparação entre cada quantização no melhor classificador.

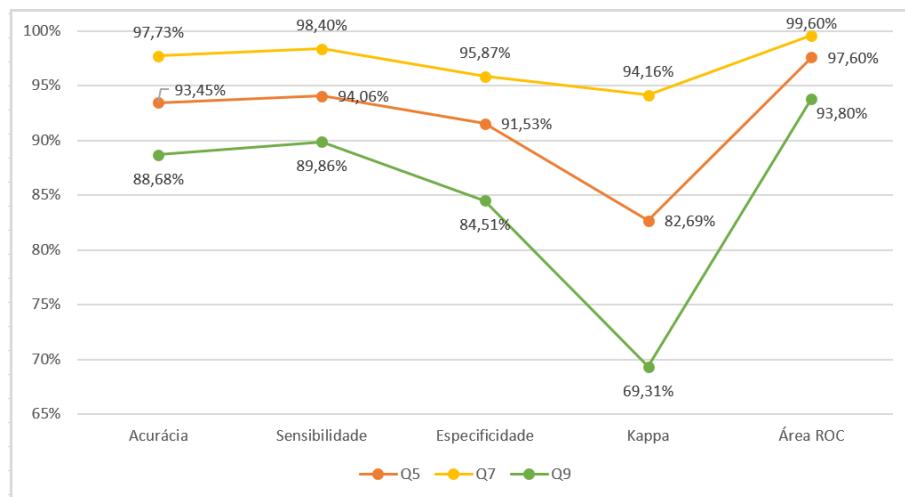
Como mostra o gráfico da Figura 48, o melhor resultado entre as quantizações com todas as distâncias

Tabela 26 – Resultados dos testes das imagens com a quantização igual a 9 Bits.

D	Classificador	Ac(%)	Sens(%)	Esp(%)	ROC	Kappa
1	Random Forest	88,05	91,08	82,60	0,9420	0,7099
	REPTree	87,09	89,61	78,92	0,8960	0,6565
	SVM	74,44	74,23	93,33	0,5190	0,0552
	IBK	80,98	85,28	66,43	0,7890	0,4893
2	Random Forest	89,88	94,13	79,08	0,9550	0,7457
	REPTree	88,49	93,08	76,74	0,9280	0,7103
	SVM	74,88	76,49	58,04	0,5710	0,1800
	IBK	80,79	88,37	65,72	0,7990	0,4874
3	Random Forest	88,22	96,30	72,09	0,9510	0,7208
	REPTree	87,42	95,65	70,94	0,9320	0,7017
	SVM	75,82	80,22	56,36	0,6410	0,3129
	IBK	81,80	88,55	64,71	0,9320	0,5429

-Negrito é o melhor resultado.

Figura 48 – Resultados de todas as distâncias juntas em cada quantização com o classificador *Random Forest*. Q5, Q7 e Q9 são a quantização com 5,7 e 9 bits, respectivamente.



juntas ficou com a quantização de 7 bits, onde a acurácia, sensibilidade, especificidade, *Kappa* e área ROC igual a 97,73%, 98,40%, 95,87%, 0,9416 e 0,996 respectivamente. Já o pior foi com a quantização em 9 bits, tendo atingido a acurácia, sensibilidade, especificidade, *Kappa* e área ROC igual a 88,68%, 89,86%, 84,51%, 0,6931 e 0,938 respectivamente.

A tarefa de comparação dos trabalhos relacionados não é uma tarefa trivial, pois, muitos dos métodos propostos utilizam bases distintas, técnicas com abordagens diferentes. Todos estes fatores devem ser levados em consideração no momento da comparação, assim, não é possível fazer uma comparação fidedigna. Dessa forma, o intuito com a Tabela 27 é apenas de apresentar um levantamento quantitativo das métricas usadas pelos autores na avaliação dos resultados em cada trabalho.

Como mostrado na Tabela 27, os resultados obtidos pelo método proposto são promissores e podem ser comparáveis com os demais trabalhos da literatura. Além disso, superamos o estado da arte para redução de falso positivos em imagens de TC do pulmão para detecção de nódulos.

Tabela 27 – Comparação do método proposto com trabalhos relacionados.

Trabalho	Base	Amostra	Ac(%)	Sens(%)	Esp(%)	ROC	Kappa
Boroczky, Zhao e Lee (2005)	Privada	189	-	92,40	85,40	-	-
Setio et al. (2016)	LIDC-IDRI	888	-	89,70	-	-	-
El-Regaily et al. (2017)	LIDC-IDRI	400	70,53	77,77	69,50	-	-
Filho et al. (2017)	LIDC-IDRI	833	95,33	-	-	-	-
Naqi, Sharife Jaffar (2018)	LIDC-IDRI	1018	-	95,60	-	-	-
Qin et al. (2018)	LUNA16	888	-	98,30	-	-	-
IDF	LIDC-IDRI	833	97,73	98,40	95,87	0,941	0,996

-Negrito é o melhor resultado.

5.6 Conclusão

O emprego do índices de diversidade funcional na tarefa de redução de falsos positivos nas imagens de TC, especificamente, na caracterização das propriedades de textura das VOIs candidatas a nódulo pulmonar. O método proposto obteve-se resultados promissores, uma acurácia de 97,73%, área sob curva ROC de 94,10% e índice *Kappa* de 99,60%. Com base nestes resultados o método proposto foi superior ao estado da arte.

A metodologia apresentada neste trabalho poderá integrar-se a uma ferramenta CAD, sendo aplicada em casos reais e atuais na detecção de nódulos pulmonares, beneficiando assim o especialista que poderá utilizar a ferramenta para ter auxílio no diagnóstico, tendo assim, uma segunda opinião.

Como trabalhos futuros, pretende-se: usar a metodologia para diagnóstico dos nódulos em benigno e maligno; implantação de mais índices funcionais para caracterização de texturas em imagens; aplicação do método juntamente aos outros índices funcionais a diferentes tipos de bases de imagens

Metodologia automática para classificação de lesões de mama utilizando Transfer Learning

Autores: Antônio Oseas de Carvalho Filho; Rafael Pedro da Silva Neto.

Resumo: As lesões de Mama é um dos tipos de lesões mais comuns entre as mulheres no Brasil e no mundo, respondendo por cerca de 28% dos novos casos a cada ano. Estatísticas indicam aumento da sua incidência tanto nos países desenvolvidos quanto nos em desenvolvimento. Quando os pacientes são diagnosticados com a lesão, precisam saber se o mesmo mostra-se Benigno ou Maligno. Tal informação pode ser obtida por meio de exames de biopsia, que é um exame de diagnóstico no qual o médico retira um pedaço de tecido do interior da mama, geralmente de um nódulo, para avaliar no laboratório e observar se existem células lesionadas. Nesse trabalho, foi desenvolvido um método computacional para classificação de imagens diferenciando lesões malignas e benignas de mama. Para tal método, foram testadas diferentes arquiteturas de Redes Neurais Convolucionais (CNN). O Transfer Learning foi empregada para lidar com a limitação da pequena quantidade de imagens na base de dados. O conjunto de imagens utilizado foi o Digital Database for Screening Mammography (DDSM). Com base nos testes realizados, a ResNet foi a arquitetura que obteve o melhor resultado utilizando do classificador Suport Vector Machine (SVM) com kernel Polinomial, atingindo uma precisão de 81,73%, uma sensibilidade de 85,66%, uma especificidade de 78,40%, Kappa de 0,63 e curva ROC de 0,82. Finalmente, acreditamos que nosso método pode integrar uma ferramenta de CAD atuando como triagem de pacientes ou fornecendo uma segunda opinião ao especialista.

Keywords: Lesões de mama, Processamento digital de imagens, Transfer Learning.

6.1 Introdução

Atualmente, o câncer de mama é uma das principais ocorrências de câncer em mulheres com mais de 40 anos de idade em todo o mundo. Um diagnóstico precoce é muito importante, e um método comum de diagnóstico é o uso da mamografia, que é basicamente um raio-X da região da mama tirada em uma condição especial. A partir da imagem, um médico treinado o examina buscando microcalcificações e massas de várias formas. Se encontrados, esses artefatos na imagem podem ser um sinal da presença de um tumor benigno ou maligno (FERREIRA; BORGES, 2003).

Segundo Pontes (2013), a diferença entre benigno e maligno é definida pela aparência e estrutura das células atacadas pelo tumor. Os tumores benignos são constituídos por células bem semelhantes às que os originaram e não possuem a capacidade de provocar metástases. Já os malignos são agressivos e possuem a capacidade de infiltrar outros órgãos. A multiplicação celular é “controlada” pelos chamados oncôgenes ativadores, que regulam o crescimento e a morte da célula. No tumor benigno há uma mutação na estrutura genética dos oncôgenes, mas nada capaz de “descontrolá-la”. O mesmo não ocorre com os tumores malignos, que crescem sem controle por conta da alteração genética.

Ainda não é possível prevenir o câncer de mama em função da multiplicidade de fatores relacionados ao surgimento da doença, desta forma, o diagnóstico precoce é a melhor forma de combater a doença dentre os métodos para detecção e classificação de câncer de mama. Tramonte et al. (2016) aponta que o diagnóstico do câncer de mama, em países com recursos limitados, é frequentemente feito em um estágio já avançado.

Especificamente, durante o procedimento de diagnóstico, os especialistas avaliam a organização do tecido por meio de imagens microscópicas. No entanto, a grande quantidade de dados e a complexidade das imagens tornam essa tarefa custosa e não trivial. O método tradicional de inspeção microscópica é importantíssimo em analisar a estrutura do tecido. No entanto, esta técnica de inspeção manual é altamente demorada e propensa a erros manuais, principalmente no processo de classificação em lesões malignas ou benignas, pois, é nesse diagnóstico que é fornecido a informação de espalhamento de células cancerosas para os tecidos circundantes a partir de seu ponto de ocorrência (INCA, 2018a).

Os sistemas de detecção auxiliada por computador (*Computer-Aided Diagnosis - CAD*) foram desenvolvidos para reduzir os gastos e melhorar a capacidade de inúmeros profissionais que lidam com imagens. Pesquisas efetuadas mostraram que o uso de tais sistemas estão se tornando primordiais na resolução de diagnóstico médico por meio de imagens Elsheikh, Mustafa e Ibraheem (2018). Um problema relacionado à eles é seu alto custo computacional, além de ser preciso um longo tempo para trazer um resultado e necessitar de uma grande quantidade de dados para processamento.

6.1.1 Objetivo Geral

Em vista do contexto apresentado, este trabalho tem como objetivo geral desenvolver um método computacional de baixo custo computacional e eficiente para a classificação automática de imagens de lesões de mama em benignas e malignas em resolução ao problema de uma limitada base de imagens, reduzindo resultados falso positivo e falso negativo.

Para a concretização do objetivo geral, tem-se os seguintes objetivos específicos:

- Avaliar a viabilidade do uso das CNNs por meio do teste de diferentes arquiteturas de redes e o uso do Transfer Learning nas arquiteturas aplicadas ao problema descrito;
- Utilizar métodos de avaliação das lesões de mama utilizado pelos especialistas com intuito de empregar na etapa de validação dos resultados obtidos pelo método proposto;
- Construir uma ferramenta capaz de quantificar automaticamente regiões de lesões de mama.

6.1.2 Organização do capítulo

As próximas seções estão dispostas da seguinte forma: a Seção 6.2 contém, de forma breve, descrições dos principais trabalhos relacionados encontrados na literatura. Na Seção 6.3 apresenta os materiais e métodos abordados no trabalho, onde são explicados as etapas de desenvolvimento da pesquisa, iniciando pela aquisição das imagens, métricas de avaliação, CNNs e a descrição das arquiteturas e descritores comparados. A Seção 6.4 trata de referenciais teóricos para o entendimento do trabalho desenvolvido. A Seção 6.5 discute os resultados obtidos com a metodologia proposta. E por fim, na Seção 6.6 são apresentadas as conclusões e trabalhos futuros.

6.2 Trabalhos Relacionados

Estudos conhecidos sobre o problema da classificação de imagens vêm se tornando relevantes com o passar dos anos e se popularizando no meio científico. Com isso, a Tabela 28 mostra um resumo das principais informações dos trabalhos relacionados para a classificação de lesões de mama, em seguida, apresentamos um resumo dos trabalhos relacionados.

Tabela 28 – Comparação entre os trabalhos relacionados.

Trabalho	Base de Imagens	Algoritmos	Acurácia
(RAKHLIN et al., 2018)	ICIAR 2018	ResNet, VGG e Inception	87,2%
(GUO et al., 2018)	ICIAR 2018	GoogleNet	80,0%
(SILVA, 2016)	DDSM	LBP	90,18%
(AREVALO et al., 2016)	BCDR-FM	CNN, HOG e HGD	-

Em [Rakhlin et al. \(2018\)](#), foi proposto uma abordagem computacional baseada em redes neurais profundas, onde são utilizadas algumas arquiteturas de redes (ResNet, VGG e Inception) pré-treinadas na base ImageNet para a classificação de imagens histológicas de câncer de mama da base *International Conference on Image Analysis and Recognition* (ICCIAR) 2018.

[Guo et al. \(2018\)](#) propôs uma estratégia que utiliza CNN híbrida que é projetada com base na arquitetura GoogleNet para mesclar mais informações importantes na decisão, onde o *Transfer Learning* e aumento de dados foram empregados para lidar com as limitações da pequena quantidade de imagens na base ICIAR 2018.

Em [Silva \(2016\)](#), utilizando a base *Digital Database for Screening Mammography* (DDSM), foram combinadas abordagens estrutural e estatística para a análise de textura. A abordagem estrutural foi realizada por meio da técnica de Local Binary Patterns (LBP) e as abordagens estatísticas, através das técnicas da função K de Ripley e de índice de diversidade.

O método proposto por [Arevalo et al. \(2016\)](#), apresenta uma metodologia para classificação automática das lesões de imagem da mama utilizando representação do aprendizado. O método é avaliado de dados de benchmarking comprovado por biópsia 736 casos. O método desenvolvido compreende dois estágios principais: (i) pré-processamento para aprimorar os detalhes da imagem e (ii) treinamento supervisionado para aprender os recursos e a extração de características de lesões de imagem da mama usando *Histogram of Oriented Gradients* (HOG) e *Histogram of the Gradient Divergence* (HGD). Os resultados apresentam desempenho de 0,787 a 0,822 em termos da área sob a curva ROC.

A análise dos trabalhos encontrados contribuiu para identificar que as técnicas de aprendizagem profunda estão sendo bastante utilizadas em aplicações de imagens médicas. Também observou-se o uso de hardwares com grande capacidade de processamento para atingir os objetivos propostos por cada trabalho, por isso, foi investigado nesse trabalho o uso dessas técnicas com um hardware de baixo poder de processamento. Para contornar o problema da baixa quantidade de imagens e da limitação do hardware utilizado neste trabalho, foi utilizado o *Transfer Learning* que será abordada mais a frente. Com as pesquisas realizadas, viu-se necessário a utilização de classificadores externos para a validação. Foram então utilizados os classificadores Support Vector Machine (SVM), Random Forest e eXtreme Gradient Boost (XGBoost). Em seguida, veio a etapa de desenvolvimento e testes computacionais.

6.3 Materiais e Métodos

Nessa seção são apresentados a base de imagens, conceitos importantes para uma melhor compreensão deste trabalho, as principais arquiteturas de CNNs utilizadas para o processamento das imagens. Por fim são apresentadas as métricas de avaliação dos resultados obtidos.

A primeira etapa foi a coleta das imagens de duas classes de lesões de mama, malignas e benignas. Para compor a base de dados utilizada nesse trabalho, foram selecionadas imagens do arquivo da DDSM, por ser de fácil acesso e aberta ao público.

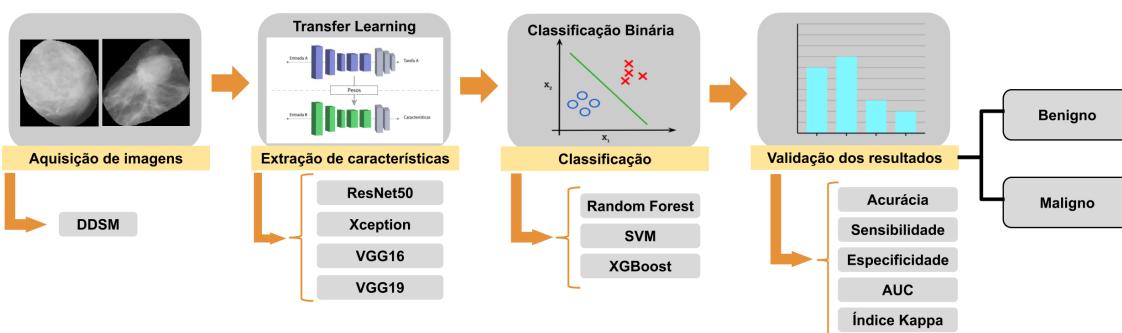
A segunda etapa consistiu em extraírem características dessas imagens utilizando as redes pré-treinadas com pesos da ImageNet, disponíveis na biblioteca Keras. Houve um processamento da base para alterar o tamanho de entrada dos dados para o formato exigido por cada uma das redes e, devido ao *Transfer Learning*, as suas camadas de classificação foram removidas.

Na terceira etapa, foram utilizados os classificadores: SVM, *Random Forest* e o XGBoost, com as características extraídas das arquiteturas para a validação. Nesta etapa, foi utilizado o método *Cross Validation*. Foram testados ao todo três kernels diferentes do SVM: Linear, Polynomial e RBF, numa tentativa de alcançar maiores resultados.

Por fim, na quarta e última etapa, para a avaliação do método proposto nesse trabalho foram utilizadas as métricas: Sensibilidade (S), Especificidade (E), Acurácia (A), Área Sob a Curva ROC (AUC) e o índice *Kappa* (K). Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em função de quatro valores, sendo eles: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

A Figura 49 descreve um resumo das etapas do método utilizado.

Figura 49 – Metodologia proposta.



6.3.1 Base de Dados

A DDSM ([HEATH et al., 2000](#)) é uma base de imagens públicas contendo 2.620 estudos/exames de mamografia digitais. Contém casos normais, benignos e malignos com informações patológicas confirmadas. A DDSM está disponível para uso da comunidade de pesquisa de análises mamográficas de imagens. O apoio primário para este projeto foi uma doação do Programa de Pesquisa do Câncer de Mama do Comando de Pesquisa Médica e Material do Exército dos EUA. A DDSM é mantida na Universidade do Sul da Flórida com o objetivo de mantê-lo acessível na web.

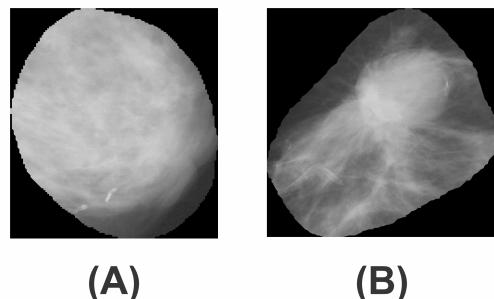
Todas os exames possuem marcações indicando os contornos das lesões. Estas marcações são realizadas por um conjunto de especialistas. Com isso, em nosso trabalho, utilizamos as marcações disponíveis para segmentar as *Region of Interest* (ROI) e com isso aplicar o método proposto. A Figura 50 mostra dois exemplos de imagens da base de dados da DDSM, uma de lesão benigna (A) e outra de lesão maligna (B) após segmentação e extração da ROI.

Tradicionalmente as mais variadas tecnologias são utilizadas para oferecer um ótimo desempenho no processamento de imagens e uma dessas está rapidamente se tornando um dos mais estudados e procurados campos dentro da ciência da computação moderna, onde ela oferece facilidades e poder computacional. Uma dessas tecnologias que foi abordada nesse trabalho é denominada *Artificial Neural Network* (ANN) e ela será apresentada a seguir.

6.4 Referencial Teórico

Esta seção aborda tópicos relevantes ao contexto deste trabalho: conceitos sobre ANN, seus métodos e arquiteturas avaliadas; conceitos sobre os classificadores utilizados e, por fim; as etapas de validação de

Figura 50 – Benigno (A) e Maligno (B).

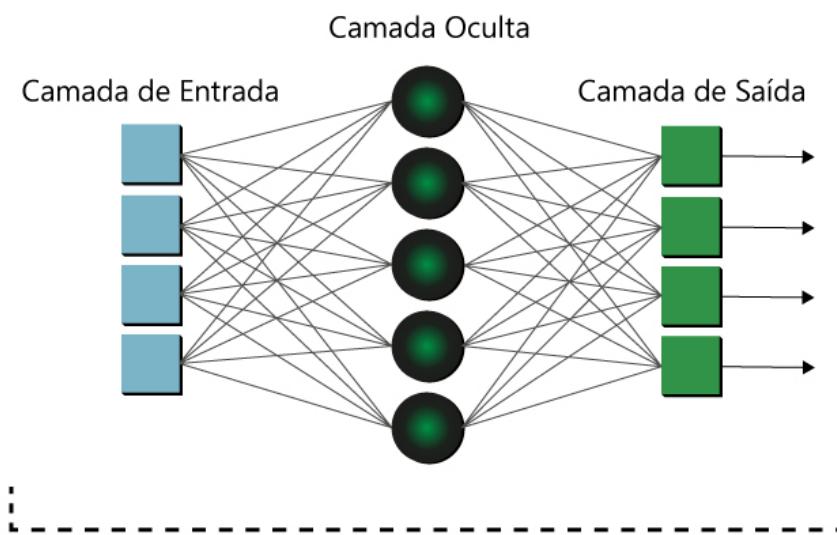


resultados.

6.4.1 Artificial Neural Network

As ANN's são redes computacionais que tentam, a grosso modo, simular o processo de decisão de uma rede de célula nervosa do sistema nervoso central biológico humano ou animal (DANIEL, 2013). São capazes de aprender à partir de um conjunto de dados, trabalhando de forma especializada com reconhecimento de padrões. Elas, semelhante ao sistema nervoso, possuem componentes lógicos (Neurônios Artificiais), organizados em um espécie de rede, que recebem entradas (características) e passam por uma camada oculta que decide quais as saídas mais adequadas a cada uma dessas entradas. A Figura 51 traz a ilustração de uma ANN.

Figura 51 – Ilustração de uma ANN.



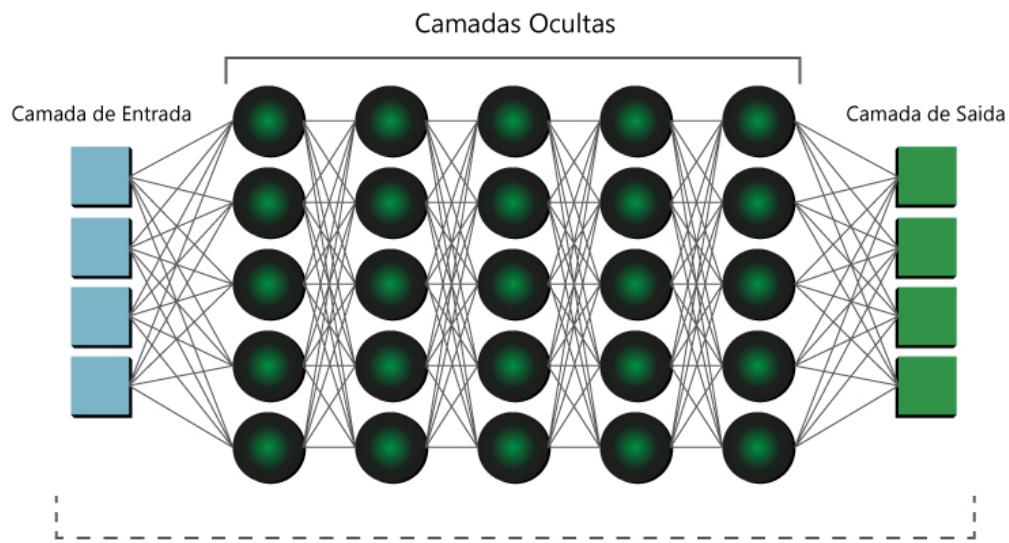
6.4.1.1 Deep Learning

A Aprendizagem Profunda (*Deep Learning* em inglês) é uma sub área do aprendizado de máquina que trata de modelos vagamente inspirados no cérebro humano. Tecnicamente, toda Rede Neural com mais de uma camada escondida é chamada de Rede Neural Profunda e tudo que ela aprende é por meio do aprendizado profundo. Para ??), o *Deep Learning* é uma área que contém uma pilha de multicamadas simples, todos (ou a

maioria) estão sujeitos à aprendizagem, e muitos dos quais calcula mapeamentos de entrada-saída não lineares.

A profundidade do modelo é determinada pela quantidade de camadas ocultas. Os modelos de *Deep Learning* atuais podem conter dezenas ou até centenas de camadas ocultas e todas são aprendidas automaticamente necessitando apenas da inserção dos dados de treinamento. A Figura 52 nos mostra uma ilustração de uma rede de *Deep Learning*.

Figura 52 – Ilustração de uma rede de *Deep Learning*.



Tradicionalmente, a qualidade dos algoritmos depende muito da representação dos dados em características. Assim, a criação recursos validos e pré-processamento consomem grande parte dos esforços dos especialistas. Já os algoritmos do tipo Aprendizado Profundo têm uma abordagem inovadora, pois dispensam grande parte desse pré-processamento e geram automaticamente propriedades invariantes em suas camadas hierárquicas de representação ([HELABS, 2019](#)).

[Brigade \(2016\)](#) afirma que atualmente o reconhecimento de imagens por máquinas treinadas através de *Deep Learning* em alguns cenários possuem uma taxa de acerto maior que a de humanos. Várias de suas arquiteturas têm sido aplicadas em áreas como visão computacional, reconhecimento automático de fala, processamento de linguagem natural, reconhecimento de áudio e bioinformática, onde elas têm se mostrado capazes de produzir resultados significantes.

Ao passar dos anos surgiu uma categoria de ANN baseada em *Deep Learning* que inovaram por utilizar convolução em algumas de suas camadas. Essa categoria que passou a ser o novo padrão de visão computacional, é chamada de Convolutional Neural Network (CNN), que será descrita a seguir.

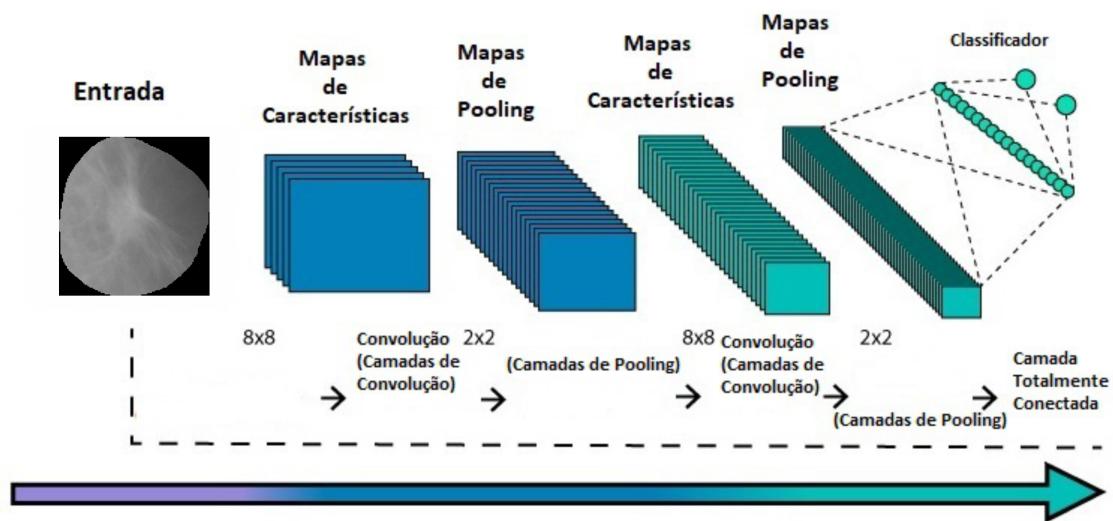
6.4.1.2 Convolutional Neural Network

As CNN's ou ConvNets são uma categoria de redes neurais usadas no reconhecimento e processamento de imagens que é especificamente projetada para processar dados de pixel. [PONTI et al. \(2018\)](#) versa que sua característica marcante é ser composta basicamente de camadas convolucionais, que processa as entradas considerando campos locais. Essas redes constituem uma escolha de topologia ou arquitetura projetadas para reduzir o número de parâmetros a serem aprendidos otimizando o tempo de treinamento.

As CNNs são projetados para processar dados que vêm na forma de múltiplos arrays, por exemplo, uma imagem colorida composta por três matrizes contendo intensidades de pixel nos três canais de cores (??). Elas são capazes de capturar com sucesso as dependências espaciais e temporais em uma imagem através da aplicação de filtros. [Kovaleski \(2018\)](#) afirma que a mesma se utiliza desses filtros para analisar pequenos

grupos de dados, extraiendo suas características mais relevantes e relata que apenas estas características são passadas adiante para as próximas camadas, como apresentado na Figura 53.

Figura 53 – Ilustração da arquitetura de uma CNN.



Ao contrário de uma rede neural regular, as camadas de uma CNN possuem neurônios dispostos em 3 dimensões: largura, altura e profundidade. Além disso, os neurônios em uma camada não se conectam a todos os neurônios da camada seguinte, mas apenas a uma pequena região dela. Por fim, a saída final será reduzida a um único vetor de pontuações de probabilidade, organizadas ao longo da dimensão de profundidade.

A arquitetura executa um melhor ajuste ao conjunto de dados da imagem devido à redução no número de parâmetros envolvidos e à capacidade de reutilização dos pesos. A grande vantagem da aplicação das CNNs para o Processamento Digital de Imagens (PDI) é sua grande capacidade de extrair características, por trabalhar com convoluções, ou seja, criando “sub-quadros de bits” de forma que seja possível realçar contornos e áreas de interesse de cada imagem (ROSEBROCK, 2017b).

Quanto maior a quantidade de amostras (base de dados) que representam as diferentes classes, mais fácil será o treinamento de uma CNN. Mas, nem sempre é possível utilizar grandes quantidades de imagens devido a empecilhos como barreiras de acesso a essas bases ou até mesmo limitação do hardware, o que atrapalha o uso dessa ferramenta. Para contornar esse problema, foi desenvolvido uma técnica chamada de *Transfer Learning*, que será descrita a seguir.

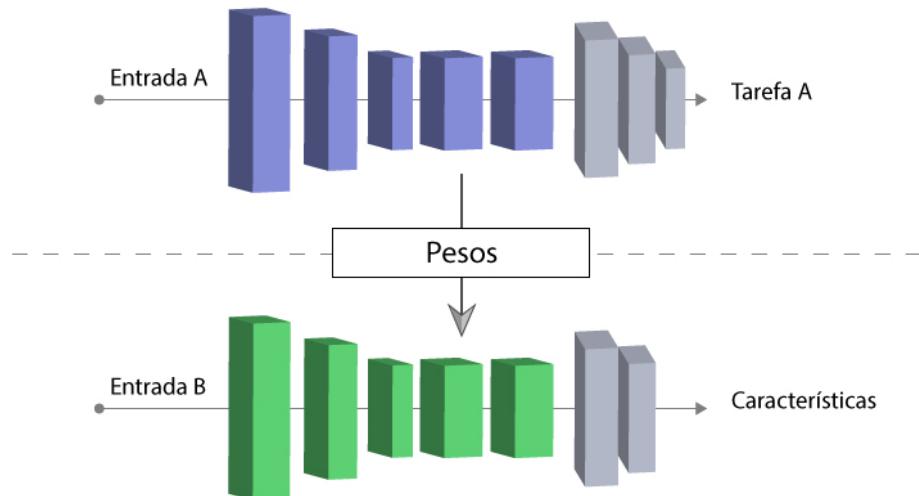
6.4.2 Transfer Learning

O *Transfer Learning* é um método de aprendizado de máquina em que um modelo desenvolvido para uma tarefa é reutilizado como ponto de partida para um modelo em uma segunda tarefa (PONTI et al., 2018). É uma abordagem popular em Aprendizado Profundo na qual os modelos pré-treinados são usados como fase inicial em tarefas de visão computacional.

Normalmente, precisa-se de muitos dados para treinar uma Rede Neural, mas nem sempre é possível ter acesso a dados suficientes. É aí que o *Transfer Learning* entra em jogo, porque com ela pode-se construir um modelo de aprendizado de máquina sólida com comparativamente poucos dados de treinamento, porque o modelo já é pré-treinado (TORREY; SHAVLIK, 2010).

Embora as redes profundas tenham mostrado-se promissoras na representação de informações, tais modelos profundos supervisionados, exigem grandes quantidades de dados rotulados e recursos de computação, que são caros (DU et al., 2017b). O processo de *Transfer Learning* é demonstrado pela Figura 54.

Figura 54 – Ilustração do Transfer Learning.



Portanto, o *Transfer Learning* também economiza muito tempo de treinamento, pois pode-se levar dias ou até semanas para treinar uma Rede Neural profunda do zero em uma tarefa complexa e, além desse benefício, resulta em menor erro de generalização (GOODFELLOW; BENGIO; COURVILLE, 2016).

A seguir serão abordados as arquiteturas utilizadas que foram avaliados neste trabalho.

6.4.3 Arquitetura das Redes Avaliadas

Após analisar a bibliografia que resultou da pesquisa a respeito do tema, foram encontrados diversos modelos de arquitetura compatíveis com a proposta desse trabalho. Com isso, foi possível selecionar três principais modelos para estudo mais profundo e consequente avaliação da melhor performance. Tais modelos são descritos à seguir.

6.4.3.1 Modelo 1 - ResNet

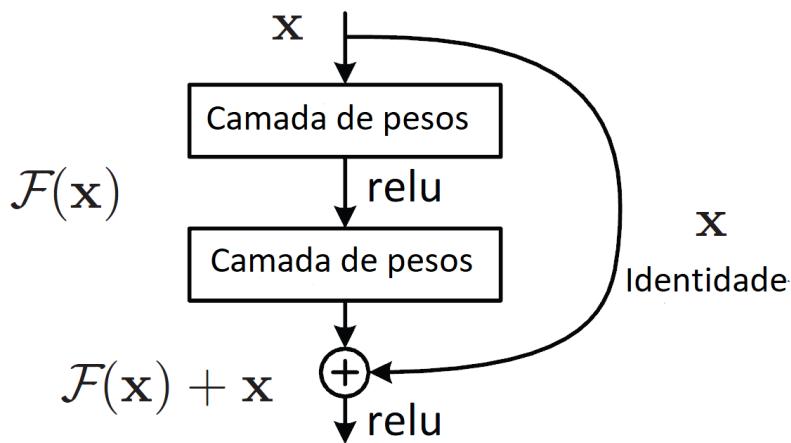
A ResNet é uma arquitetura que tem como principal característica suas conexões residuais. Na aprendizagem residual, em vez de tentar aprender algumas características, tal arquitetura é voltada para aprender alguns resíduos. Residual pode ser simplesmente entendido como subtração de características aprendidas da entrada dessa camada e a ResNet faz isso usando conexões de atalho, que são conexões em que o gradiente é diretamente retro propagado para camadas anteriores. Na Figura 55 é demonstrado o funcionamento de um bloco residual.

Um dos problemas que a ResNet resolve é o famoso desaparecimento de gradiente. Isso ocorre porque, quando a rede é muito profunda, os gradientes de onde a função de perda é calculada diminuem facilmente para zero após a aplicação da regra da cadeia por várias vezes. Este resultado nos pesos nunca atualiza seus valores e, portanto, nenhum aprendizado será realizado. Com tal arquitetura, os gradientes podem fluir diretamente através das conexões de salto. Para Chen et al. (2017) a ResNet fornece uma estrutura de treinamento para facilitar formação de redes substancialmente mais profundas que as utilizadas anteriormente.

6.4.3.2 Modelo 2 - VGG

A VGG é um modelo de CNN inicialmente proposto por Simonyan e Zisserman (2014a). A mesma garantiu a primeira e segunda posições nas tarefas de localização e classificação, respectivamente, no desafio ImageNet 2014. Ela faz a melhoria sobre o AlexNet construída por Krizhevsky, Sutskever e Hinton (2012) substituindo

Figura 55 – Ilustração da arquitetura da rede ResNet.



grandes filtros do kernel por múltiplos filtros do tamanho 3x3 um após o outro. Nela as profundidades mais comuns são as de 16 e 19 camadas de profundidade.

Com a introdução da VGG, elas trouxeram alguns padrões: sugeriu-se que todos os filtros tivessem tamanho de 3x3, pools máximos deveriam ser colocados após cada 2 convoluções. A VGGNet é considerada a rede preferida pela comunidade para *Transfer Learning*, pois sua arquitetura uniforme é boa para extrair características de imagens. A Figura 56 nos mostra uma ilustração da arquitetura utilizada na VGG.

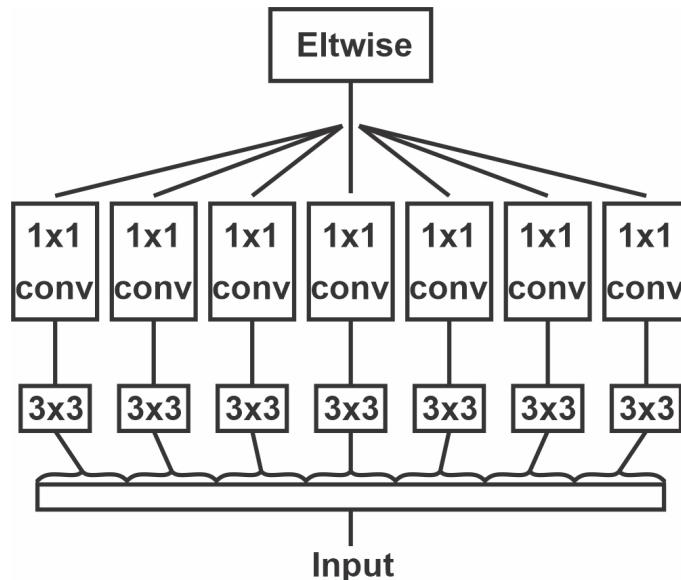
Figura 56 – Ilustração da arquitetura da rede VGG.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
FC-4096					
FC-4096					
FC-1000					
soft-max					

6.4.3.3 Modelo 3 - Xception

A arquitetura Xception foi proposta por ([CHOLLET, 2017](#)) e significa *Extreme version of Inception* sendo uma versão da arquitetura Inception, onde essa arquitetura tem 36 camadas convolucionais formando a base de extração de características da rede. A sua principal característica é que ela substitui o módulo Inception por convoluções separáveis em profundidade, isso faz com que o número de conexões sejam menores e o modelo se torne mais leve. A Figura [57](#) demonstra um módulo da rede Xception.

Figura 57 – Ilustração da arquitetura da rede Xception.



A Figura [57](#) mostra que a arquitetura Xception em vez de partitionar dados de entrada em vários blocos compactados, a mesma leva isso um passo adiante, ela mapeia as correlações espaciais para cada canal de saída separadamente e, em seguida, executa uma convolução de 1x1 em profundidade para capturar a correlação entre canais.

6.4.4 Classificação

A classificação consiste em reconhecer novos objetos de um conjunto de recursos extraídos de um grupo de objetos de interesse e, em seguida, treinar um sistema para reconhecer novos objetos desses recursos ([AZEVEDO; CONCI, 2008](#)). Para esta classificação, foram utilizados: SVM, Random Forest e Xgboost.

6.4.4.1 Support Vector Machine

É um algoritmo de classificação de aprendizado de máquina supervisionado proposto por [Vapnik \(2013\)](#), [Vapnik \(1998\)](#). Neste trabalho, usamos o SVM com o kernel da função base radial (RBF), parâmetro gama = 10 e C = 10000. O valor de C foi escolhido alto para que uma pequena margem fosse aceita pela função de decisão na classificação correta de todos os pontos. O parâmetro gama foi escolhido baixo, de modo que um único exemplo teve pouca influência no treinamento de dados.

6.4.4.2 Random Forest

É uma combinação de múltiplas previsões de árvores em que cada árvore depende dos valores de um vetor independente, sorteado aleatoriamente e com a mesma distribuição para todas as árvores da floresta ([WRIGHT; WAGER; PROBST, 2016](#)). Para este trabalho, o Random Forest foi usado com os seguintes parâmetros: bag Size Percent = 100, batch Size = 100, número de execuções slots = 1, max Depth = 0 (ilimitado), número de atributos escolhidos aleatoriamente = 0, número de iterações a ser realizado = 100,

número mínimo de instâncias por folha = 1,0, variação mínima para divisão = 0,001 e o número aleatório semente a ser usado = 1.

6.4.4.3 *eXtreme Gradient Boost*

É uma biblioteca proposta por [Chen et al. \(2015\)](#) com base na estrutura de aumento de gradiente desenvolvida por [Friedman \(2001\)](#), [Friedman, Hastie e Tibshirani \(2000\)](#). O Xgboost pode ser usado para várias funções objetivas, como regressão e classificação ([SONG et al., 2016](#)). O Xgboost é extensível e seus parâmetros podem ser facilmente alterados. Os parâmetros utilizados no XGBoost foram os seguintes: profundidade máxima = 7, taxa de aprendizado = 0,1, ite = 1000, gama = 0, passo delta máximo = 1 e objetivo = "multi:softmax".

6.4.5 Métricas de Avaliação

Para a avaliação do método proposto nesse trabalho foram utilizadas as seguintes métricas: Sensibilidade (S), Especificidade (E), Acurácia (A), Área Sob a Curva ROC (AUC) e *Kappa* (K). Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em função de quatro valores, sendo eles: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

6.4.5.1 Acurácia

Esta é a métrica mais simples. Consiste basicamente o número de acertos (positivos) dividido pelo número total de exemplos. Essa métrica é mais apropriada para bases com a mesma proporção de exemplos para cada classe, e quando as penalidades de acerto e erro para cada classe forem as mesmas. Em problemas com classes desproporcionais, ela pode causar uma falsa impressão de bom desempenho. A acurácia pode ser calculada conforme a Equação [6.1](#).

$$A = \frac{VP + VN}{VP + VN + FP + FN} \quad (6.1)$$

6.4.5.2 Sensibilidade

A sensibilidade é uma medida da proporção de casos positivos reais que foram previstos como positivos (ou positivos verdadeiros). Em outras palavras, um teste altamente sensível é aquele que identifica corretamente pacientes com uma doença, o mesmo pode ser útil para descartar uma doença se uma pessoa tiver um resultado negativo. A sensibilidade pode ser calculada conforme a Equação [6.2](#)

$$S = \frac{VP}{VP + FN} \quad (6.2)$$

6.4.5.3 Especificidade

A especificidade de um teste (também chamado de Taxa Negativa Verdadeira) é a proporção de pessoas sem a doença que terão um resultado negativo. Em outras palavras, a especificidade de um teste se refere a quanto bem um teste identifica pacientes que não têm uma doença. A especificidade pode ser calculada com a Equação [6.3](#)

$$E = \frac{VN}{VN + FP} \quad (6.3)$$

6.4.5.4 *Kappa*

O coeficiente *Kappa* é uma medida de concordância estatística formulada por Jacob Cohen, tal métrica mede concordância entre dois avaliadores que classificam cada N itens em C categorias mutuamente exclusivas. O

Kappa de Cohen é forte dependente de vários fatores, incluindo, por exemplo, número de classes e número de exemplos a serem classificados. O *Kappa* pode ser calculado conforme a Equação 6.4

$$K = \frac{\theta_1 - \theta_2}{1 - \theta_2}, \quad (6.4)$$

onde

$$\theta_1 = \frac{VP + VN}{VP + VN + FP + FN}, \quad (6.5)$$

e

$$\theta_2 = \frac{[(VP + FN)(VP + FP)] + [(VN + FN)(VN + FP)]}{(VP + VN + FP + FN)^2} \quad (6.6)$$

Sobre a avaliação dos classificadores, Landis e Koch (1977b) partitionaram o nível de exatidão do Índice *Kappa* conforme a Tabela 29.

Tabela 29 – Nível de exatidão de uma classificação, conforme o valor de índice *Kappa*

Índice <i>Kappa</i> (K)	Qualidade
$K < 0.2$	Ruim
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito Boa
≥ 0.8	Excelente

6.4.5.5 Área Sob a Curva ROC

É uma métrica para classificação binária, é provavelmente uma das mais populares. Braga (2001) afirma que a curva ROC é um método gráfico simples e robusto, permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. AUC é uma abreviação da área sob a curva. É usado na análise de classificação para determinar qual dos modelos utilizados prediz melhor as classes. Quanto mais próximo o AUC de um modelo chegar a 1, melhor ele será. Assim, os modelos com AUCs mais altos são preferidos em relação àqueles com AUCs mais baixos.

Após apresentar fundamentações, métodos e técnicas ao longo desse trabalho, a seguir são descritos os resultados obtidos ao fim das etapas de treinamento e testes avaliativos, além disso será abordado uma discussão sobre o processo utilizado para resolver o problema proposto.

6.5 Resultados e Discussões

Esta Seção mostra e discute os resultados obtidos com a metodologia proposta por esse trabalho, obtendo assim, a classificação das regiões de interesse de imagens de lesões de mama. Além disso, ele apresenta o comparativo das arquiteturas de redes convolucionais apresentadas na seção 6.4.3.

6.5.1 Treinamento e Avaliação

Nesse trabalho, propomos um método automático para classificação de imagens de lesões de mama em duas classes (benigno e maligno) que foram coletadas da base DDSM usando abordagens de *Transfer Learning*, sendo 1155 ROIs, compostas de 530 lesões benignas e 625 malignas.

Foram utilizadas redes pré-treinadas com pesos da ImageNet (DENG et al., 2009b) para extração das características dessas imagens. No processo final de classificação é utilizado os classificadores SVM com kernels (Linear, RBF e Polynomial), Random Forest e o XGBoost, conforme descrito na Seção 6.4.4.

O método de avaliação empregado nesse trabalho é o *Cross Validation*, onde consiste em dividir a base de imagens em k partes, usando $k-1$ partes para treino e a parte remanescente para teste, fazendo isso k vezes. Também foi feito a concatenação de todas as características extraídas pelos descritores para classificação, e

por fim, fazemos um último teste que conta com a seleção de atributos utilizando o *Greed StepWise* (CARUANA; FREITAG, 1994).

6.5.2 Testes Avaliativos

Para a realização dos experimentos aqui descritos, utilizou-se um computador, cujas especificações são descritas na Tabela 30.

Tabela 30 – Descrição do Ambiente de Desenvolvimento

Características	Descrição
Modelo	Acer Aspire VX5-591G-54PG
RAM	8,00 GB
Processador	Intel Core i5-7300HQ 2.5 GHz
Sistema Operacional	Windows 10

Nesse trabalho foram empregados dois cenários de testes. No primeiro cenário, temos os resultados com todas as características dos modelos. Já no segundo cenário, temos os resultados com a seleção de atributos feitas com *Greedy StepWise*.

A seguir, os resultados obtidos com a metodologia proposta serão apresentados e discutidos. As métricas nas tabelas são: Acurácia (A), Sensibilidade (S), Especificidade (E), *Kappa* (K) e área sob a curva ROC (AUC). Os melhores resultados obtidos em cada método estão destacados em negrito.

Tabela 31 – Resultados da rede Resnet.

Rede	Classificador	A(%)	S(%)	E(%)	K	AUC
Resnet50 Sem seleção 2048 features	RF	79,48	81,88	77,44	0,58	0,85
	SVM LINEAR	78,70	77,54	79,68	0,57	0,78
	SVM RBF	81,38	85,47	77,76	0,62	0,81
	SVM POLY	81,73	85,66	78,40	0,63	0,82
	XGBoost	77,84	80,84	75,84	0,56	0,78
ResNet50 Com seleção 137 features	RF	78,00	81,13	75,36	0,56	0,85
	SVM LINEAR	79,22	83,20	75,84	0,58	0,79
	SVM RBF	76,71	79,62	74,24	0,53	0,76
	SVM POLY	78,96	85,47	73,44	0,58	0,79
	XGBoost	75,06	78,11	72,48	0,50	0,75

Tabela 32 – Resultados da rede Xception.

Rede	Classificador	A(%)	S(%)	E(%)	K	AUC
Xception Sem seleção 2048 features	RF	78,09	80,18	76,32	0,56	0,85
	SVM LINEAR	76,36	76,41	76,32	0,52	0,76
	SVM RBF	79,91	83,58	76,80	0,59	0,80
	SVM POLY	80,00	83,96	76,64	0,60	0,80
	XGBoost	76,28	78,11	74,72	0,53	0,76
Xception Com seleção 135 features	RF	77,66	80,37	76,96	0,55	0,85
	SVM LINEAR	78,44	82,26	75,20	0,56	0,78
	SVM RBF	76,96	85,47	69,76	0,54	0,77
	SVM POLY	79,22	84,52	74,72	0,58	0,79
	XGBoost	75,50	77,74	73,60	0,55	0,76

No primeiro cenário, foi obtido o melhor resultado com a ResNet50 com classificador SVM Polynomial, com acurácia de 81.73%, sensibilidade de 85.66%, especificidade de 78.40%, Kappa de 0.63 e curva ROC de 0,82. No segundo cenário, com seleção de atributos, não houveram melhorias nos resultados, sendo o

Tabela 33 – Resultados da rede VGG16.

Rede	Classificador	A(%)	S(%)	E(%)	K	AUC
VGG16 Sem seleção 4096 features	RF	78,44	85,09	72,80	0,57	0,85
	SVM LINEAR	78,70	79,05	77,92	0,57	0,78
	SVM RBF	79,65	86,98	73,44	0,59	0,80
	SVM POLY	79,91	88,49	72,64	0,60	0,80
	XGBoost	74,81	78,30	71,84	0,50	0,75
VGG16 Com seleção 120 features	RF	77,57	83,96	72,16	0,55	0,84
	SVM LINEAR	78,78	87,73	71,20	0,57	0,79
	SVM RBF	76,27	89,24	65,28	0,53	0,77
	SVM POLY	78,87	90,00	69,44	0,58	0,79
	XGBoost	74,72	80,38	69,92	0,50	0,75

Tabela 34 – Resultados da rede VGG19.

Rede	Classificador	A(%)	S(%)	E(%)	K	AUC
VGG19 Sem seleção 4096 features	RF	78,18	83,39	73,76	0,56	0,85
	SVM LINEAR	75,75	75,09	76,32	0,51	0,75
	SVM RBF	79,48	88,22	73,76	0,59	0,80
	SVM POLY	79,13	86,79	72,64	0,58	0,79
	XGBoost	76,10	79,06	73,60	0,52	0,76
VGG19 Com seleção 113 features	RF	79,30	85,47	74,08	0,58	0,84
	SVM LINEAR	77,57	83,96	72,16	0,55	0,78
	SVM RBF	76,36	83,01	70,72	0,53	0,76
	SVM POLY	78,00	86,98	70,40	0,56	0,78
	XGBoost	74,89	79,25	71,20	0,50	0,75

Tabela 35 – Resultados de todas as features concatenadas.

Rede	Classificador	A(%)	S(%)	E(%)	K	AUC
Concatenadas Sem seleção 12288 features	RF	78,78	81,88	76,16	0,57	0,85
	SVM LINEAR	77,48	76,41	78,40	0,54	0,77
	SVM RBF	80,80	80,62	79,42	0,60	0,80
	SVM POLY	81,29	85,47	77,76	0,62	0,81
	XGBoost	80,87	82,83	79,20	0,62	0,81
Concatenadas Com seleção 244 features	RF	80,51	84,15	77,44	0,61	0,79
	SVM LINEAR	80,51	84,71	76,96	0,61	0,80
	SVM RBF	80,34	86,60	75,04	0,60	0,80
	SVM POLY	81,38	86,60	76,96	0,62	0,81
	XGBoost	81,13	82,83	79,68	0,62	0,81

melhor com a VGG19 com classificador *Randon Forest*, com com acurácia de 79.30%, sensibilidade de 85.47%, especificidade de 74.08%, Kappa de 0.58 e curva ROC de 0,84.

Tendo em vista todos os testes realizados, nota-se que o modelo ResNet50 obteve melhores resultados na maioria dos casos, assim como o classificador SVM com kernel Polynomial, que obteve melhores resultados que os demais classificadores. Um ponto de destaque é o valor do *Kappa*, pois com base na Tabela 29 o algoritmo utilizado tem um nível muito bom de exatidão na classificação. Os demais resultados obtidos foram satisfatórios de acordo com as métricas especificadas, indicando uma boa capacidade discriminativa e classificatória do algoritmo para as classes definidas.

A comparação com outros trabalhos dentro da literatura é uma tarefa um pouco complicada, já que há uma algumas diferenças nas maneiras de abordagem do problema e também nas ferramentas e técnicas utilizadas para solucioná-los. A Tabela 36 apresenta um resumo quantitativo em relação aos trabalhos apresentados na Seção 6.2.

Em comparação com os demais trabalhos, este trabalho se mostra eficiente no uso do Transfer Learning

Tabela 36 – Tabela de comparação dos resultados

Trabalho	A(%)	S(%)	E(%)	K	AUC
(RAKHLIN et al., 2018)	93,80	96,50	88,00	–	97,3
(GUO et al., 2018)	80,00	–	–	–	–
(SILVA, 2016)	90,18	–	–	–	–
(AREVALO et al., 2016)	–	–	–	–	0,82
Esse trabalho	81,73	85,66	78,40	0,63	0,82

na extração de características da base DDSM, visto que o resultado ultrapassou em acurácia o trabalho de Guo et al. (2018) e se igualou na métrica de AUC conseguido no trabalho de Arevalo et al. (2016).

Por fim, o método se mostra promissor, apesar de não alcançar resultados que excedam o estado da arte, foram aplicadas diferentes estratégias e arquiteturas que possibilitam estudos futuros com base nas experiências apresentadas em neste trabalho.

6.6 Conclusão

Neste trabalho foi avaliado e comparado o desempenho de quatro arquiteturas de redes utilizando *Transfer Learning* para a extração de características e três classificadores para a classificação automática de lesões de mama em imagens médicas.

A arquitetura ResNet50 atingiu o melhor resultado dentre os algoritmos usados neste trabalho, alcançando valores acima de 78% nas métricas de avaliação, com a utilização do classificador SVM com kernel Polynomial. Os bons resultados na avaliação fornecem oportunidades para propor novas metas de pesquisa na área de lesões de mama.

As arquiteturas de redes investigadas transferiram com sucesso o conhecimento do ImageNet codificados como características convolucionais para o problema na presença de dados limitados de treinamento.

O trabalho apresentado mostra uma boa classificação no uso da *Transfer Learning* evidenciando uma eficiente ferramenta para a análise automática de imagens, mesmo com um número limitado de imagens e recursos computacionais escassos.

Desta forma, este trabalho apresenta contribuições em duas áreas de estudo. Na área da saúde, oferecendo uma metodologia como segunda opinião a profissionais da área médica para auxiliar no diagnóstico de lesões de mama, e na área da computação, fornecendo informações importantes para a área de processamento de imagens, como forma de auxiliar pesquisas futuras e também de terceiros.

Como trabalhos futuros, pretende-se:

1. Utilizar outras arquiteturas de CNN como *Transfer Learning* para comparação de resultados;
2. Utilizar outros classificadores; e,
3. Usar métodos automáticos para estimar parâmetros para classificação, afim de melhora os resultados obtidos.

Método Automático para Classificação de Imagens Histopatológicas da Mama Utilizando Transfer Learning e Análise de Textura

Autores: Antônio Oseas de Carvalho Filho; Milton Martins de Oliveira Junior.

Resumo: O câncer de mama é uma doença causada pela multiplicação desordenada de células mamárias anormais, formando massas. Essa doença é uma das principais causas de mortalidade entre as mulheres. A detecção precoce é essencial para que o tratamento comece o mais cedo possível, proporcionando melhores taxas de sobrevivência entre pacientes. A triagem de doenças através de imagens histopatológicas pode ser um meio promissor para a discriminação do tecido examinado. Nos últimos anos, vêm crescendo o número de sistemas de Detecção Auxiliada por Computador (CAD). Estes sistemas sofisticados são cruciais para complementar as análises feitas pelos especialistas. Neste trabalho, apresentaremos uma abordagem computacional para classificar tecidos mamários normais ou tumorais por meio de imagens histopatológicas. Foi utilizado o *dataset* fornecido pela Breast Cancer Histology ICIAR 2018, que contém 400 imagens de biópsias da mama. Houve a extração de características utilizando abordagens de *Transfer Learning* (ResNet50, VGG16 e VGG19) e índices de diversidade atuando como descritores de textura, como também a concatenação dos melhores conjuntos. Os atributos extraídos foram utilizados para classificação nos algoritmos *Random Forest* e *Support Vector Machine*. Os resultados alcançaram uma acurácia de 91,75%, mostrando ser uma metodologia promissora para a classificação de tumores e câncer de mama.

Keywords: Câncer de Mama. Imagens Histopatológicas. Índices de Diversidade. *Transfer Learning*.

7.1 Introdução

O Câncer de mama trata-se de uma doença causada pela multiplicação desordenada das células mamárias. Isto acaba acarretando na geração de células anormais, que quando se multiplicam, transformam-se em

um tumor.

Segundo o Instituto Nacional do Câncer (INCA), o câncer de mama é o mais comum entre as mulheres no mundo e no Brasil, depois do de pele não melanoma, correspondendo a cerca de 25% dos casos novos a cada ano (INCA, 2019).

A detecção do câncer de mama pode ocorrer nas fases iniciais e na maioria dos casos, isso possibilita um tratamento menos agressivo ao paciente. Os sintomas da doença devem ser investigados para que ocorra a confirmação do diagnóstico. A identificação do câncer de mama geralmente consiste em exames por palpação, mamografia e ultra-sonografia. Após o exame, é realizada a biópsia (quando necessário), do tecido mamário para determinar se o diagnóstico indica um crescimento maligno de tecido. Por meio das biópsias tornam-se viável avaliar a histologia microscópica e os elementos do tecido. Assim, pode distinguir o tecido normal de lesões benignas e malignas.

O tecido coletado durante a biópsia geralmente é corado com hematoxilina e eosina (H&E) antes da análise visual realizada pelos especialistas. Segundo Elmore et al. (2015), o processo de diagnóstico usando biópsias coradas com H&E não é uma atividade simples e a concordância diagnóstica média entre especialistas é de aproximadamente 75%.

Nos dias atuais, é crescente o interesse no desenvolvimento e aplicação de técnicas de processamento de imagens, onde os principais objetivos são aumentar a precisão e rapidez do diagnóstico e fornecer uma segunda opinião aos médicos. Dentro das técnicas de Processamento Digital de Imagens (PDI), as soluções de aprendizado profundo tem se mostrado muito eficazes para resolução de problemas semelhantes.

As técnicas de PDI tem mostrado robustez para análise e interpretação de imagens médicas (LEDERLIN et al., 2013b). Este auxílio ao especialista vem se tornando crucial para que se obtenha um diagnóstico preciso o mais rápido possível. O desenvolvimento dos métodos automáticos tem sido cada vez mais explorados, pois possibilitam detectar áreas suspeitas de anomalias.

O aprendizado profundo se mostra robusto e eficiente em descobrir estruturas complexas em dados de alta complexidade e, portanto, pode ser aplicado em vários contextos distintos. Os métodos de *deep learning* são capazes de transformar representações brutas em representações mais abstratas (??).

Nos últimos anos, houve um crescente interesse no desenvolvimento e uso de técnicas de processamento de imagens, onde os principais objetivos são aumentar a precisão do diagnóstico e fornecer uma segunda opinião aos médicos. Como alternativa para auxiliar no diagnóstico por imagem, entram sistemas como Detecção Auxiliada por computador (CAD) e Diagnóstico Auxiliado por Computador (CADx) (OLIVEIRA et al., 2015a), responsáveis por realizar detecção e diagnóstico de doenças de forma automática, respectivamente.

No presente trabalho, é proposta uma abordagem computacional para realizar a classificação de imagens de biópsia identificando se as mesmas possuem ou não tecidos cancerígenos. O intuito deste método é fornecer uma segunda opinião aos especialistas buscando tornar o diagnóstico mais preciso, precoce e de baixo custo.

Para viabilizar o desenvolvimento da metodologia proposta, aplicaremos a técnica de *Transfer Learning*, que consiste em utilizar redes neurais convolucionais pré-treinadas. Desta maneira, é possível utilizar o conhecimento já adquirido por elas para a extração de características. Além disso, será utilizada a análise de textura, que será feita por meio de índices de diversidade ecológicos. Por fim, utilizamos os classificadores *Random Forest* e *Support Vector Machine* (SVM) para realizar a classificação.

Como contribuição apresentamos um método totalmente automático para classificação de imagens histopatológicas em cancerígenas e saudáveis. No âmbito da computação apresentamos novas técnicas que são capazes de mensurar propriedades de textura das imagens, que quando utilizadas em conjunto com a técnica de *Transfer Learning* possam fornecer melhores resultados para o especialista ou atuar na triagem de pacientes.

7.1.1 Objetivos

7.1.1.1 Objetivo Geral

A partir do contexto apresentado, o presente trabalho possui como objetivo principal desenvolver um método automático para a classificação de imagens histopatológicas da mama em doentes e saudáveis.

7.1.1.2 Objetivos Específicos

1. Avaliar o desempenho da extração de características utilizando *Transfer Learning*;
2. Avaliar a viabilidade do uso de índices de diversidade como forma de extração de características;
3. Realizar a concatenação de características para alcançar melhores resultados;
4. Testar o desempenho dos classificadores SVM e *Random Forest* nas características extraídas; e
5. Construir uma metodologia que forneça uma segunda opinião ao especialista no diagnóstico do câncer de mama.

7.1.2 Organização do capítulo

Além da Introdução, este trabalho está dividido na seguinte ordem: na Seção 7.2, é apresentado o referencial teórico; na Seção 7.3, apresentam-se os trabalhos relacionados; na Seção 7.4, descreve-se a metodologia do trabalho; na Seção 7.5, são descritos os resultados e discussões da execução do projeto; e por fim, são apresentadas as conclusões e sugestões para trabalhos futuro na Seção 7.6.

7.2 Referencial Teórico

Com o propósito de facilitar o entendimento do trabalho aqui proposto, serão abordados nesta Seção conceitos, conhecimentos e informações pertinentes ao trabalho.

7.2.1 Câncer de Mama

O Câncer de mama é a patologia maligna de maior mortalidade na população feminina ([BUITRAGO; UEMURA; SENA, 2011](#)). É uma doença que pode ter seu quadro agravado quando o diagnóstico ocorre numa fase tardia, o que diminui as possibilidades de cura do paciente.

De acordo com a Organização Mundial da Saúde (OMS), há duas estratégias para a detecção precoce do câncer: diagnóstico precoce, ou abordagem ágil e oportuna de pessoas com sinais e sintomas de câncer; e rastreamento, aplicação regular de um teste em pessoas aparentemente saudáveis, pertencentes a uma faixa etária de maior risco para a doença, com o objetivo de identificá-las em fase pré-clínica e reduzir a mortalidade por essa causa ([TOMAZELLI et al., 2016](#)).

Alguns dos sinais e sintomas deste câncer podem ser notados em fases iniciais, por exemplo: nódulos fixos e indolores na mama, axila e pescoço; pele da mama avermelhada; alterações no mamilo; líquido anormal sendo expelido pelos mamilos ([INCA, 2019](#)).

Muitos avanços vêm ocorrendo no tratamento do câncer de mama nas últimas décadas. Nos dias atuais, o conhecimento sobre as várias maneiras da doença se apresentar e as diversas terapêuticas estão disponíveis.

Segundo o ([INCA, 2019](#)), o tratamento do câncer de mama depende da fase em que a doença se encontra (estadiamento) e do tipo do tumor. Pode incluir cirurgia, radioterapia, quimioterapia, hormonioterapia e terapia biológica.

7.2.2 Processamento Digital de Imagens

O PDI é uma forma de processamento de dados, em que sua entrada e saída são imagens. O objetivo de seu uso é melhorar o aspecto visual de certas feições estruturais para o analista humano e fornecer outros

subsídios para a sua interpretação, inclusive gerando produtos que possam ser posteriormente submetidos a outros processamentos (CÂMARA et al., 1996).

Graças aos avanços tecnológicos obtidos nos últimos anos, nota-se que vem crescendo o leque de aplicações na área de PDI (FILHO et al., 2016a). Esses avanços fazem com que o numero de novas pesquisas cresça e as já existentes sejam aprimoradas e expandidas.

O uso das técnicas de PDI se expandiram, e são usadas para vários tipos de tarefas em diversas áreas, como: computação gráfica, processamento de dados, visão computacional, inteligência artificial, reconhecimento de padrões. Essas técnicas vêm sendo aproveitadas em diferentes áreas, como na medicina, na biologia, na astronomia, na área industrial, no auxilio a lei, entre outras aplicações.

No geral, o processo do PDI é subdividido em algumas etapas. Essas etapas podem vir a mudar de um trabalho para o outro, mas geralmente essas etapas são: aquisição da base de imagens, pré-processamento, segmentação, extração de características e classificação ou reconhecimento.

7.2.2.1 Aquisição das Imagens

A Aquisição das imagens é a primeira etapa a ser realizada. Para a realização desta etapa são necessários: um instrumento formador de imagem um dispositivo físico sensível a uma faixa do espectro eletromagnético, capaz de produzir um sinal elétrico proporcional ao nível de energia percebida; e um digitalizador, que converterá a saída elétrica do dispositivo de sensoreamento físico para a forma digital (GONZALEZ; WOODS; EDDINS, 2002).

7.2.2.2 Pré-Processamento

No pré-processamento, é feita uma melhora na qualidade da imagem. Esta melhora envolve duas categorias principais: métodos que operam no domínio espacial e métodos que operam no domínio de frequência.

Técnicas de processamento no domínio espacial baseiam-se em filtros que manipulam o plano da imagem, enquanto que as técnicas de processamento no domínio da frequência se baseiam em filtros que agem sobre o espectro da imagem (ESQUEF; ALBUQUERQUE; ALBUQUERQUE, 2003). É comum utilizar ambas as técnicas combinadas, para realçar determinadas características de uma imagem.

7.2.2.3 Segmentação

A segmentação é a etapa onde é realizada a separação da imagem como um todo em partes que constituem a mesma, mas que se diferenciem entre si. Os *pixels* de interesse são geralmente denominados “objetos”, e é chamado de “fundo” os *pixels* que podem ser desprezados ou que não tem utilidade.

Dentre todas as etapas do processamento de imagens, a segmentação é considerada uma das mais críticas no tratamento da informação, pois é aqui onde são definidas as regiões de interesse para análises e processamentos posteriores. Assim, distorções ou erros nesta etapa refletem nas demais, o que pode vir a causar resultados indesejados causando contribuições negativas na eficiência do processo.

A segmentação é um processo adaptativo, ou seja, ela busca sempre se adequar as características particulares de cada imagem e objetivo que se pretende alcançar. Ainda que exista uma vasta diversidade de técnicas de segmentação de imagens, ainda assim há um grande interesse no estudo, desenvolvimento e adaptação de novas técnicas.

A segmentação possui duas abordagens principais, a similaridade entre os *pixels* e a descontinuidade entre eles. Na similaridade a técnica mais usada é a Binarização (*Image Thresholding*), pois é uma técnica eficiente e simples do ponto de vista computacional. A binarização é usada quando as amplitudes dos níveis de cinza são suficientes para caracterizar os “objetos” presentes na imagem (ESQUEF; ALBUQUERQUE; ALBUQUERQUE, 2003).

Já as técnicas baseadas em descontinuidade, procuram determinar variações bruscas entre *pixels* próximos, o que permite identificar os grupos de *pixels* que formam o limite das bordas dos objetos da imagem. Essa é uma técnica de segmentação muito eficiente para a detecção de bordas

7.2.2.4 Extração de Características

A etapa de extração de características é uma das etapas finais de um sistema de processamento de imagens. É nesta etapa que são retiradas as informações úteis da imagem que foi segmentada ou pré-processada. Quando o objetivo do processamento é a obtenção de informações em números, é realizada uma extração de atributos.

De forma geral, a extração de características pode ser subdividida em algumas categorias de análise, sendo elas por: textura e forma. O objetivo geral de uma análise por textura, é descrever aspectos da imagem que dizem respeito a suavidade, rugosidade e regularidade. Já na análise baseada na forma da imagem, o intuito é extrair informações que mensuram sobre propriedades estruturais da imagem (FILHO et al., 2016a).

Nessa etapa pode-se utilizar aprendizado profundo, como por exemplo uma Rede Neural Convolucional, porém, extração ocorre de maneira diferente. Essas redes profundas possuem em sua arquitetura muitas camadas que são responsáveis pela extração de características. Uma característica interessante é a capacidade da rede treinar as camadas de extração subsequentes, ao invés de usar valores predeterminados.

7.2.2.5 Classificação/Reconhecimento

Na classificação, o objetivo é realizar, de forma automática, a identificação dos objetos segmentados na imagem. Isto é realizado por um algoritmo que mede a similaridade dos objetos e classes, define uma função discriminante que separe os grupos e classifica os objetos de acordo com a sua função programada.

Podemos subdividir os classificadores em dois tipos, supervisionados e não supervisionados. Em ambos os casos são necessárias duas fases, que são o treinamento e a classificação de fato.

A classificação supervisionada se baseia na identificação de diferentes classes com comportamentos espectrais diferenciados. Para isso, algoritmos de classificação são adotados com o intuito de extrair as feições de interesse em um espaço multidimensional (CARVALHO; JÚNIOR, 2007).

Na classificação não-supervisionada o usuário utiliza algoritmos pra reconhecer as classes da imagem, e ao ter as áreas definidas, o usuário não se preocupa com a homogeneidade das classes. Esta classificação ocorre pelo submetimento dos *pixels* de uma área a um algoritmo de agrupamento que determina o agrupamento do dado.

7.2.3 Deep Learning

Deep Learning, trata-se de um conjunto de métodos de *machine learning*, com foco em problemas com o nível de abstração elevado (LECUN et al., 1998), e é vista como uma das sub-áreas da inteligência computacional. As técnicas de aprendizado profundo também são muito utilizadas para realizar classificação, eliminando variações irrelevantes das informações por meio de uma ampliação dos aspectos de entrada.

Dentre suas características, estão a presença de estruturas de grafos profundos separados por níveis de processamento. Esses algoritmos se assemelham à complexa rede de neurônios humanos, onde as ligações são representadas pelos nós dos grafos. Por sua robustez e abstração, os métodos de aprendizado profundo podem ser usados para muitos objetivos e aplicações, como por exemplo: reconhecimento facial, segmentação e classificação de imagens.

7.2.3.1 Redes Neurais Convolucionais

Uma Rede Neural Convolucional (CNN) é uma variação das redes de perceptrons de múltiplas camadas, tendo sido inspirada no processo biológico de processamento de dados visuais (VARGAS; PAES; VASCONCELOS, 2016). Este tipo de rede vem sendo amplamente utilizada em aplicações de classificação, extração de características, detecção e reconhecimento em imagens e vídeos.

Uma CNN é capaz de aplicar filtros de dados visuais mantendo a relação dos *pixels* vizinhos durante o processamento da rede. É uma arquitetura de aprendizado profundo composta por uma ou várias camadas convolucionais e subsequentes, seguidas opcionalmente de camadas totalmente conectadas, como em uma rede neural multicamada padrão (LECUN et al., 1998).

A utilização de CNN tem se tornado bem popular em tarefas de visão computacional. Elas possuem uma facilidade em serem treinadas quando a quantidade de amostras rotuladas que representam as diferentes classes alvos é grande. O treinamento delas é facilitado pelo fato de cada unidade de camada não estar conectada a todas as unidades da camada seguinte.

De forma simplificada, pode ser considerada uma CNN toda rede neural que utiliza a convolução no lugar da multiplicação de matrizes em pelo menos uma de suas camadas (GOODFELLOW; BENGIO; COURVILLE, 2016). Na atualidade, as redes convolucionais existentes trazem decisões de projeto que são importantes, decisões essas que foram adotadas e aprimoradas ao longo de seu desenvolvimento.

As CNNs são arquiteturas compostas por diversas camadas, entre elas temos a camada convolucional. Esta é responsável pela parte mais pesada do processamento computacional. Ela é constituída por filtros que recebem um arranjo de *pixels*, que pode ser chamado de volume. Os conjuntos de filtros dessa camada são capazes de aprender de acordo com seu treinamento.

Esses filtros realizam convoluções com os dados de entradas para obter um mapa de características, que indicam regiões onde características específicas em relação ao filtro, são encontradas na entrada. Assim, os valores reais dos filtros são alterados ao longo do treinamento, tornando possível que a rede aprenda a identificar regiões significantes para extrair características dos dados.

É comum que em seguida a uma camada convolucional, exista uma camada de *pooling*. Essa camada tem como objetivo reduzir gradualmente a dimensão espacial do volume de entrada, para que seja reduzido também o custo computacional da rede e evitar *overfitting* (ARAÚJO et al., 2017a).

Na operação de *pooling*, ocorre uma substituição dos valores pertencentes a uma região do mapa de atributos que foi gerado pela camada convolucional, por alguma das métricas dessa região. É comum a utilização de uma técnica em que os valores são substituídos pelo valor máximo de uma região (GOODFELLOW; BENGIO; COURVILLE, 2016), isso é chamado de *max pooling*.

7.2.3.2 Transfer Learning

Transfer Learning é a reutilização de um modelo pré-treinado de CNN em um novo problema. Basicamente, é o processo de transferir o conhecimento de uma tarefa de origem para uma tarefa de destino permitindo progresso rápido ou duradouro ao modelar a segunda tarefa (PAN; YANG, 2009).

A ideia geral sugere a utilização de uma rede neural que já foi treinada em um novo conjunto de dados, com o objetivo de solucionar um problema diferente. Em alguns estudos, é difícil obter um conjunto de dados grande o suficiente, o que descarta a possibilidade de treinar a rede do zero. Outro fator é o custo computacional para treinar uma CNN. Quanto mais profunda a arquitetura, mais caro e custoso o treinamento da mesma.

A *Transfer Learning* pode ser bem eficaz quando comparada com o método normal de *machine learning*, se saindo melhor em alguns aspectos como velocidade, precisão e a necessidade de um conjunto de treino menor. A Figura 58 resume como os dois processos ocorrem.

Machine Learning tradicional vs Transfer Learning

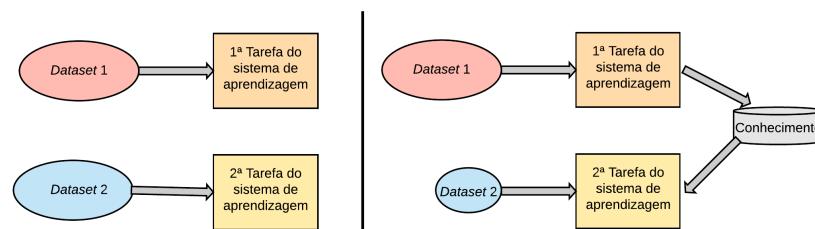


Figura 58 – Ao lado esquierdo, o processo de realização de duas tarefas de *Machine Learning*. Na direita, o *Transfer Learning* sendo utilizado para passar o conhecimento da tarefa de origem, para a de destino.

Para este trabalho foram selecionadas 3 arquiteturas de CNN previamente treinadas, com pesos pré-definidos e estas foram testadas e tiveram suas performance avaliadas quanto ao *Transfer Learning*. Os modelos utilizados foram as VGGNets (VGG16 e VGG19) (SIMONYAN; ZISSERMAN, 2014a) e a Residual Network (ResNet50) (HE et al., 2016).

7.2.3.2.1 VGGNets

A VGG é uma rede neural convolucional profunda proposta e desenvolvida por um grupo de Oxford (SIMONYAN; ZISSERMAN, 2014a). A mesma garantiu a primeira e segunda posições nas tarefas de localização e classificação, respectivamente, no desafio *ImageNet* 2014. Esta rede geralmente possui as profundidades de 16 e 19 camadas.

Esta rede traz consigo alguns padrões: seus filtros são de tamanho 3x3, *pools* máximos são alocados após cada duas convoluções e o número de filtros deveria ser dobrado após cada *max-pooling*.

A VGG é uma rede amplamente utilizada pela comunidade, inclusive para *Transfer Learning*. Ela possui arquitetura homogênea e robusta para extração de características.

Neste trabalho, realizamos a extração de características por meio da VGG com 16 e com 19 camadas. A VGG19 é um modelo originado pela VGG16, com a principal diferença em seu número de camadas (A VGG19 possui apenas 3 camadas de convolução a mais).

Inicialmente, o conceito da VGG19 era diferente da VGG16, sendo que a VGG19 foi utilizada com o objetivo de detecção de objetos (SIMONYAN; ZISSERMAN, 2014a) quando foi desenvolvida.

7.2.3.2.2 ResNet50

A ResNet50 é uma arquitetura que tem como principal característica suas conexões residuais. Esta arquitetura foi a vencedora do desafio *ImageNet* (DENG et al., 2009b) em 2015, onde realizam a classificação de imagens em 1000 classes diferentes. Esta arquitetura, trabalha de maneira a aprender resíduos, em vez de aprender características (HE et al., 2016).

A parte residual pode ser entendida como a subtração de características aprendidas da entrada dessa camada e a ResNet faz isso usando conexões de atalho, que são conexões em que o gradiente é diretamente retro-propagado para camadas anteriores.

A ResNet50 é capaz de solucionar o problema do desaparecimento de gradiente, que é ocasionado por meio da extrema profundidade da rede. Devido a essa profundidade da rede, os gradientes calculados onde é a função de perda acabam diminuindo para zero após as diversas aplicações da regra da cadeia. Dessa maneira, os pesos nunca tem seus valores atualizados e a rede torna-se incapaz de aprender. Já na arquitetura residual, os gradientes podem proceder através das conexões de salto de maneira direta.

7.2.4 Índices de Diversidade

A diversidade é um termo muito utilizado na área da ecologia. O seu objetivo é informar variedade de espécies presentes em uma comunidade ou área. Para realizar a medição da diversidade são utilizados dois componentes: A riqueza da espécie que consiste no número das espécies em uma determinada região e a abundância relativa, que é o número de indivíduos de uma determinada espécie numa determinada área.

O resultado obtido por qualquer índice de diversidade é representado por um único valor. A forma mais básica de utilizar índices de diversidade em imagens, é quando a comunidade representa uma imagem ou região da mesma, as espécies sendo os níveis de cinza e os indivíduos sendo os pixels.

Estes índices podem agir como descritores de textura das imagens possibilitando a extração de características das mesmas. Desta maneira, é possível reconhecer a heterogeneidade e homogeneidade das imagens, o que auxilia a classificar as mesmas como normais ou doentes.

A Tabela 37 mostra como são interpretados cada um dos elementos destes índices quando aplicados ao processamento de imagens.

Foi realizada a extração das características baseada em textura utilizando 10 índices: Índice de Shannon, Índice de Simpson, Índice de Margalef, Índice de McIntosh, Índice de Camargo, Índice de Smith & Wilson

Tabela 37 – Interpretação dos elementos da ecologia dos índices de diversidade no Processamento de Imagens

Metodologia Proposta	Ecologia
Região de Interesse da Imagem	Comunidade
Níveis de Cinza da Imagem	Espécies
Pixeis da Imagem	Indivíduos

B, Índice de Smith & Wilson 1-D, Índice de Smith & Wilson -lnD, Índice de Shannon máximo e Índice de Shannon mínimo.

7.2.4.1 Índice de Shannon

O Índice de Shannon assume que os indivíduos são randomicamente amostrados a partir de uma comunidade infinitamente grande, e que todas as espécies estão representadas na amostra (SHANNON; WEAVER, 1949). O cálculo deste índice ocorre conforme a Equação 7.1

$$H' = - \sum_{i=1}^S p_i \ln p_i, \quad (7.1)$$

onde S é total de espécies, p_i é a proporção de indivíduos pertencentes a espécie i na amostra, calculado com a Equação 7.2

$$p_i = \frac{n_i}{N}, \quad (7.2)$$

onde n_i é o número de indivíduos na espécie i na amostra e N é o número total de indivíduos na comunidade.

7.2.4.2 Índice de Simpson

O Índice de Simpson (D) é a medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma comunidade infinitamente grande, pertencerem à mesma espécie (SIMPSON, 1949). O Índice é calculado da pela Equação 7.3

$$D^1 = 1 - \sum_{i=1}^S p_i^2, \quad (7.3)$$

onde p_i é o número de indivíduos na espécie i na comunidade, n_i é o número de indivíduos na espécie i e N é o número total de indivíduos na comunidade. S é o total de espécies predominante na comunidade.

7.2.4.3 Índice de Margalef

Índice de Margalef (D) é usado na ecologia para estimar a biodiversidade de uma comunidade com base na distribuição numérica dos indivíduos das diferentes espécies em função do número total de indivíduos existentes na amostra analisada (MARGALEF, 1969). O Índice é calculado como mostra a Equação 7.4

$$D^2 = \frac{(S - 1)}{\ln(N)}, \quad (7.4)$$

onde S é o numero de espécies e $\ln(N)$ é o logaritmo do número de indivíduos pertencente a comunidade.

7.2.4.4 Índice de McIntosh

No índice proposto por McIntosh (1967), a comunidade pode ser vista como um ponto em um hiper-volume S-dimensional e a distância euclidiana da comunidade para a origem pode ser utilizada como uma medida de diversidade. E uma medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma

comunidade infinitamente grande pertencerem a mesma espécie (ARAÚJO et al., 2017). O Índice é calculado a partir da Equação 7.5,

$$D^3 = \frac{(N - U)}{N - \sqrt{N}}, \quad (7.5)$$

onde N é o número total de indivíduos na amostra e U é dado pela Equação 7.6

$$U = \sqrt{\sum n_i^2}, \quad (7.6)$$

onde n_i é o número de indivíduos na espécie i e o somatório é realizado sobre todas as espécies. U é a distância euclidiana da comunidade desde a origem quando plotada em um hiper volume S-dimensional.

7.2.4.5 Índice de Camargo

O índice de Camargo (1993) é a relação entre pares de espécies i e j . O Índice é definido pela Equação 7.7

$$E^1 = 1 - \left[\sum_{i=1}^S \sum_{j=i+1}^S \frac{p_i - p_j}{S} \right], \quad (7.7)$$

onde p_i é a proporção de espécies i na amostra, p_j é a proporção de espécies j na amostra e S é o número total de espécies.

7.2.4.6 Índice de Smith & Wilson B

O índice de B de Smith e Wilson (1996) é baseado na variação de abundância. Essa variância é calculada usando abundâncias de logaritmo, o que significa que o índice examina as diferenças proporcionais entre as espécies. Com a obtenção da variância, a mesma é multiplicada pelo fator $-2 / \pi \arctan$ para dar um valor entre o intervalo de 0 a 1. A Equação 7.8 define o índice

$$E^2 = 1 - \left[\frac{2}{\pi \arctan \left\{ \frac{\sum_{i=1}^S (\ln n_i - \sum_{j=1}^S \frac{\ln n_j}{S})^2}{S} \right\}} \right], \quad (7.8)$$

sendo n_i o número de indivíduos nas espécies i , n_j é o número de indivíduos nas espécies j e S é o número total de espécies.

7.2.4.7 Índice de Smith & Wilson 1-D

O índice de uniformidade de (SMITH; WILSON, 1996) 1-D utiliza o cálculo da dominância de (SIMPSON, 1949) (D) de várias maneiras. A mais comum é usar o complemento de D como índice de diversidade e para dados contínuos ou com números grandes de cortes discretos. O cálculo é definido conforme a Equação 7.9

$$E^3 = \frac{1 - D}{1 - \frac{1}{S}}, \quad (7.9)$$

onde D é o índice de diversidade de Simpson e S é o número total de espécies.

7.2.4.8 Índice de Smith & Wilson -lnD

O índice de uniformidade de Smith e Wilson (1996) 1/D converte a dominância de Simpson (1949) (D) em uma medida de diversidade. O índice é definido pela Equação 7.10

$$E^4 = \frac{-\ln D}{\ln S}, \quad (7.10)$$

onde D é o índice de diversidade de Simpson e S é o número total de espécies.

7.2.4.9 Índice de Shannon *maximum*

Este índice é simplesmente o valor máximo que o índice de Shannon pode produzir para um conjunto de dados fornecido. O cálculo é dado por $\ln(S)$, onde S é o número total de espécies.

7.2.4.10 Índice de Shannon *minimum*

Este índice é simplesmente o valor mínimo que o índice de Shannon pode produzir para um conjunto de dados fornecido. O mesmo é dado pela Equação 7.11

$$S_{min} = -\ln(N) - \left[\frac{(N - S + 1) \ln(N - S + 1)}{N} \right], \quad (7.11)$$

onde N é o número total de indivíduos na amostra e S é o número total de espécies.

7.3 Trabalhos Relacionados

Atualmente, vem se difundindo cada vez mais o uso de técnicas de *machine learning* para atividades em áreas diversas, e a área médica é uma delas. Nessa seção serão abordados brevemente alguns trabalhos correlatos diretamente ao método aqui proposto.

Em Araújo et al. (2017) é proposto um método para a classificação de imagens de biópsia mamárias, utilizando CNN. As imagens são classificadas em quatro classes, tecido normal, lesão benigna, carcinoma *in situ* e carcinoma invasivo, e em duas classes, carcinoma e não carcinoma. As características extraídos pela CNN também são usados no classificador SVM. A acurácia alcançada foi de 77,8% para quatro classes e 83,3% para carcinoma / não carcinoma.

Nawaz et al. (2018) apresenta o ajuste fino de uma CNN profunda (ALEXNET) alterando e inserindo camadas convolucionais da camada de entrada e a camada totalmente conectada. Este método apresenta resultados experimentais que alcançam a acurácia de 81,25%.

Golatkar, Anand e Sethi (2018) propõe um método baseado em aprendizado profundo, onde é realizado o ajuste fino de uma CNN Inception-v3 para classificação de imagens de tecido mamário coradas com H&E liberadas para o desafio BACH (ARESTA et al., 2019a). Foi obtida uma acurácia média de 85% nas quatro classes e 93% para não-câncer (isto é, normal ou benigno) vs. tumor maligno (carcinoma invasivo ou *in situ*).

Rakhlin et al. (2018) desenvolve uma abordagem computacional baseada em redes neurais convolucionais profundas para classificação de imagens histológicas de câncer de mama. Neste trabalho, a classificação de 4 classes alcançou acurácia de 87,2%. Para classificação de 2 classes para detectar carcinomas, relatamos 93,8% de acurácia.

Ferreira et al. (2018) propõe uma Inception ResNet v2 que por meio de *Transfer Learning*, ajuste fino e aumento de dados, realiza a classificação de imagens histológicas de câncer de mama. A rede é capaz de aprender quais características são relevantes para a classificação. Este trabalho utiliza a base BACH ICIAR 2018 e atingiu acurácia de 76% para a classificação de 4 classes no conjunto de teste.

Em Vesal et al. (2018) é proposta uma abordagem baseada em *Transfer Learning* para a classificação de imagens histopatológicas da mama em 4 classes: normal, benigno, carcinoma *in situ* e carcinoma invasivo. As imagens são da base BACH ICIAR 2018. Primeiramente as imagens são normalizadas para corrigir variações de cores induzidas durante a preparação das lâminas. Após isto, são extraídos patches das imagens e estes são usados para ajustar as CNNs Inception-V3 e ResNet50, ambas pré-treinadas. Foi utilizado *cross-validation* com k=3 para avaliação. A Inception-V3 alcançou uma acurácia de teste média de 97,08% para quatro classes, superando marginalmente a rede ResNet50, que alcançou uma média de 96,66%.

Dentro da literatura, há poucos trabalhos semelhantes onde são utilizados os índices de diversidade ecológicos para caracterizar textura, voltado ao problema do câncer de mama.

O trabalho de Carvalho, Filho e Dalilia (2017), onde é apresentado um método de classificação de tecidos da mama. São usados descritores de textura baseados em índices filogenéticos para extração de características, em seguida feita uma classificação usando Bagging, BayesNet e Random Forest. A base utilizada é a DDSM. Os resultados alcançaram uma acurácia de 90%, sensibilidade de 88,5% e especificidade de 91,7%.

Como na literatura há trabalhos que utilizam este método de análise de textura para a tarefa de texturização de imagens suspeitas de câncer de mama, foi selecionado o trabalho de [Reis et al. \(2012\)](#). Em seu trabalho é investigado o potencial grau de caracterização de textura dos índices de diversidade aplicado em imagens digitais de faces humanas. São utilizados os índices como extratores de características e o SVM como classificador. Dentre os 18 testes realizados foram obtidas acuráncias superiores a 95%.

Abaixo, a Tabela 38 resume os trabalhos relacionados, as ferramentas e tecnologias utilizadas e a base de imagens.

Tabela 38 – Resumo dos trabalhos relacionados.

Trabalho	Método	Base de Imagens
(REIS et al., 2012)	Índices de Diversidade	CBCL, ORL, Libor Spacek e CMU AMP Face Expression
(ARAÚJO et al., 2017)	CNN	Bioimaging Challenge 2015 Breast Histology
(CARVALHO; FILHO; DALILIA, 2017)	Índices Filogenéticos	DDSM
(GOLATKAR; ANAND; SETHI, 2018)	Ajuste Fino da Inception-v3	BACH
(RAKHLIN et al., 2018)	Deep CNN	ICCIAR 2018
(NAWAZ et al., 2018)	CNN AlexNet	BACH
(FERREIRA et al., 2018)	CNN Inception	ICCIAR 2018
(VESAL et al., 2018)	CNN ResNet v2	BACH
Método Proposto	ResNet50 e Inception-v3	ICCIAR 2018
	<i>Transfer Learning</i> e Índices de Diversidade + Random Forest e SVM	BACH ICCIAR 2018

De acordo com a literatura pesquisada, as técnicas de PDI e *machine learning* vem tornando-se imprescindíveis no auxílio aos especialistas em diagnósticos envolvendo imagens médicas. Uma desvantagem do aprendizado profundo é seu alto custo computacional, pois estes algoritmos necessitam um grande período de treino e grande quantidade de imagens para atingir bons resultados. Este empecilho pode ser resolvido utilizando *Transfer Learning*, que trata-se da utilização de redes neurais convolucionais pré treinadas. Esta abordagem é efetiva e muito útil principalmente quando o conjunto de dados é pequeno.

Na abordagem aqui proposta, serão utilizados alguns modelos de CNN pré-treinados (ResNet50, VGG16 e VGG19) juntamente com os índices de diversidade para a extração de características. É feita a comparação da eficiência dos dois métodos testando os atributos extraídos nos classificadores *Random Forest* e *SVM*. Há também a concatenação entre as características dos índices de diversidade e do modelo de *Transfer Learning* que obteve o melhor resultado, e são refeitos os testes com os dois classificadores.

7.4 Metodologia

A metodologia que será seguida para a realização deste trabalho consiste em: aquisição do *dataset* de imagens; extração das características aplicando a técnica de *Transfer Learning* e análise de textura utilizando índices de diversidade ecológicos; classificação com *Random Forest* e *Support Vector Machine*; e validação dos resultados. A Figura 59 ilustra a metodologia.

7.4.1 Aquisição de Imagens

Para este trabalho, a base de imagens utilizada foi a base de imagens BACH ICIAR 2018 ([ARESTA et al., 2019a](#)). A mesma é utilizada em um desafio que contém como um de seus objetivos a classificação do tipo de câncer de mama. A base de dados utilizada no trabalho consiste em 400 imagens divididas em 4 classes e cada uma das classes possui 100 imagens.

Cada imagem é rotulada com uma das quatro classes balanceadas: carcinoma normal (*normal*), benigno (*benign*), *in situ* (*InSitu*) e carcinoma invasivo (*invasive*), onde a classe é definida como um tipo de câncer predominante na imagem. A anotação da imagem foi realizada por dois médicos especialistas.

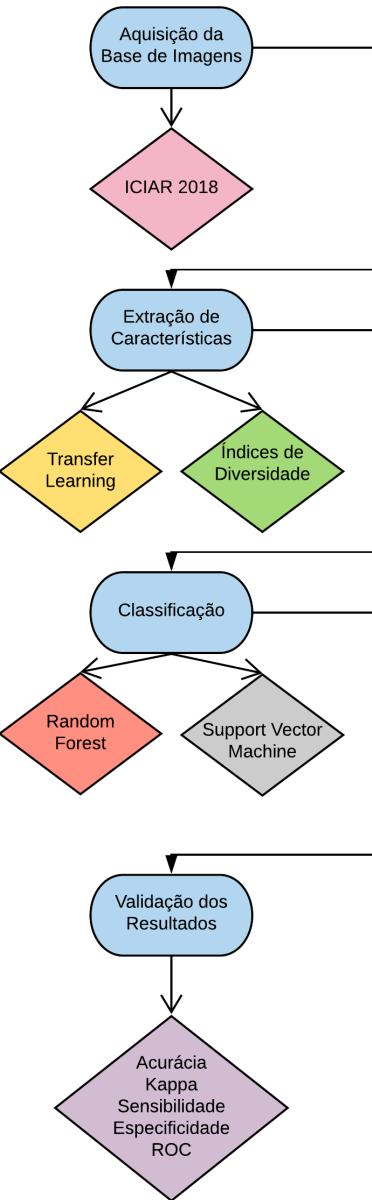


Figura 59 – Representação resumida da metodologia proposta.

Na abordagem aqui proposta, a base foi agrupada em duas classes de imagens. A classe normal permanece sem alterações, enquanto as outras 3 classes (benigna, in situ e invasivo) se tornam uma classe que denominamos doente. Desta maneira, teremos 100 imagens que não possuem carcinomas, enquanto outras 300 possuem.

A seguir, na Figura 60, são mostrados exemplos de imagens contidas no *dataset* utilizado durante este trabalho.

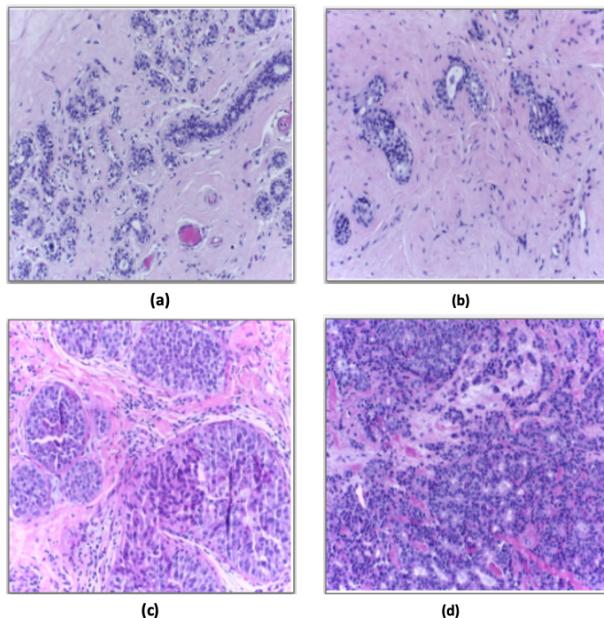


Figura 60 – Exemplos da Base de imagens BACH ICIAR 2018. (a) normal, (b) benigno, (c) insitu e (d) invasivo.

7.4.2 Extração de Características

A etapa de extração de características foi concebida utilizando duas técnicas diferentes: *Transfer Learning* e análise de textura. Foram utilizadas 3 arquiteturas de CNN unidas ao *Transfer Learning* e índices de diversidade ecológica atuando como descriptores de textura para realizar a etapa em questão.

7.4.2.1 Transfer Learning

Para mensurar a efetividade das diferentes formas de implementar uma CNN, foram utilizadas diferentes arquiteturas para a extração dos atributos das imagens. Visando melhores resultados e diminuir o tempo e custo computacional das CNNs, utilizamos *Transfer Learning*.

Para este trabalho, foram selecionados os três dos principais modelos para serem estudados e terem sua performance avaliados para o problema em questão. Os modelos utilizados neste estudo foram a ResNet50, a VGG16 e a VGG19. Estas arquiteturas foram escolhidas pois apresentam bons resultados ao solucionar problemas semelhantes, como foi visto em alguns trabalhos relacionados.

Todos estes modelos foram pré-treinados na mesma base de imagem, a ImageNet (DENG et al., 2009b). Os modelos foram treinados para classificar imagens entre 1000 classes diferentes. Para utilizar estes modelos como extrator de características, foi retirada a última camada, e foram salvos os valores que estavam contidos nesta camada.

7.4.2.2 Índices de Diversidade Ecológicos

O objetivo geral de uma análise por textura, é descrever aspectos da imagem que dizem respeito a suavidade, rugosidade e regularidade. Desta maneira, é possível reconhecer a heterogeneidade e homogeneidade das

imagens, o que auxilia a classificar as mesmas como normais ou doentes. A Tabela 37 mostra como são interpretados cada um dos elementos destes índices quando aplicados ao processamento de imagens.

Os índices de diversidade de espécies no contexto da ecologia indicam variedade de espécies, possibilitando abordar informações de como estão distribuídas as espécies em uma comunidade. Com isso podemos abordar riqueza e equalidade de espécies. A riqueza refere-se ao numero de espécies, quanto maior a riqueza, mais espécies há em uma comunidade. A equalidade refere-se a abundância relativa, ou seja, quanto mais próximas as abundâncias das espécies dentro de uma comunidade, maior é a equalidade.

Neste trabalho, os índices de diversidade são utilizados para medir a diversidade na distribuição dos níveis de cinza presente nas regiões de interesse de uma imagem. A finalidade é utilizar o valor do índice para caracterizar textura nas imagens fornecendo os dados necessários para o reconhecimento de um padrão que possibilite ao classificador distinguir as regiões que correspondem o tecido tumoral e àquelas que correspondem a tecidos normais, não-massas.

7.4.3 Classificação

A etapa da classificação tem como objetivo mensurar a similaridade dos objetos e classes contidos na imagem, definir uma função discriminante que separe estes grupos e classificar os objetos de acordo com sua função programada. Em nossa abordagem, foram utilizados os algoritmos *Random Forest* e *Support Vector Machine*, pois os mesmos apresentam bons resultados em diversas tarefas semelhantes que envolvem classificação.

Foi utilizado o método de *cross validation k-folds* com o $k=10$. Este processo realiza a divisão do conjunto de dados em 10 grupos, separando 9 grupos de treino, e um para teste, realizando 10 cruzamentos alterando o grupo de teste. Desta maneira todos os grupos são testados, e é gerada uma média que atribui-se como resultado.

7.4.3.1 Random Forest

O *Random Forest* é um algoritmo de classificação que combina múltiplas árvores de previsão, onde cada árvore depende dos valores de um vetor independente amostrado aleatoriamente e com a distribuição de amostras para todas as árvores da floresta (WRIGHT; WAGER; PROBST, 2016).

Neste trabalho, utilizamos os valores de parâmetros setados por padrão do *Random Forest*. Os mesmos vêm setados das seguintes maneiras:

- I: Número de interações = 100;
- K: Número de atributos que serão utilizados para investigação = 0;
- M: Número mínimo de instâncias por folha = 1;
- V: Proporção mínima da variação da classe numérica = 0.001;
- S: Seed para o gerador de números aleatórios = 1.

7.4.3.2 Support Vector Machine

O *Support Vector Machine* (SVM) é um algoritmo de classificação supervisionada de *machine learning* proposto por (DRUCKER et al., 1997). Neste trabalho, realizamos testes com os diversos parâmetros do SVM para obter melhores resultados.

Os parâmetros utilizados no SVM neste trabalho são apresentados à seguir:

- S: Tipo de SVM utilizado = C-SVC;
- K: Tipo de função do *kernel* = linear, polinomial, RBF (Função de Base Radial) e sigmoide;
- C: Custo da função = 1 e 10;

- D: Degrau na função de *kernel* (aplica-se somente no *kernel* polinomial) = 6;
- G: Número do *gamma* na função do *kernel* = 0;
- R: Número do *tetha* na função do *kernel* = 0;
- E: Tolerância do critério de finalização = 0,001;
- P: Valor do epsilon na função de *loss* = 0.1.

7.4.4 Métricas de Validação

Para que os resultados obtidos com o presente método possam ser validados, foram utilizadas as métricas: acurácia, especificidade, sensibilidade, coeficiente Kappa e Área Sobre a Curva (ROC).

Para que algumas dessas métricas sejam calculados é necessária uma matriz de confusão. A mesma esta ilustrada segundo a Tabela 39.

Tabela 39 – Matriz de confusão

Resultado	Doença	
	Normal	Doente
Positivo	Verdadeiro	Falso
	Positivo (VP)	Positivo (FP)
Negativo	Falso	Verdadeiro
	Negativo (FN)	Negativo (VN)

7.4.4.1 Acurácia

Acurácia (A) é a proporção de predições corretas, sem levar em consideração o que é positivo e o que é negativo. Esta medida é altamente suscetível a desbalanceamentos do conjunto de dados e pode facilmente induzir a uma conclusão errada sobre o desempenho do sistema. Representado pela Equação 7.12

$$A = \frac{VP + VN}{VP + FP + VN + FN}. \quad (7.12)$$

7.4.4.2 Sensibilidade

Sensibilidade (S) é a proporção de verdadeiros positivos, ou seja, a capacidade do sistema em predizer corretamente a condição para casos que realmente a têm. Representado pela Equação 7.13

$$S = \frac{VP}{VP + FN}. \quad (7.13)$$

7.4.4.3 Especificidade

Especificidade (E) é a proporção de verdadeiros negativos, isto é, a capacidade do sistema em predizer corretamente a ausência da condição para casos que realmente não a têm. Representado pela Equação 7.14

$$E = \frac{VN}{VN + FP}. \quad (7.14)$$

7.4.4.4 Kappa

O coeficiente de concordância Kappa (*k*) é uma estatística que mede a concordância entre dois ou mais julgadores, quando realizam uma avaliação nominal ou original de um mesmo conjunto ou amostra (COHEN, 1960). O Kappa pode ser interpretado como é exibido na Tabela 40.

Tabela 40 – Interpretação de concordância para os valores do índice Kappa.

Kappa (k)	Classificação
$k < 0.2$	Fraca
$0.2 \leq k < 0.4$	Razoável
$0.4 \leq k < 0.6$	Moderada
$0.6 \leq k < 0.8$	Forte
$k \geq 0.8$	Perfeita

7.4.4.5 Área Sobre a Curva

A Área Sobre a Curva possibilita quantificar a exatidão de um teste diagnóstico (proporcional à área sob a curva), além de possibilitar a comparação de testes de diagnósticos. A Área sobre a Curva ROC constitui um dos índices mais usados para sumarizar a qualidade da curva.

As curvas ROC estão sendo usadas para avaliar a capacidade de discriminação de vários métodos estatísticos que combinam várias pistas, resultados de testes e entre outros propósitos preditivos ([PARADELA; LOURENÇO; VERAS, 2005](#)).

7.5 Resultados e Discussões

Para viabilizar que uma abordagem computacional possa auxiliar um especialista na classificação de biópsias mamárias, é crucial que haja um bom resultado nos testes realizados e que as métricas de avaliação do método sejam bem equilibradas. É preferível também que estes testes não sejam tão demorados.

Para a realização destes experimentos, foi utilizada a plataforma Google Colaboratory ([CARNEIRO et al., 2018](#)). A mesma trata-se de um serviço baseado em *cloud* que utiliza Jupyter Notebooks para disseminar a pesquisa em *machine learning*. No Colab, como também é conhecido, é fornecido um ambiente totalmente configurado e pronto para ser utilizado, com acesso gratuito a uma GPU robusta.

Algo a ser levado em conta é o tempo. a extração de características utilizando *Transfer Learning* é mais demorada do que a feita pelos índices de diversidade. A Tabela 41 ilustra o tempo médio aproximado que os extratores de características levam para atuar em cada imagem.

Tabela 41 – Tempo aproximado de extração de característica de uma imagem em segundos.

Extrator	Tempo médio
ResNet50	≈ 10 segundos
VGG16	≈ 12 segundos
VGG19	≈ 14 segundos
Índices de Diversidade	≈ 3 segundos

Nos testes realizados com o *Random Forest*, as características extraídas por meio de *Transfer Learning* levaram a resultados que se aproximam um do outro. No entanto, os índices de diversidade tiveram um desempenho um pouco abaixo dos demais. A Tabela 42 mostra os resultados referentes ao classificador *Random Forest*.

Tabela 42 – Resultados com *Random Forest*

Extrator	A (%)	k	S (%)	E (%)	ROC (%)
ResNet50	82,75	0,41	81,64	94,28	91,1
VGG16	84,5	0,5	89,58	83,8	93,2
VGG19	82,5	0,41	91,8	81,7	92,1
Índices de Diversidade	71,5	0,09	36,53	76,7	65,4

Já no classificador SVM, realizamos testes variando o parâmetro do *kernel*, alternando mesmo entre: função de base radial (RBF), sigmoide, polinomial e linear. Foram também realizados testes com o parâmetro

de custo variando entre 1 ou 10. A Tabela 43 exibe o desempenho do classificador em questão com C=1 e em todos os *kernels*.

Tabela 43 – Resultados com *Support Vector Machine* com C=1

Método de Extração	Kernel	A (%)	k	S (%)	E (%)	ROC (%)
ResNet50	RBF	87,5	0,61	86,54	92,1	76,3
	Sigmoide	85,25	0,51	95,55	83,94	71,2
	Polinomial	83,5	0,45	92,5	82,5	68
	Linear	89,75	0,72	79,79	93,02	86,2
VGG16	RBF	88,75	0,61	88,73	88,75	80,2
	Sigmoide	82,25	0,39	93,93	81,19	65,2
	Polinomial	84,25	0,54	71,41	86,91	75,5
	Linear	86,25	0,62	73,1	90,42	81,2
VGG19	RBF	86	0,56	86,66	85,88	74,7
	Sigmoide	84,25	0,48	91,11	83,38	69,8
	Polinomial	83,25	0,47	82,45	83,57	69,8
	Linear	88	0,67	77,65	91,17	83
Índices de Diversidade	RBF	75	0	-	75	50
	Sigmoide	75	0	-	75	50
	Polinomial	74	0,11	44,44	76,92	54,7
	Linear	75	0	-	75	50

Com o parâmetro de custo = 10, a ResNet50 seguiu superior nos resultados, como é mostrado na Tabela 44.

Tabela 44 – Resultados com *Support Vector Machine* com C=10

Método de Extração	Kernel	A (%)	k	S (%)	E (%)	ROC (%)
ResNet50	RBF	91,75	0,77	86	93,5	87,8
	Sigmoide	90,5	0,72	89,7	90,7	83,7
	Polinomial	88	0,66	79,5	90,4	82,0
	Linear	89,75	0,72	79,8	93,0	86,2
VGG16	RBF	89	0,7	79,8	91,8	84,3
	Sigmoide	78,25	0,39	57,3	84,2	69,2
	Polinomial	85,5	0,6	72,3	89,5	79,7
	Linear	86,25	0,62	73,2	90,4	81,2
VGG19	RBF	87	0,64	76,7	90	81
	Sigmoide	84,25	0,55	72,8	87,1	75,8
	Polinomial	85,5	0,58	76,9	87,6	77,0
	Linear	88	0,67	77,7	91,2	83,0

Diante dos resultados expostos nas Tabelas 42, 43 e 43 pode-se observar que utilizando a *Transfer Learning* para a extração de características obtivemos resultados superiores aos índices de diversidade. As redes pré-treinadas atingiram bons números, em ambos os classificadores, já os índices atingiram apenas uma média razoável durante os experimentos.

Dentre as redes pré-treinadas, a ResNet50 obteve resultados ligeiramente superiores aos outros extrautores. Uma razão para isto ocorrer é sua concepção robusta para a extração de características devido a sua homogeneidade. Também observamos que diante dos resultados apresentados, o classificador SVM obteve resultados melhores quando comparado ao *Random Forest* para o problema abordado.

Um evento interessante ocorreu nos testes do SVM. Como exibe a Tabela 43 alguns valores de kappa e Sensibilidade foram zerados. Isto ocorreu devido o classificador reconhecer todas as instâncias como

pertencentes a classe doente quando os *kernels* RBF, sigmoide e linear eram utilizados. Quando testado com o *kernel* polinomial o SVM foi capaz de classificar as imagens nas duas classes, apesar dos resultados não serem tão satisfatórios.

Foi realizado um terceiro experimento, onde houve a concatenação das características extraídas com a ResNet50 e os índices de diversidade. Os resultados do teste citado acima aparecem nas Tabela 45. Foram testados os 2 classificadores novamente, com o *Random Forest* em sua parametrização padrão e variando os parâmetros de *kernel* e custo (C) do SVM entre os valores mencionados anteriormente nesta seção.

Tabela 45 – Resultados dos classificadores com as concatenação dos atributos da ResNet50 e dos índices de diversidade.

Classificador	Kernel	A (%)	k	S (%)	E (%)	ROC (%)
SVM (C=1)	RBF	89	0,67	88,8	89	80,7
	Sigmoide	82,25	0,39	91,3	81,1	65,2
	Polinomial	84,5	0,56	73,1	87,4	81,2
	Linear	86,25	0,62	73,2	83	92,6
SVM (C=10)	RBF	91,5	0,76	85,1	93,5	87,7
	Sigmoid	89,5	0,69	88,2	89,8	82,0
	Polynomial	90,25	0,73	81,4	93,1	86,5
	Linear	89,75	0,72	79,8	93	86,2
<i>Random Forest</i>	-	81,75	0,37	90,9	80,9	91,9

Após a realização deste último teste, foi notada uma leve melhora em alguns resultados. O classificador SVM continua a superar o *Random Forest* na tarefa de classificação aqui abordada. É importante mencionar que houve também a leve baixa em alguns resultados, como é exibido na Tabela 45.

Após estes testes, pudemos constatar que o melhor resultado foi utilizando as características extraídas pela ResNet50 no classificador SVM, com o *kernel* RBF e C=10.

A comparação com outros trabalhos dentro da literatura é uma tarefa um pouco complicada, já que há uma algumas diferenças nas maneiras de abordagem do problema e também nas ferramentas e técnicas utilizadas para solucioná-los. A Tabela 46 exibe alguns números comparativos.

Tabela 46 – Comparação do método proposto com alguns trabalhos relacionados.

Trabalho	Método	Base de Imagens	A (%)	k	S (%)	E (%)	ROC (%)
(GOLATKAR; ANAND; SETHI, 2018)	Ajuste Fino da Inception-v3	BACH ICIAR 2018	93	-	-	-	-
(CARVALHO; FILHO; DALILIA, 2017)	Índices Filogenéticos	DDSM	90	-	88,5	91,7	-
(ARAÚJO et al., 2017)	CNN	<i>Bioimaging Challenge 2015 Breast Histology</i>	83,3	-	-	-	-
(NAWAZ et al., 2018)	AlexNet	BACH ICIAR 2018	81,25	-	-	-	-
Método Proposto	Transfer Learning e Índices de Diversidade + Random Forest e SVM	BACH ICIAR 2018	91,75	0,77	86	93,5	87,8

No trabalho de Carvalho, Filho e Dalilia (2017), utilizou-se os índices como descritores de textura, porém, em um *dataset* para classificação de mamografias em benigno e maligno. No mesmo apresentaram bons resultados, mas nas imagens histopatológicas os mesmos não tiveram tal sucesso.

Já nos trabalhos de Araújo et al. (2017) e Nawaz et al. (2018) são utilizados métodos baseados em redes neurais convolucionais. É uma abordagem eficaz, no entanto, na maioria dos casos requer um grande poder de processamento, tempo de treinamento da rede e vasta quantidade de dados necessária para que se obtenha bons resultados. O trabalho de Golatkar, Anand e Sethi (2018) realiza o ajuste fino da arquitetura Inception-v3, onde a última camada é re-treinada para realizar a tarefa de classificação.

A abordagem aqui proposta utiliza redes pré-treinadas, por isso o tempo de execução é diminuído. Além disto, os testes foram realizados no Colab, que provê hardware robusto para realização de testes.

É importante ressaltar que alguns dos trabalhos relacionados realizam a classificação entre carcinoma / não carcinoma, diferente de nossa abordagem, que pretende identificar e separar biópsias normais de biópsias que possuem tumores. Há também trabalhos que realizam a classificação de 4 classes (normal, benigno, *inSitu* e invasivo).

7.6 Conclusão

No presente trabalho, testamos e avaliamos a eficiência de algoritmos na tarefa de classificação automática de imagens histopatológicas da mama em normal e doentes. Utilizando índices de diversidade e técnicas de *deep learning* como *Transfer Learning* para extração das características os resultados obtidos foram satisfatórios e foram alcançados com certa rapidez na classificação com o *Random Forest* e *SVM*.

A abordagem de aprendizado profundo VGG16 mostrou-se mais eficiente dentre as demais na tarefa de extração usadas no trabalho em questão. Já para a classificação, o *SVM* mostrou-se superior ao *Random Forest*, baseando-se nas métricas de avaliação que utilizamos para este trabalho.

O uso de *Transfer Learning* combinado com o *SVM* pode ser usado de forma eficiente na análise automática de biópsias mamárias, mesmo com um número pequeno de imagens. Acreditamos que a abordagem proposta poderá ser integrada a uma ferramenta de processamento digital de imagens sendo aplicada a casos reais, auxiliando especialistas fornecendo-lhes uma segunda opinião no processo de diagnóstico da biópsia. Com isso, espera-se garantir ao paciente um prognóstico mais favorável quanto à sua saúde e uma melhor qualidade de vida.

Como trabalhos futuros, destacamos:

- Realizar novos testes com uma quantidade maior de imagens / outros *datasets*;
- Realizar o ajuste fino para obter melhores resultados;
- Adaptar a abordagem para a classificação de 4 classes; e
- Utilizar outros classificadores.

Classificação automática de tecidos mamários em malignos e benignos utilizando Índices de diversidade

Autores: Antônio Oseas de Carvalho Filho; Igor Miranda Grangeiro.

Resumo: O câncer é um problema de saúde pública mundial. O câncer de mama é o tipo de câncer mais comum entre as mulheres no mundo e no Brasil, é causada pela multiplicação desordenada de células da mama. O presente estudo apresenta um método de classificação de tecidos mamários em maligno e benigno por meio de exames por imagens. Sendo assim, realizado através de rastreamento, onde o meio mais promissor para o diagnóstico precoce é em exames de mamografia. Na primeira etapa, é feita a aquisição das imagens oriundas da base Digital Data base for Screening Mammography (DDSM). Na segunda, o pré-processamento, através de técnicas de processamento de imagem. Na terceira etapa, a extração de características, através dos índices de diversidade. E na quarta etapa, é a classificação em benignos ou malignos, onde foram utilizados os classificadores Random Forest, Sequential Minimal Optimization (SMO), Multilayer Perceptron (MLP) e Instance Based Learning (IBK). Portanto, o processo é finalizado com a validação dos resultados. Neste método foram usados descritores de textura baseado em índices de diversidade para extração de características. Para todos os testes realizados, o melhor resultado alcançou uma acurácia de 77,92%, sensibilidade de 81,51%, especificidade de 74,88%, curva ROC de 0,852 e uma Kappa de 0,5592. O uso dos índices de diversidade para descrever padrões em regiões de imagens de mamografias mostrou-se moderado na categorização de maligno e benigno. O mesmo contribuiu de forma positiva em duas distintas áreas. Sendo que na área da saúde, disponibilizou uma metodologia automática para auxílio no diagnóstico de tecidos mamários em maligno e benigno, que apesar de não fornecer excelentes resultados mostram-se promissor. E na computação, com a adaptação de técnicas de outras áreas do conhecimento que serviram como descritores de textura, podendo então, serem utilizados em caracterizações em outros tipos de imagens.

Keywords: Câncer de mama, Mamografia, Descritores de Textura, Índice de Diversidade.

8.1 Introdução

O câncer é um problema de saúde pública mundial, tornando-se entre as principais causas de morbidade e mortalidade do mundo. A evolução das taxas de incidência e de mortalidade marcam um contínuo crescimento desta doença para as próximas décadas. A Organização Mundial de Saúde (OMS), em consequência do aumento da prevalência dos fatores de risco, estima ter um aumento de 70% de novos casos para as próximas duas décadas, principalmente em países de baixa renda ([CARVALHO et al., 2019](#))

Segundo [INCA \(2019\)](#), o câncer é um conjunto de mais de 100 doenças, que têm em comum o crescimento desordenado de suas células que invadem os tecidos e órgãos. O câncer de mama é o tipo de câncer mais comum entre as mulheres no mundo e no Brasil, correspondendo a cerca de 25% dos casos novos a cada ano, sendo importante ressaltar que no Brasil, esse percentual é de 29%. Em 2018, estimativas indicam 59.700 novos casos, sendo excluído o câncer de pele não melanoma. Os casos de mama são mais frequentes nas mulheres das regiões Sul, Sudeste, Centro-Oeste e Nordeste. [INCA \(2019\)](#) ainda cita, que o câncer de mama também é acometido em homens, porém raro, representando apenas 1% do total de casos da doença.

O câncer de mama não apresenta uma causa única. Contudo, diversos fatores podem estar relacionados ao desenvolvimento da mesma, tais como: fatores genéticos/hereditários (histórico de caso de câncer de mama na família), fatores comportamentais/ambientais (alcoolismo, obesidade), fatores endócrinos/história reprodutiva (história de menarca precoce, nuliparidade), além da idade. Os sintomas iniciais da doença podem ser perceptíveis, através de presença de nódulo, pele da mama avermelhada, retraída ou semelhante a casca de laranja, descarga papilar sanguinolenta, pequenos nódulos no pescoço ou nas axilas.

Tais sintomas citados podem ser palpáveis pelo autoexame da mulher, qualquer aparecimento destes deve ser investigado, procurando o serviço especializado, levando em conta que nem sempre estarão relacionados a um evento maligno. É de grande importância a detecção precoce, pois aumenta as chances de cura e tratamento. O tratamento do câncer de mama pode incluir intervenção cirúrgica, radioterapia, quimioterapia e hormonoterapia. Contudo, mesmo após o tratamento, é necessário o acompanhamento médico dos pacientes ([MAZALA, 2019](#)).

Na medicina, o uso de imagens são considerados importantes recurso para a elaboração de diagnósticos médicos. No auxílio desses diagnósticos, as técnicas de análise auxiliadas por computador têm sido utilizadas como segundo parâmetro para a elaboração de diagnósticos com uma maior precisão de acerto ([BATISTA et al., 2010](#)). Os Sistemas *Computer-Aided Detection* (CAD) são responsáveis por auxiliar os especialistas a detectar alterações em imagens digitais e em seguida serem analisadas pelos mesmos. Já os sistemas *Computer-Aided Diagnostic* (CADx) são responsáveis por diagnosticar (malignos ou benignos) um problema em uma imagem digital, auxiliando os especialistas com uma segunda opinião ([FILHO et al., 2016a](#)).

Os processos envolvidos nos sistemas CAD e CADx geralmente são compostos por quatro etapas. Primeira etapa, é aquisição que se refere à digitalização ou obtenção de regiões de interesse. Após a aquisição pode ser feita o pré-processamento da imagem ou região, cujo objetivo é suprimir os ruídos e melhorar o seu contraste. A segunda etapa, é extração de características e/ou segmentação, que consiste na localização das regiões suspeitas de conterem as massas. Normalmente são geradas muitas regiões suspeitas. Daí tem a necessidade da terceira etapa, a extração de características, para reavaliar as regiões segmentadas, reduzindo os falsos positivos. A última etapa, é a de classificação, que utiliza do conjunto de características previamente extraídas para informar se a região possui características de massa em maligno ou benigno.

A metodologia proposta irá atua em três etapas do sistema CADx, na etapa que consiste na obtenção da região de interesse e no pré-processamento da imagem, na etapa de extração das características através de análise de textura e na etapa de classificação do tecido mamário em maligno e benigno.

A análise de textura, é útil em aplicações, por se aproximar da avaliação feita pelo sistema visual humano. Os índices de diversidade são utilizados como descritores na extração de características de textura e são responsáveis por caracterizar as regiões de interesse, onde através do reconhecimento de padrões utiliza de múltiplos classificadores para a avaliar eminência da metodologia proposta.

8.1.1 Objetivos

De forma geral, em vista do contexto apresentado, este trabalho tem como principal objetivo desenvolver uma metodologia para classificação de regiões da mama em maligno e benigno.

8.1.1.1 Objetivos Específicos

1. Desenvolver e adaptar técnicas para caracterizar propriedades de textura;
2. Classificar as regiões de interesse em maligno e benigno;
3. Avaliar a viabilidade do uso de índices de diversidade como forma de extração de características; e
4. Construir uma metodologia capaz de sugerir ao especialista uma segunda opinião no diagnóstico do câncer de mama.

8.1.2 Organização do capítulo

Além da Introdução, este trabalho está dividido seguindo a ordem: na Seção 8.2, são apresentados os trabalhos relacionados; na Seção 8.3, apresenta-se o referencial teórico; na Seção 8.8, descreve-se a metodologia proposta; na Seção 8.9, são descritos os resultados e discussões da execução do projeto; e por fim, são apresentadas as conclusões e sugestões para trabalhos futuro na Seção 8.10.

8.2 Trabalhos Relacionados

Na literatura é possível encontrar trabalhos relacionados ao tema proposto. A seguir, tem-se resumos de alguns destes.

A metodologia proposta por [Rocha et al. \(2014\)](#), para a extração de características utilizou os índices de diversidades para detectar padrões de coocorrência de espécies. Utilizou os índices *Gleason* e *Menhinick* e na classificação em maligno ou benigno utilizou da Máquina de vetor de suporte (SVM). O melhor resultado obtido foi 86,66% de acurácia, 90% de sensibilidade, 83,33% de especificidade e área sob a curva ROC de 0,86.

No trabalho proposto em [Paramkusham et al. \(2015\)](#), apresenta a classificação dos tecidos mamários em normal, maligno e benigno, utilizou a extração de características de textura *Local Binary Pattern* (LBP) e classificou os tecidos mamários em normal e anormal. E utilizou de descritores de características geométricas na classificação em benignos e malignos. O melhor resultado é na classificação em normal e anormal tendo uma acurácia de 99,27%.

[Rouhi et al. \(2015\)](#), apresentou dois métodos automatizados para diagnosticar massas benignas e malignas extraídas de mamografias digitais. Inicialmente, é feita a etapa de segmentação usando região de crescimento automatizado cujo limiar é obtido por uma rede neural artificial treinada (RNA). No outro método a segmentação é realizada por uma *Cellular neural network* ([CHUA; YANG, 1988](#)) cujos parâmetros são determinados por um Algoritmo Genético (AG). As características de intensidade, textura e forma são extraídas das regiões segmentadas. O AG é usado para selecionar características mais relevantes. As RNAs são usadas para classificar as regiões como benignas ou malignas. Para avaliar o desempenho dos métodos propostos foram utilizados diferentes classificadores *Random Forest*, *Naive Bayes*, *SVM* e *K-Nearest Neighhor* (KNN). A metodologia apresentou resultado um percentual de 96,87% de sensibilidade, 95,94% de especificidade e 96,47% de acurácia.

Foram utilizados [Valarmathie et al. \(2016\)](#), um método para classificar massas mamográficas utilizando características de textura, forma e margem geométricas, selecionadas com classificador *Multilayer Perceptron* (MLP). As massas são segmentadas a partir da mamografia utilizando Limiares de Nível de Cinza e são extraídas as características. Os recursos são *fuzzificados* usando valores de associação difusos. O classificador MLP obteve uma precisão de 100% e 0,99 de curva ROC.

O CAD proposto por [Kanadam e Chereddy \(2016\)](#) modela automaticamente, região de interesse (ROI) identificadas à medida que ocorrem usando matriz esparsa, sendo nomeado como sparse-ROI. A massa

é detectada através de uma nova tecnologia de sparse-ROI e o classificador multi-SVM é usado para a classificação. O desempenho do classificador é avaliado na base de imagens MIAS. Para este propósito, dois algoritmos são utilizados com base nas matrizes estatísticas, Matriz de Co-ocorrência de Nível de Cinza (GLCM) e Matriz de Aura de Cível de Cinza (GLAM). A eficácia do classificador dos dois novos algoritmos desenvolvidos é avaliada em termos de precisão, sensibilidade, tamanho e tempo computacional. Os resultados do estudo tem redução no tempo computacional em 99,93% no GLCM e 75,73% no GLAM com a retenção concomitante da acurácia da classificação de 97,2%

Foi proposto por [Rabidas et al. \(2016\)](#), a introdução de um novo Padrão Binário Local Discriminativo Robusto (DRLBP) e Padrão Ternário Local Robustivo Discriminativo (DRLTP) ([GUO; ZHAO; PIETIKÄINEN, 2012](#)) para a classificação de massas benignas ou malignas. O método de Análise Discriminante Linear de Fisher é incorporado com características discriminantes, selecionadas pelo método de regressão logística para a classificação da massas em benignas e malignas. Para avaliar as características DRLBP e DRLTP é usada uma técnica de validação cruzada de dez vezes com 58 massas do banco de dados mini-MIAS, e o melhor resultado é observado com DRLBP tendo uma área sob a curva ROC de 0,982.

No trabalho de [Lima et al. \(2016\)](#), os autores apresentam um método para detectar e classificar lesões mamográficas. A proposta consiste em decompor cada imagem usando Wavelets multi-resolução. Os momentos de Zernike são extraídos de cada componente Wavelets. Com o uso dessa abordagem, puderam combinar características de textura e forma, foram aplicadas tanto na detecção quanto na classificação de lesões mamárias. A classificação utilizou das redes SVM com kernels modificados para otimizar as taxas de precisão. O método apresentou uma precisão de 94,11%.

Em [Li et al. \(2016\)](#), é proposto um método de classificação de massa em mamografias, com base em máscaras concêntricas e Textura discriminante. Utilizando de máscaras concêntricas, onde dividiram cada região de massa na região central e na região periférica. Em seguida para a Análise Discriminante Linear (LDA) com Textura tradicional, a textura discriminante é proposta. A falta de considerar as informações de classe na textura tradicional é melhorada. Finalmente, os recursos são extraídos com textura discriminante para a região central e a região periférica. Assim, o problema de desconsiderar as informações de layout espacial é aliviado. O método proposto é testado em 130 regiões de massa do banco de dados Digital Data base for Screening Mammography (DDSM). O método apresentou acurácia de 86,92% e a curva ROC de 0,91.

O trabalho de [Carvalho et al. \(2018a\)](#), apresenta uma metodologia para classificação de tecidos mamários em malignos e benignos. Utilizando na extração de características os índices filogenéticos. A classificação foi conduzida usando vários classificadores. Os resultados mostram que o método atinge 99,73% de precisão, 99,41% de sensibilidade, 99,84% de especificidade e uma curva de características de operação do receptor (ROC) com um valor de um ao usar imagens do Banco DDSM. Uma precisão de 100% é obtida ao usar o banco de dados de imagens do Mammography Imaging Analysis Society (MIAS).

Segundo o método proposto em [Wang et al. \(2018\)](#), utilizou Rede neural convolucional (CNN) genérica para extração de características e assumindo uma configuração de múltiplas visualizações, uma rede baseada em atenção é utilizada para selecionar automaticamente os recursos informativos da massa mamária. O mecanismo de atenção tenta fazer com que a CNN se concentre nas regiões relacionadas à semântica para um resultado de classificação mais interpretável. Então, os recursos de massa dos dados de múltiplas visualizações são efetivamente agregados por uma Rede Neural Recorrente (RNN) apresentando uma precisão de 0,85 e a curva ROC de 0,89.

A Tabela 47 resume as abordagens dos trabalhos relacionados expostos. Nota-se que houve utilização de vários métodos para extração dos atributos. Alguns trabalhos mostraram bons resultados. No entanto, ainda é necessário identificar técnicas que permitam melhorar e consolidar estes resultados. Podemos verificar que a classificação de massa mamária quanto a sua malignidade e benignidade é ainda um problema em aberto, e que medidas de textura se mostram promissoras para essa caracterização. Este trabalho tem como objetivo propor descritores de textura, levando em consideração o comportamento de espécies dentro de uma comunidade, grau de parentesco e riqueza de espécies.

Tabela 47 – Resumos dos trabalhos relacionados.

Trabalhos	Bases	Descritores / Classificados
(ROCHA et al., 2014)	DDSM	Índices de diversidades MVS
(PARAMKUSHAM et al., 2015)	IRMA	Local Binary Pattern Características geométricas SMV
(ROUHI et al., 2015)	MIAS DDSM	Random Forest Naive Bayes SVM KNN
(VALARMATHIE et al., 2016)	MIAS	Limiares de nível de cinza MLP
(KANADAM; CHEREDDY, 2016)	MIAS	sparse-ROI multi-SVM
(RABIDAS et al., 2016)	MIAS	DRLBP DRLTP
(LIMA et al., 2016)	IRMA	SVM ELM
(LI et al., 2016)	DDSM	Textura Discriminante na região central e região periférica KNN
(CARVALHO et al., 2018a)	DDSM MIAS	Índices Filogenéticos Random Forest MLP Neural Network SMO
(WANG et al., 2018)	BCDR	CNN RNN

8.3 Referencial Teórico

Nesta Seção serão apresentados conceitos pertinentes ao referencial teórico necessário ao entendimento e embasamento científico do método proposto.

8.3.1 Câncer de Mama

O câncer pode ser definido como uma doença degenerativa que se desenvolve no próprio organismo, resultante de um acúmulo de lesões no material genético das células, que induz ao processo de crescimento, reprodução e dispersão anormal destas com controle alterado sobre a proliferação e morte celular. O câncer é uma doença de proporções graves, colocando em risco a vida do indivíduo e podendo afetar qualquer parte de seu organismo, sem predisposição de idade e de maneira quase igualmente proporcional em ambos os sexos (AMORIM; SIQUEIRA, 2017).

O câncer de mama é mais comum após os 40 anos, a partir dessa idade deve-se realizar anualmente exames de mamografia. Embora, menos de 20% dos casos de câncer de mama são relacionados a fatores genéticos, recomenda-se que as mulheres que tenham casos na família próximos, começem a fazer a mamografia 10 anos antes da idade que seus familiares diagnosticaram a doença (BRUNO et al., 2018).

O câncer de mama é uma doença causada pela multiplicação desordenada de células da mama. Esse processo gera células anormais que se multiplicam, formando um tumor. Por isso, a doença pode evoluir de diferentes formas. Alguns tipos têm desenvolvimento rápido, enquanto outros crescem mais lentamente.

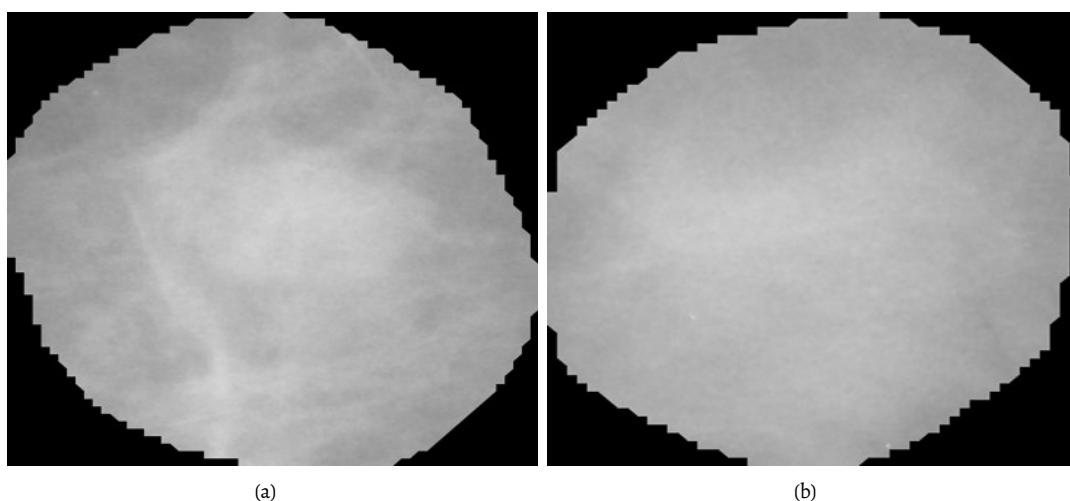
Esses comportamentos distintos se devem a características próprias de cada tumor, que pode ser maligno ou benigno ([INCA, 2019](#)).

A classificação entre tumores benignos e malignos é feita com base em três pontos principais, que são:

- A aparência;
- A estrutura, e
- O comportamento reprodutivo das células atingidas.

Nos tumores benignos as células sofrem uma pequena mutação, porém não é tão grave a ponto de criar uma expansão acelerada. Já os tumores malignos, as células sofrem grande mutação, essas células se multiplicam rapidamente e causam o surgimento de pequenos vasos que nutrem o tumor e corre o risco de que ele invada outros órgãos, formando novos tumores de crescimento desordenado. Segundo [ROCHA et al. \(2014\)](#) os passos fundamentais de processamento digital de imagens podem ser divididos em cinco etapas. A Figura 61 mostra a imagem de tumor benigno e maligno.

Figura 61 – Exemplos de imagens para câncer (a) benigno e (b) maligno.



O Ministério da Saúde oferece tratamento para câncer de mama, por meio do Sistema Único de Saúde (SUS). Um nódulo ou outro sintoma suspeito nas mamas deve ser investigado para confirmar se é ou não câncer de mama. Para a investigação, além do exame clínico das mamas, exames de imagem podem ser recomendados, como mamografia, ultrassonografia ou ressonância magnética. A confirmação diagnóstica só é feita, porém, por meio da biópsia, técnica que consiste na retirada de um fragmento do nódulo ou da lesão suspeita por meio de punções (extração por agulha) ou de uma pequena cirurgia. O material retirado é analisado pelo patologista para a definição do diagnóstico ([SAÚDE, 2019](#)).

8.3.2 Sistemas Computacionais de Auxílio à Detecção e Diagnóstico

Diagnóstico auxiliado por computador (CADx) pode ser definido como um diagnóstico utilizando os resultados de análises quantitativas automatizadas de imagens digitais, auxiliando os profissionais na tomada de decisões para o diagnóstico. É importante ressaltar que o computador é utilizado somente como uma ferramenta para obtenção de informação adicional, sendo o diagnóstico final sempre feito pelo profissional da área ([AZEVEDO-MARQUES, 2001b](#)).

A detecção de anomalias em imagens médicas, em geral é um procedimento demorado, propício a erros e a algum grau de subjetividade devido a várias razões. Sendo a existência de estruturas complexas e de grande número de imagens normais; a grande variação na aparência dos tecidos (mesmo os normais); a

sutileza das anormalidades; a superposição dos tecidos; a necessidade de grande sensibilidade e, ao mesmo tempo, de minimizar o retorno desnecessário dos pacientes.

O CADx alerta o profissional a examinar detalhadamente, padrões suspeitos detectados pelo sistema. Isto melhora a acurácia do diagnóstico e a consistência da interpretação da imagem, servindo como uma “segunda opinião” quando da tomada de decisões diagnósticas (PIRES et al., 2006).

8.4 Processamento Digital de Imagens

Processamento de imagens, de acordo com Filho e Neto (1999), é um conjunto de métodos e técnicas capazes de transformar imagens de forma adequada para uma melhor compreensão da visão humana ou análise computacional. Podemos classificar em três níveis:

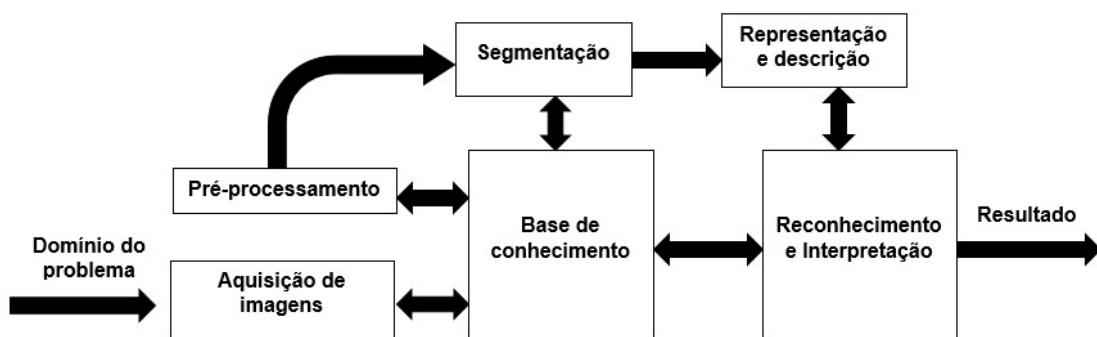
- Baixo nível:
 - Operações primitivas (redução de ruído, aumento de contraste, etc).
 - Entrada: Imagem - Saída: Imagem;
- Nível intermediário:
 - Segmentação, descrição e classificação de objetos.
 - Imagem - Atributos (bordas, contornos, nível de cinza).
- Alto nível:
 - Atribuir “sentido” há um conjunto de objetos reconhecidos.

Uma imagem pode ser definida como uma função bidimensional, onde x e y são coordenadas espaciais e a amplitude de f no par de coordenadas $f(x,y)$ é denominado intensidade ou nível de cinza. Quando x e y e a intensidade dos valores de f são finitos e discretos, tem-se então uma imagem digital. Esses elementos possuem uma localização e um valor particular, chamado de pixel (FERREIRA et al., 2017).

Para aplicações práticas, a imagem é uma função contínua, representada por medidas obtidas em intervalos regularmente espaçados. Os valores assumidos em cada ponto medido são quantificados em um número pertencente a uma escala de diferentes cores. Em imagens médicas, geralmente essas cores são relacionadas a níveis de cinza, sendo atribuído o valor zero à cor mais escura (preto) e o valor máximo M à cor mais clara da escala (branco) (FERREIRA et al., 2017).

Segundo ROCHA et al. (2014) os passos fundamentais de processamento digital de imagens podem ser divididos em cinco etapas conforme é mostrado na Figura 62.

Figura 62 – Etapas Fundamentais do Processamento Digital de Imagens



A primeira etapa, é à Aquisição de Imagens essa etapa responsável pela obtenção da representação digital da imagem;

A segunda etapa, é o Pré-processamento que consiste em tornar certas estruturas da imagem mais simples de serem definidas. Para tanto, utiliza-se de técnicas, tais como diminuição de ruídos, realce de contraste, filtros morfológicos e etc;

A terceira etapa, é a Segmentação consiste em qualquer operação que possa distinguir os objetos contidos na imagem ou de alguma forma isolando-os entre si;

A quarta etapa, é a Representação e Descrição é também chamada de extração de características. Tem como finalidade determinar as características básicas de cada objeto que resultem em informações importantes para discriminação entre classes distintas.

E finalmente a última etapa, que é o Reconhecimento e Interpretação tem por finalidade atribuir um rótulo a um objeto com base nos seus descritores através de uma base de conhecimento que foi construída na etapa anterior.

8.4.1 Pré-processamento

O pré-processamento consiste na utilização de mecanismo de processamento de imagens para o realce das imagens, aumento de contraste, retirada de possíveis ruídos entre outras para que aja uma nitidez nas características da imagem. Na metodologia proposta utilizou-se de técnicas visando o melhoramento do contraste das imagens.

A Equalização de Histograma tem como finalidade obter um histograma uniforme através do espalhamento da distribuição dos níveis de cinza ao longo de toda a escala de contraste, aumentando dessa maneira, a detectabilidade de aspectos da imagem (relacionados a contraste) ([NASCIMENTO et al., 2012](#)).

Como dedução dessa distribuição uniforme obtém-se uma imagem com melhor contraste, como demonstra a Figura 63 e a Figura 64 mostras os histogramas.

Figura 63 – (a) Imagem original e (b) Imagem Equalizada.

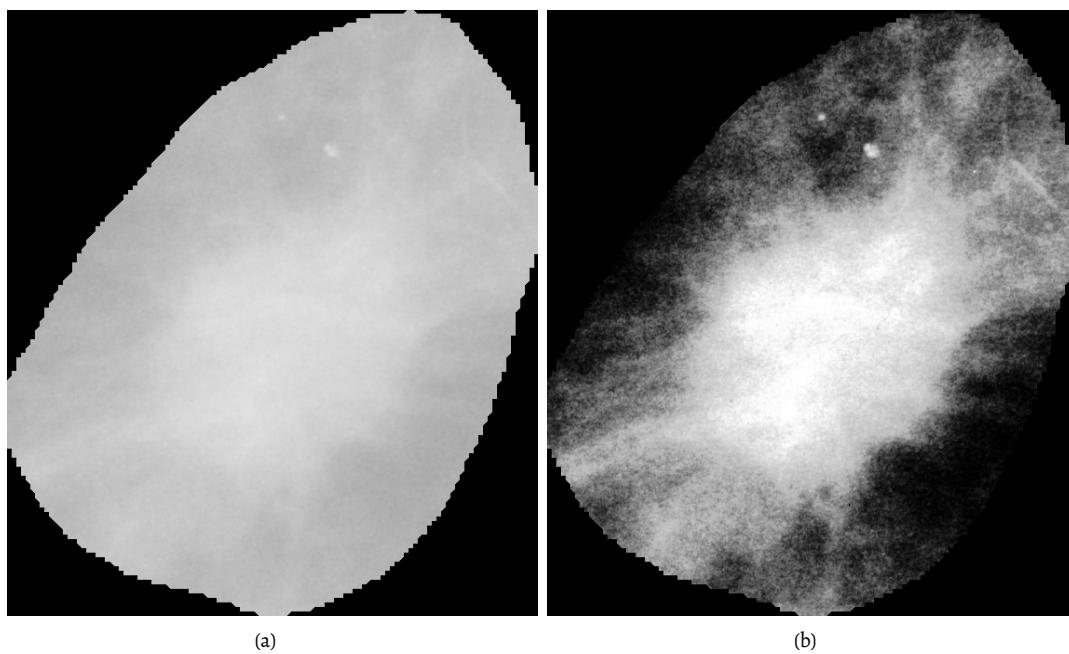
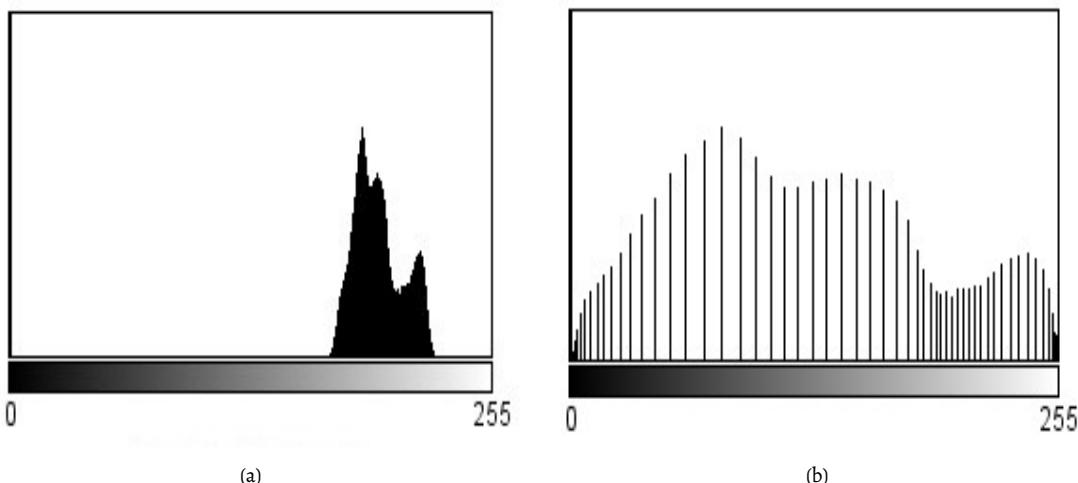


Figura 64 – (a) Histograma da Imagem Original e (b) Histograma da Imagem Equalizado.



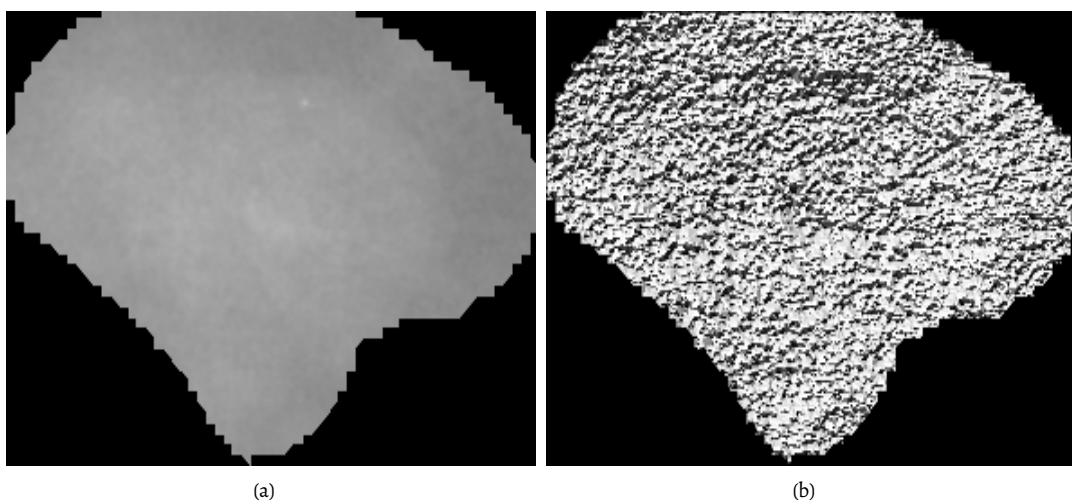
8.4.1.1 Local Binary Patterns

Segundo Pires e Neto (2015), a técnica *Local Binary Pattern* (LBP) é robusta em termos de variações nos níveis de cinza e que discrimina efetivamente uma larga gama de texturas rotacionadas. O LBP é um operador de textura invariante aos níveis de cinza e rotação, baseado em padrões locais binários. O LBP é calculado como mostrar a Equação 8.1,

$$LBP_{k,r} = \sum_{i=0}^{k-1} |S(N_i - N_c)| 2^i \quad (8.1)$$

onde, N_c é o pixel central e N_i são os pixels vizinhos, k é o numero de pontos e r o raio do circulo.

Figura 65 – (a) Imagem Original e (b) Imagem resultado após aplicação do LBP.



O LBP possui um padrão de textura vizinho ao redor de cada pixel. Onde o calcular é feito utilizando de uma janelar 3x3. O valor do pixel central é subtraído de cada um dos valores dos pixels vizinhos, o resultado da subtração é utilizado na função $S(x)$ que substituirá o valor anterior do pixel vizinho por 0 ou 1, dependendo do retorno da função $S(x)$. Cada valor da matriz binária é multiplicado pela sua respectiva

posição na matriz de pesos. O LBP e o resultado da soma de todos os valores resultantes das multiplicações. A Figura 65 mostra uma imagem original e LBP.

8.5 Índices de Diversidade

O termo diversidade, em ecologia, pode ser definido como a variedade e a variabilidade entre os organismos vivos e os complexos ecológicos em que ocorrem. Uma medida de diversidade é um parâmetro extremamente reducionista que objetiva expressar toda a complexidade estrutural de uma comunidade ecológica através de um único número (NASCIMENTO et al., 2012). A Tabela 48 resume como a metodologia utiliza os índices de diversidade.

Tabela 48 – Índice de Diversidade.

Metodologia Proposta	Ecológia
Região de Interesse da Imagem	Comunidade
Níveis de Cinza da Imagem	Espécies
Píxels da Imagem	Indivíduos

8.5.1 Índice de Shannon

Originado da teoria da informação (SHANNON; WEAVER, 1949), o Índice de Shannon assume que os indivíduos são randomicamente amostrados a partir de uma comunidade infinitamente grande, e que todas as espécies estão representadas na amostra. O Índice de Shannon é calculado conforme a Equação 8.2,

$$H' = - \sum_{i=1}^S p_i \ln p_i \quad (8.2)$$

onde, S é o total de espécies, p_i é a proporção de indivíduos pertencentes a espécie i na amostra, calculado com a Equação 8.3,

$$p_i = \frac{n_i}{N} \quad (8.3)$$

e n_i é o número de indivíduos na espécie i na amostra e N é o número total de indivíduos na comunidade.

8.5.2 Índice de Simpson

Proposto em 1949, por E. H. Simpson (SIMPSON, 1949), O Índice de Simpson (D) é a medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma comunidade infinitamente grande, pertencerem à mesma espécie. O Índice é calculado conforme Equação 8.4,

$$D = 1 - \sum_{i=1}^S p_i^2 \quad (8.4)$$

onde, p_i é a proporção de indivíduos pertencentes a espécie i na comunidade, calculado com a equação 8.3, n_i é o número de indivíduos na espécie i e N é o número total de indivíduos na comunidade. S é o total de espécies predominante na comunidade.

8.5.3 Índice de Margalef

Índice de Margalef (D) (MARGALEF, 1969) é usado na ecologia para estimar a biodiversidade de uma comunidade com base na distribuição numérica dos indivíduos das diferentes espécies em função do número total de indivíduos existentes na amostra analisada. O Índice é calculado conforme Equação 8.5,

$$D = \frac{(S - 1)}{\ln(N)} \quad (8.5)$$

onde, S é o número de espécies e $\ln(N)$ é o logaritmo do número de indivíduos pertencente a comunidade.

8.5.4 Índice de McIntosh

O Índice de McIntosh foi proposto por (MCINTOSH, 1967). A comunidade pode ser vista como um ponto em um hiper-volume S -dimensional e a distância euclidiana da comunidade para a origem pode ser utilizada como uma medida de diversidade. É uma medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma comunidade infinitamente grande pertencerem a mesma espécie (ARAÚJO et al., 2017). O índice de diversidade de McIntosh (D) é estimado através das seguintes Equações 8.6,

$$D = \frac{N - U}{N - \sqrt{N}} \quad (8.6)$$

onde, N é o número total de indivíduos na amostra e U é dado pela expressão 8.7,

$$U = \sqrt{\sum n_i^2} \quad (8.7)$$

onde, n_i é o número de indivíduos nas espécies i e o somatório é realizada sobre todas as espécies. U é a distância euclidiana da comunidade desde a origem quando plotada em um hiper volume S -dimensional.

8.5.5 Índice de Camargo

O índice de Camargo (CAMARGO, 1993) é a relação entre pares de espécies i e j , definido pela Equação 8.8,

$$E = 1 - \left[\sum_{i=1}^s \sum_{j=i+1}^s \frac{|p_i - p_j|}{S} \right] \quad (8.8)$$

onde, p_i é a proporção de espécies i na amostra, p_j é a proporção de espécies j na amostra e S é o número total de espécies.

8.5.6 Índice de Smith & Wilson B

O índice de B de Smith e Wilson (SMITH; WILSON, 1996). Baseia-se na variação de abundância. Onde variância é calculada usando abundâncias de logaritmo, o que significa que o índice examina as diferenças proporcionais entre as espécies. Com a obtenção da variância, é multiplicada pelo fator $-2 / \pi \arctan$ para dar um valor entre o intervalo de 0 a 1. O índice é definido pela Equação 8.9,

$$E = 1 - \left[\frac{2}{\pi \arctan \left\{ \frac{\sum_{i=1}^s (\ln n_i - \sum_{j=1}^s \frac{\ln n_j}{S})^2}{S} \right\}} \right] \quad (8.9)$$

sendo, n_i é o número de indivíduos nas espécies i , n_j é o número de indivíduos nas espécies j ; e S é o número total de espécies.

8.5.7 Índice de Smith & Wilson 1-D

Índice de uniformidade de Smith e Wilson 1-D (SMITH; WILSON, 1996), utilizar de dominância D (Simpson), converter o índice de Simpson (D) de várias maneiras. A mais comum é usar o complemento de D como índice de diversidade e para dados contínuos ou com números grandes de cortes discretos. Definido pela Equação 8.10,

$$E = \frac{1 - D}{1 - \frac{1}{S}} \quad (8.10)$$

onde, D é o índice de diversidade de Simpson representado pela equação 8.4 e S é o número total de espécies.

8.5.8 Índice de Smith & Wilson -lnD

O índice de uniformidade de Smith e Wilson 1 / D (SMITH; WILSON, 1996), converter o índice de dominância de Simpson em uma diversidade de índice. A medida equivalente de uniformidade é a anterior. O índice é definido na Equação 8.11,

$$E = \frac{-\ln D}{\ln S} \quad (8.11)$$

onde, D é o índice de diversidade de Simpson representado pela equação 8.4 e S é o número total de espécies.

8.5.9 Índice de Shannon maximum

Este é simplesmente o valor máximo que o índice de Shannon 8.5.1 poderia produzir para o dado conjunto de dados e é dado por $\ln(S)$, onde S é o número total de espécies.

8.5.10 Índice de Shannon minimum

Este é simplesmente o valor mínimo que o índice de Shannon 8.5.1 poderia produzir para o dado conjunto de dados e é dado pela Equação 8.12,

$$S_{min} = \ln(N) - \left\lfloor \frac{(N - S + 1) \ln(N - S + 1)}{N} \right\rfloor \quad (8.12)$$

sendo, N é o número total de indivíduos na amostra e S é o número total de espécies.

8.6 Reconhecimento de padrões

O ato de reconhecer é utilizado diariamente por todos os seres vivos, através dos sentidos (visão, olfato, paladar, audição e tato). Esse ato de reconhecer utiliza-se de padrões que ao longo da vida são aprendidos. Para reconhecer algum objeto, terá que ter um breve conhecimento sobre o mesmo, no caso o padrão. Se esse padrão não for de conhecimento, então o objeto não será reconhecido. Na computação utiliza-se desse reconhecimento em diversas áreas.

Segundo Filho et al. (2016a) o reconhecimento de padrão envolve duas tarefas: classificação e reconhecimento. A classificação divide um conjunto de dados obedecendo métricas, formando classes ou grupos. Reconhecimento trata-se de novo dado reconhecido e atribuído em uma classe gerada.

Para a análise de padrões, na metodologia proposta, foram escolhidas quatro classificadores levando em consideração sua popularidade, eficácia, facilidade de adaptação nos experimentos e sua integração com a ferramenta de extração de características em imagens, sendo eles: Random Forest (WRIGHT; WAGER; PROBST, 2016), Sequential Minimal Optimization (SMO) (PLATT, 1998), Multi Layer Perceptron (MLP) (PAL; MITRA, 1992) e K-Nearest Neighbor (KNN) (KELLER; GRAY; GIVENS, 1985) que no weka é chamado de IBK. Os parâmetros utilizados para cada classificador, foram os padrões da ferramenta Weka.

8.7 Métricas de Validação

As métricas de valiação baseiam em estatísticas como, Acurácia (A), Sensibilidade (S), Especificidade (E), Curva ROC e Kappa. A matriz de confusão oferece uma hipótese das medidas efetivas do modelo de classificação, mostrando o número de classificações corretas versus as classificações preditas para cada classe, como mostra a Tabela 49.

Acurácia (A) (Equação 8.13), é a proporção de acertos, ou seja, o total de verdadeiramente positivos e verdadeiramente negativos, em relação a amostra estudada.

$$A = \frac{VP + VN}{VP + VN + FP + FN} \quad (8.13)$$

Tabela 49 – Matriz de Confusão.

Resultado do Teste	Positivo	Negativo
Negativo	Verdadeiros Positivos - VP	Falsos Negativos - FN
Positivo	Falsos Positivos - FP	Verdadeiros Negativos - VN

A sensibilidade (S) (Equação 8.14), é a capacidade de um teste diagnóstico identificar os verdadeiros positivos nos indivíduos verdadeiramente doentes.

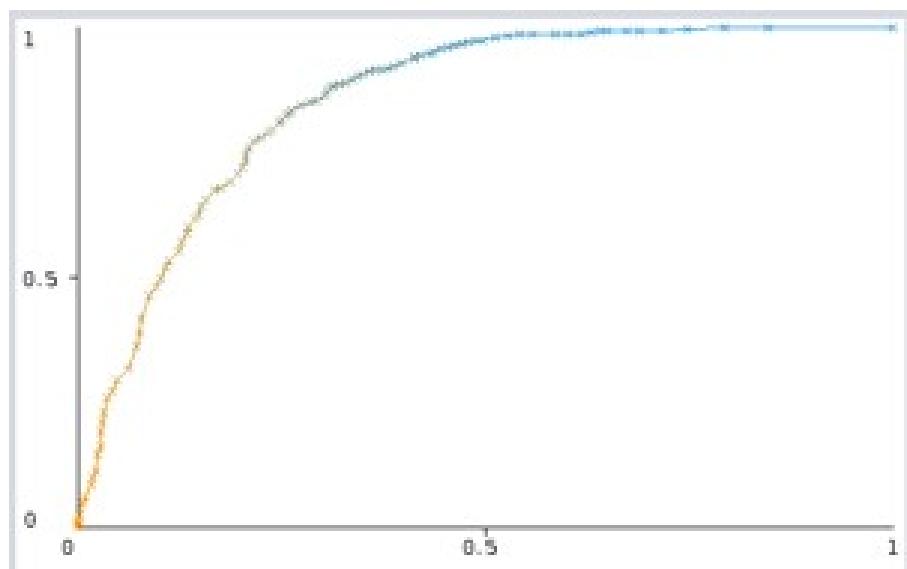
$$S = \frac{VP}{VP + FN} \quad (8.14)$$

Especificidade (E) (Equação 8.15), é a capacidade de um teste diagnóstico identificar os verdadeiros negativos nos indivíduos verdadeiramente sadios. Quando um teste é específico raramente cometerá o erro de dizer que pessoas sadias são doentes.

$$E = \frac{VN}{VN + FP} \quad (8.15)$$

A curva *Receiver Operating Characteristic* (ROC) é uma métrica de avaliação que compara o desempenho de duas ou mais modalidades de imagens. A área sobre a curva ROC representa a probabilidade de que, dado um caso positivo e um negativo, a regra do classificador vai ser mais elevada para o caso positivo. Quanto maior for a curva ROC (Figura 66), maior é a probabilidade do sistema fazer uma decisão correta. A curva ROC representa a dependência entre a sensibilidade e a especificidade de um classificador. Cada ponto é representado por um par de valor, sensibilidade e especificidade, e a linha diagonal um classificador que não consegue discriminar, devido o número de VP ser igual ao percentual de FP.

Figura 66 – Área sobre a curva ROC.



A estatística Kappa é freqüentemente usada para testar a confiabilidade entre avaliadores. A importância da confiabilidade do avaliador reside no fato de representar a medida em que os dados coletados no estudo são representações corretas das variáveis medidas. A medição da extensão em que os coletores de dados (avaliadores) atribuem a mesma pontuação à mesma variável é chamada de confiabilidade entre avaliadores. Embora existam uma variedade de métodos para medir a confiabilidade entre avaliadores, tradicionalmente foi medida como concordância percentual, calculada como o número de pontuações de concordância dividido pelo número total de pontuações (MCHUGH, 2012).

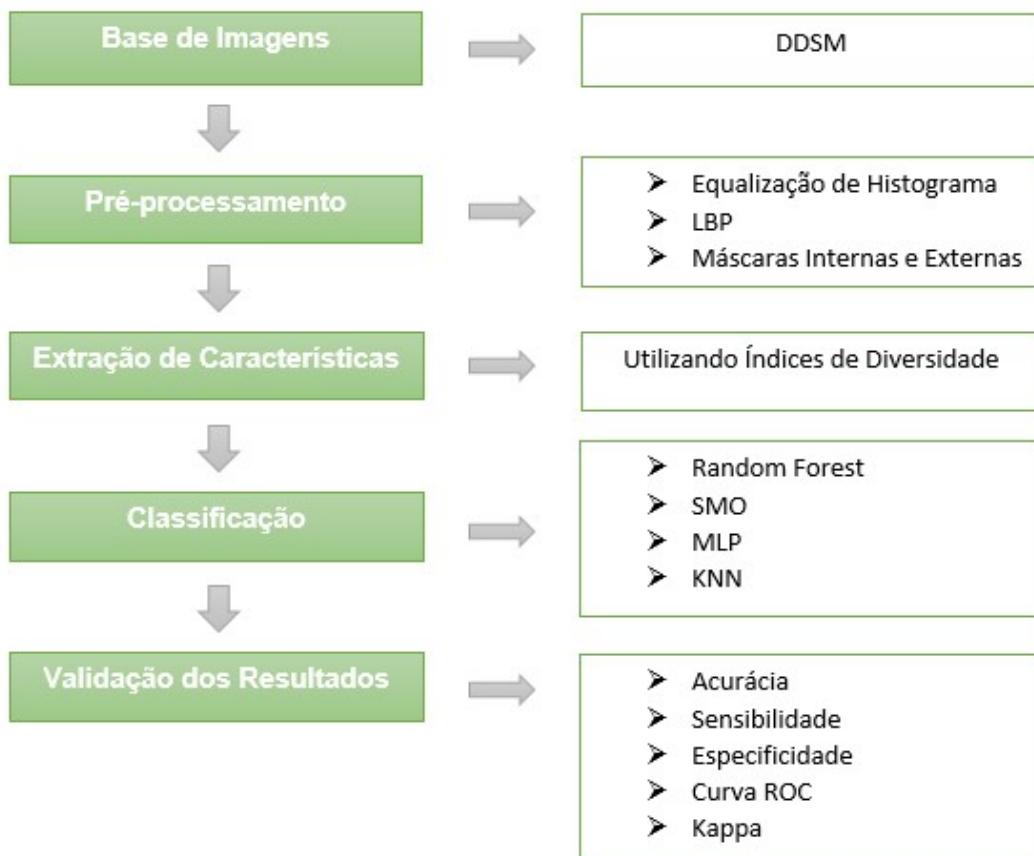
Tabela 50 – Interpretação para os valores do índice Kappa.

VALOR DO KAPPA	NÍVEL DE CONCORDÂNCIA
≤ 0	Não existe concordância
Entre 0 e 0,2	Concordância Mínima
Entre 0,21 e 0,4	Concordância Razoável
Entre 0,41 e 0,6	Concordância Moderada
Entre 0,61 e 0,8	Concordância Substancial
Entre 0,81 e 1	Concordância Perfeita

8.8 Metodologia

A metodologia proposta neste trabalho segue as etapas da Figura 67. Na primeira etapa, é feita a aquisição das imagens oriundas da base *Digital Data base for Screening Mammography* (DDSM) (HEATH et al., 2000). Na segunda etapa, é o pré-processamento, através de técnicas de processamento de imagem. Na terceira etapa, é a extração de características, através dos índices de diversidade. Na quarta etapa, é a classificação em benignos ou malignos, foram utilizados os classificadores Random Forest, SMO, MLP e KNN. O processo é finalizado com a validação dos resultados.

Figura 67 – Metodologia proposta.



8.8.1 Aquisição das Imagens

A base de imagens DDSM é uma base pública contendo imagens de mamografias, que tem como objetivo facilitar a pesquisa e desenvolvimento de algoritmos para ajudar no diagnóstico de anomalias da mama

(HEATH et al., 2000).

Este trabalho utiliza *Region of Interest* (ROI) adquiridas da base DDSM, onde cada ROI possui tamanhos diferentes e apenas uma região de massa. Para a concretização desse trabalho, foram utilizadas 1155 ROIs de imagens de mamografias; sendo 625 ROIs com a presença de massa maligna e 530 ROIs com presença de massa benigna.

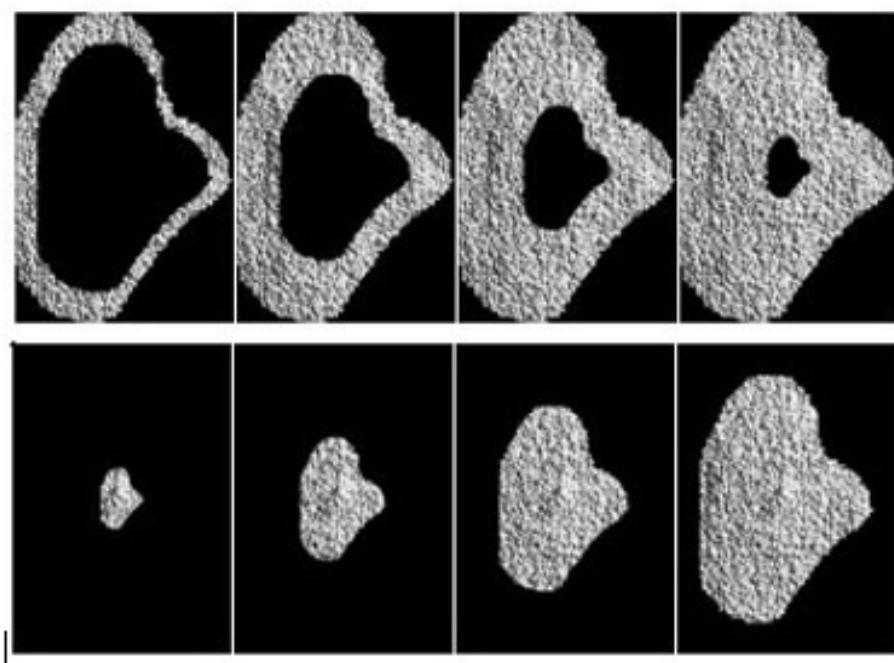
8.8.2 Pré-processamento

Na metodologia utiliza-se técnica de Equalização de Histograma, para um espalhamento da distribuição dos níveis de cinza ao longo de toda a escala de contraste, com a finalidade de aumentar, a detectabilidade de aspectos da imagem.

A técnica de LBP foi utilizada para gerar uma nova representação da imagem, com intuito de evidenciar propriedades de textura, que apenas com a imagem normal não fossem tão evidentes.

A metodologia utiliza de máscaras internas e externas (Figura 68). A proposta desta abordagem é baseada em descobrir padrões de diversidade nas áreas perto das bordas e internamente. Neste trabalho, os valores de 20%, 40%, 60% e 80% foram definidos para aplicação escalonada. Os tamanhos das máscaras são empiricamente definidos para permitir uma análise viável de regiões em partes isoladas da ROI. As

Figura 68 – Máscaras Internas e Externas.



máscaras internas foram geradas a partir da diminuição da escala em relação a original pelo centro de massa e as sucessivas máscaras foram adquiridas a partir das suas anteriores na sequência até a mais interna. A máscara externa é similar à abordagem com máscara interna. As máscaras são formadas pela diferença entre as máscaras internas, onde a primeira foi criada pela junção entre a primeira e a segunda interna, a segunda e a terceira interna. As demais máscaras externas foram criadas seguindo a sequência das diferenças até o último par de máscaras internas.

8.8.3 Extração de Características

Após o pré-processamento das imagens, inicia-se à fase de extração de características baseada em textura. Para descrever a textura dos nódulos, são utilizados os índices de diversidade. Esses índices descrevem as texturas das imagens e os atributos fornecidos buscam reconhecer as medidas de homogeneidade ou heterogeneidade da ROI. Servindo com base para classifica em maligno ou benigno.

8.8.4 Classificação e validação dos Resultados

Para a classificação, foi utilizado a ferramenta suíte de algoritmos de mineração de dados e Aprendizado de Máquina WEKA, que contém algoritmos para pré-processamento de dados, classificação, regressão, agrupamento, regras de associação e visualização.

Foram selecionados os algoritmos Random Forest, SMO, MLP e KNN, utilizando os parâmetros com os valores padrões em conjunto com a validação cruzada de k-folds, sendo $k = 10$; esse método que tem como finalidade dividir as características em 10 grupos, de forma a realizar o treino em 9 grupos e utilizando um grupo para testes, são realizados 10 cruzamentos sempre trocando o grupo de teste, ao final é gerado uma média, onde se configura o resultado.

Por fim, após a etapa de reconhecimento de padrão, a fim de se considerar a presença ou ausência de massas malignas e benignas em imagens de mamografia, é necessário a utilização de métricas para validar os resultados e analisar possíveis melhorias do mesmo. A metodologia proposta, utiliza métricas de avaliação baseadas em estatísticas, como, Acurácia (A), Sensibilidade (S), Especificidade (E), Coeficiente Kappa e Área sobre a Curva (ROC).

8.9 Resultados e Discussões

Para os testes realizados neste trabalho, utilizou-se a base DDSM. A extração de características de textura foi realizada a partir dos índices de diversidade especificados na Seção 8.5 e a classificação das massas em maligno e benigno, utilizou os classificadores e parâmetros definidos na Seção 8.6, o método de teste foi a validação cruzada com $k = 10$.

A Tabela 51 apresenta os resultados para as imagens originais, com e sem Equalização de Histograma. Cada massa é representada por 10 características, isto é, para os 10 (dez) índices propostos nesse método.

Tabela 51 – Resultados dos testes utilizando imagens normais

Imagens Originais					
Sem Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	64,24	59,81	68,00	0,700	0,278
SMO	65,45	50,75	77,92	0,643	0,292
MLP	64,33	45,28	80,48	0,710	0,264
KNN	61,21	60,38	61,92	0,607	0,222
Com Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	64,50	60,19	68,16	0,712	0,284
SMO	65,37	50,94	77,60	0,643	0,290
MLP	64,59	43,58	82,40	0,714	0,267
KNN	60,26	56,98	63,04	0,589	0,200

De acordo com a Tabela 51, o classificador SMO obteve o melhor resultado nas imagens sem realce, com uma taxa de acurácia de 65,45%, uma sensibilidade de 50,75%, especificidade de 77,92%, uma curva ROC de 0,643 e uma Kappa de 0,292, indicando que o classificador apresenta uma razoável capacidade de diagnóstico, de acordo a Tabela 50. O resultado menos significativo foi do classificador KNN nas imagens comum com realce, com taxa de acurácia de 60,26%, curva ROC de 0,589 e Kappa de 0,2001. Como podemos observar na Tabela 51 o classificador KNN foi capaz de identificar um número baixo de casos com doença, como mostra a sensibilidade de 56,98% e poucos casos sadios foram classificados como doentes, especificidade de 63,04%.

A Tabela 52 demostra os resultados para as imagens originais e máscara internas e externa, com e sem Equalização de Histograma. Cada massa é representada por 90 características, isto é, para os 10 (dez) índices propostos nesse método, sendo aplicado na imagem original obtendo 10 características, nas máscaras internas (20%,40%,60% e 80%) obtendo 40 características e externas (20%,40%,60% e 80%) obtendo 40 características.

Tabela 52 – Resultados dos testes utilizando imagens normais e máscaras.

Imagens originais e Máscaras					
Sem Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	69,18	66,42	71,52	0,758	0,379
SMO	69,61	66,42	72,32	0,694	0,387
MLP	68,23	67,55	68,80	0,721	0,362
KNN	60,78	58,30	62,88	0,594	0,211
Com Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	69,09	65,09	72,48	0,761	0,376
SMO	70,48	66,60	73,76	0,702	0,404
MLP	66,15	65,28	66,88	0,701	0,320
KNN	61,90	61,32	62,40	0,612	0,236

Entre os resultados apresentados na Tabela 52, o classificador SMO obteve o melhor resultado nas imagens com realce, com uma taxa de acurácia de 70,48%, uma sensibilidade de 66,60%, especificidade de 73,76%, uma curva ROC de 0,702 e uma Kappa de 0,404. O pior resultado foi do classificador KNN nas imagens sem realce, com taxa de acurácia de 60,78%, curva ROC de 0,594 e Kappa de 0,211. O classificador identificou um número baixo de casos com doença, como mostra a sensibilidade de 58,30% e poucos casos sadios foram classificados como doentes, especificidade de 62,88%.

A Tabela 53 representa os resultados para as imagens LBP, com e sem Equalização de Histograma. Cada massa é representada por 10 características.

Tabela 53 – Resultados dos testes utilizando imagens LBP

Imagens LBP					
Sem Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	77,06	78,68	75,68	0,840	0,540
SMO	72,03	91,70	55,36	0,735	0,455
MLP	75,93	78,30	73,92	0,827	0,518
KNN	71,95	72,83	71,20	0,734	0,438
Com Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	76,19	78,30	74,40	0,835	0,523
SMO	71,08	94,91	50,88	0,729	0,440
MLP	74,37	77,17	72,00	0,804	0,488
KNN	67,27	66,23	68,16	0,593	0,343

Entre os resultados obtidos na Tabela 53, as imagens sem realce em conjunto com o classificador Random Forest obtiveram o melhor resultado, com uma taxa de acurácia de 77,06%, uma sensibilidade de 78,68%, especificidade de 75,68%, uma curva ROC de 0,840 e uma Kappa de 0,540. A imagens com realce em conjunto com o classificador KNN tiveram o pior resultado, com taxa de acurácia de 67,27%, curva ROC de 0,593 e Kappa de 0,343. Como podemos observar na Tabela 53 o classificador identificou um número baixo de casos com doença, como mostra a sensibilidade de 66,23% e poucos casos sadios foram classificados como doentes, especificidade de 68,16%.

A Tabela 54 mostra os resultados para as imagens LBP e máscara interna e externa, com e sem Equalização de Histograma. Cada massa é representada por 90 características, isto é, para os 10 (dez) índices propostos nesse método, sendo aplicado na imagem original obtendo 10 características, nas máscaras internas (20%,40%,60% e 80%) obtendo 40 características e externas (20%,40%,60% e 80%) obtendo 40 características.

Tabela 54 – Resultados dos testes utilizando imagens LBP e máscaras

Imagens LBP e Máscaras					
Sem Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	77,92	81,51	74,88	0,852	0,559
SMO	74,46	88,68	62,40	0,755	0,498
MLP	77,75	84,72	71,84	0,833	0,558
KNN	71,00	70,19	71,68	0,717	0,417
Com Equalização de Histograma					
Classificador	A(%)	S(%)	E(%)	ROC	Kappa
Random Forest	77,40	80,19	75,04	0,851	0,548
SMO	76,10	82,64	70,56	0,766	0,525
MLP	72,64	75,09	70,56	0,806	0,453
KNN	67,10	65,47	68,48	0,680	0,339

Dos resultados exibidos na Tabela 54, o classificador Random Forest em conjunto com as imagens sem realce obteve o melhor resultado, com uma taxa de acurácia de 77,92%, uma sensibilidade de 81,51%, especificidade de 74,88%, uma curva ROC de 0,852 e uma Kappa de 0,559. O classificador KNN em conjunto com as imagens com realce tiveram o resultado menos significativo, com taxa de acurácia de 67,10%, curva ROC de 0,680 e Kappa de 0,339. Onde foi observado na Tabela 54, o classificador KNN foi capaz de identificar um número baixo de casos com doença, como mostra a sensibilidade de 65,10% e poucos casos sadios foram classificados como doentes, especificidade de 68,48%.

Na Tabela 55 mostra uma breve comparação entre os resultados encontrados utilizando a metodologia propostas e alguns trabalhos citados na Seção 8.2, que realizam a classificação de tecidos mamários em malignos e benignos. Portanto, a comparação com os trabalhos é meramente subjetiva, pois os casos analisados variam, assim como as bases de imagens utilizadas, sendo estes, fatores primordiais para uma comparação de confiança.

Tabela 55 – Comparaçao da metodologia com os trabalhos relacionados.

Trabalhos	Bases	A(%)	S(%)	E(%)	ROC	Kappa
(ROCHA et al., 2014)	DDSM	86,66	90	83,33	0,86	-
(PARAMKUSHAM et al., 2015)	IRMA	89,05	-	-	-	-
(ROUHI et al., 2015)	MIAS DDSM	96,47	96,87	95,94	-	-
(VALARMATHIE et al., 2016)	MIAS	100	-	-	0,99	-
(KANADAM; CHEREDDY, 2016)	MIAS	97,2	-	-	-	-
(RABIDAS et al., 2016)	MIAS	-	-	-	0,982	-
(LIMA et al., 2016)	IRMA	94,11	-	-	-	-
(LI et al., 2016)	DDSM	86,92	87,18	86,54	0,91	-
(CARVALHO et al., 2018a)	DDSM MIAS	99,73 100	99,41 100	99,84 100	1,00	-
(WANG et al., 2018)	BCDR	85,00	-	-	0,89	-
Metodologia	DDSM	77,92	81,51	74,88	0,852	0,559

No que diz respeito às métricas de desempenho utilizadas neste trabalho, o ideal para um sistema CADx é ter um bom equilíbrio entre as três métricas de avaliação (acurácia, sensibilidade e especificidade), pois, um bom método deve ser capaz de classificar com sucesso, tantos os casos de positivos (que possuem a doença) como os casos de negativos (que não possuem a doença). Fica claro que na presente metodologia além dos valores de validação apresentarem equilíbrio, e de acordo com a curva roc e a kappa, motivando sua aplicação de acordo com os seus valores

8.9.1 Discussões

Aqui apresentamos e discutimos os resultados produzidos pelos testes realizados utilizando o método proposto para diagnosticar massa mamária em maligno e benigno. Também realizou-se uma comparação do melhor resultado obtido com outros expostos na literatura.

Assim, utilizou-se técnicas para melhorar o contraste dos objetos de interesse, para prover uma melhor descrição da textura, o LBP se mostrou eficaz, pois em todos os teste, melhoraram os resultados alcançados. Convém destacar a eficiência das máscaras em abranger padrões em áreas de borda e interna.

8.10 Conclusão

O presente estudo apresentou um método automático, com o uso dos Índices de Diversidade, junto com reconhecimentos de padrões, capaz de discriminar e classificar, regiões de tecidos da mama em maligno e benigno.

Os índices de diversidade, apresentaram-se moderado na tarefa de caracterização das regiões em maligno e benigno, apresentando uma taxa de acurácia de 77,92%, sensibilidade de 81,51%, especificidade de 74,88%, curva roc de 0,852 e kappa de 0,5592.

Os resultados obtidos nas imagens LBP, demonstraram o desempenho próspero das técnicas de extração de textura pelos índices de diversidade com o classificador Random Forest, sendo confirmados pelo resultados da estatística Kappa, que é representado na Tabela 50.

No entanto, este trabalho contribuiu de forma positiva em duas distintas áreas. Sendo que na área da saúde, disponibilizou uma metodologia automática para auxílio no diagnóstico de tecidos mamários em maligno e benigno, que apesar de não fornecer excelentes resultados mostram-se promissor. E na computação, com a adaptação de técnicas de outras áreas do conhecimento que serviram como descritores de textura, podendo então, serem utilizados em caracterizações em outros tipos de imagens.

E ainda, como sugestão para trabalhos futuros, pretendendo-se:

- Aumentar a quantidade e variabilidade das amostras de lesões utilizando mais bases de imagens;
- Aplicar os índices propostos para investigação de outras doenças; e,
- Aplicar os índices propostos em conjunto com outros descritores, inclusive, com abordagens de aprendizado profundo.

Análise filogenética aplicada a exploração de textura de imagens de pele para auxílio no diagnóstico do melanoma

Autores: Antônio Oseas de Carvalho Filho; Wellington João da Silva Sousa Júnior.

Resumo: O câncer de pele é definido pelo crescimento anormal e descontrolado das células que compõem a pele. Entre os cânceres de pele, o menos frequente é o melanoma, porém possui o pior prognóstico, sendo o mais letal e correspondendo a 3% dos 33% de cânceres de pele no Brasil. Apesar de ser o mais nocivo, se houver uma detecção precoce as chances de cura são de 95%. O método aplicado para o diagnóstico precoce do melanoma é o exame dermatológico. Este trabalho mostra uma metodologia de classificação em melanoma e não melanoma, com o uso de descritores de textura baseados em índices de diversidade filogenética para extração de características, no reconhecimento de padrões é realizado a classificação usando Random Forest, SMO, MLP e o IBK. Como resultado, foi obtido uma acurácia de 99,8%, sensibilidade 100% e especificidade de 99,9%. O uso de índices de diversidade filogenéticos para detectar padrões em imagens de pele, mostrou-se relevante para categorizar o melanoma e auxiliar os especialistas no diagnóstico da doença.

Keywords: Câncer de Pele, Precoce, Diagnóstico.

9.1 Introdução

A interação entre a pele e a luz solar é inevitável. O potencial de dano depende do tipo e da duração da exposição. A luz solar tem efeitos profundos sobre a pele, e está associada a uma variedade de doenças ([SAMIS, 2006](#)). Dentre essas doenças, o câncer de pele.

O câncer de pele responde por 33% de todos os diagnósticos desta doença no Brasil, sendo que o Instituto Nacional do Câncer (INCA) registra, a cada ano, cerca de 180 mil novos casos ([INCA, 2018b](#)). Entre os cânceres da pele, o melanoma é o menos frequente, porém o mais letal. Segundo o [INCA \(2018b\)](#), no Brasil, embora o câncer de pele seja o mais frequente tendo 30% de todos os tumores malignos registrados, apenas 3% são do tipo melanoma, com 6.260 novos casos diagnosticados com essa doença, as mulheres são mais frequentes

com um total de 3.340 casos e os homens 2.940, o número de mortes são de 1.794, sendo 1.012 homens e 782 mulheres.

A detecção precoce do câncer de pele aumenta a chance de cura dos pacientes, sendo realizado através de exames dermatológicos. Com o uso deste, a incidência de câncer de pele pode ser tratada de forma precoce, porém, alguns fatores podem comprometer os prognósticos, como a inexperiência do especialista, a qualidade do exame, diagnósticos de doenças parecidas na pele, acarretando em possíveis erros na detecção da doença, como afirma [Bakos \(2006\)](#) e [INCA \(2018b\)](#).

No diagnóstico dermatológico, as lesões são examinadas clinicamente, utilizando primeiramente a técnica de análise das características e textura, para então diagnosticá-las e tratá-las. Para facilitar este processo, os dermatologistas podem dispor de sistemas computacionais, que analisam as características das lesões de forma mais precisa, utilizando imagens digitais, para auxiliar no seu diagnóstico ([OLIVEIRA, 2012a](#)). A Figura 1 apresenta uma imagem do tipo melanoma.



Figura 69 – Imagem de pele do tipo melanoma

O processamento digital de imagens tem um grande potencial nas diversas modalidades de imagens médicas. A análise e interpretação computadorizada de imagens médicas são fundamentais no auxílio a um diagnóstico preciso e no estudo da patologia da doença. Essa área ficou conhecida como diagnóstico por imagem. Com advento da inteligência artificial e sistemas inteligentes, o diagnóstico pode ser incorporado na análise informatizada da imagem ([SOARES, 2008](#)).

Como visto em [Azevedo-Marques \(2001b\)](#), os sistemas de diagnóstico auxiliados por computador (*computer-aided diagnosis, CADx*) são responsáveis por quantificar as características das imagens e reconhecer os padrões e classificar as imagens médicas em normais ou anormais. Geralmente são utilizados acoplados em equipamentos médicos com o intuito de auxiliar o especialista no diagnóstico de doenças.

Neste trabalho foi desenvolvido um sistema *CADx* para o diagnóstico automático do melanoma, categorizando em melanoma ou não. Para isso, são extraídos as características das imagens baseados em textura.

Por fim, é feito o reconhecimento de padrões com múltiplos classificadores para avaliar a relevância da pesquisa proposta, com base nos resultados obtidos em cada classificador e determinar se o trabalho proposto é relevante para que os especialistas tenham uma segunda opinião no diagnóstico da doença.

9.1.1 Objetivos

Este trabalho tem como objetivo geral desenvolver uma abordagem para a classificação de regiões em melanoma e não melanoma, utilizando descritores de textura baseado nos índices de diversidade filogenética.

Possuem como objetivos específicos:

1. Desenvolver e adaptar técnicas baseadas no comportamento filogenético para caracterização da textura das regiões;

2. Analisar a usabilidade dos descritores filogenéticos na categorização de imagens dermatológicas;
3. Construir uma metodologia para que os especialistas tenham uma segunda opinião no diagnóstico do melanoma.

9.1.2 Organização do capítulo

Além da introdução, este trabalho está organizado em outras 5 Seções. A Seção 2 descreve todo o embasamento teórico para o entendimento da metodologia utilizada no trabalho. Na Seção 3 são mostrados os principais trabalhos relacionados. Na Seção 4 é abordado todo o desenvolvimento da pesquisa, tendo como início a aquisição das imagens, extração de características e de textura baseado em índice de diversidade filogenética, classificação e por último a validação da metodologia proposta. O Seção 5 detalha os resultados obtidos com a metodologia proposta. Por fim, na Seção 6 é apresentado a conclusão do trabalho proposto.

9.2 Referencial Teórico

Para melhor entendimento deste trabalho, faz-se necessário a descrição de conceitos relacionados ao tema. Esta Seção aborda todo o embasamento teórico utilizado na metodologia proposta para facilitar sua compreensão. Diante disto, serão abordados os seguintes tópicos: Melanoma, processamento digital de imagens, índices de diversidade filogenética, reconhecimento de padrões e métricas de validação.

9.2.1 Melanoma

O câncer de pele é uma doença que é provocada pelo crescimento anormal e descontrolado das células que compõem a pele. Essas células se dispõem formando camadas e, de acordo com as que forem afetadas, são definidos os diferentes tipos de câncer. Os mais comuns são os carcinomas basocelulares e os espinocelulares. O câncer de pele mais agressivo é conhecido como melanoma, sendo o mais raro e nocivo entre os outros tipos de cânceres de pele ([INCA, 2018b](#)).

O câncer de pele é o tipo de câncer mais incidente em ambos os sexos no Brasil. Entretanto, as lesões são de fácil diagnóstico clínico e podem atingir índices de curas superiores a 95%, desde que diagnosticados nos estágios iniciais e tratados corretamente ([GOMES, 2013](#)).

O melanoma tem origem nos melanócitos, as células que produzem melanina, o pigmento que dá cor à pele. Normalmente, surge nas áreas do corpo mais expostas à radiação solar. Em estágios iniciais, o melanoma se desenvolve apenas na camada mais superficial da pele, o que facilita a remoção cirúrgica e a cura do tumor. Nos estágios mais avançados, a lesão é mais profunda e espessa, o que aumenta a chance de se espalhar para outros órgãos (metástase) e diminui as possibilidades de cura ([COSTA et al., 2019](#)).

Por isso, o diagnóstico precoce do melanoma é fundamental. Embora apresente pior prognóstico, avanços na medicina e o recente entendimento das mutações genéticas, que levam ao desenvolvimento dos melanomas, possibilitaram que pessoas com melanoma avançado hoje tenham aumento na sobrevida e na qualidade de vida ([GUERRA et al., 2019](#)).

Quando uma pessoa apresenta sinais de melanoma, o especialista analisa todo o histórico do paciente, observando os sintomas e fatores de risco que desencadeiam o melanoma. Para o diagnóstico do melanoma e determinar o grau de comprometimento da doença o profissional solicita alguns exames.

Um dos exames solicitados pelo especialista é o exame físico, o profissional observa a forma, tamanho, cor e textura das lesões e se há sangramento ou descamação. Em conjunto com o exame físico, alguns especialistas utilizam dermatoscopia para avaliar manchas de pele, possuindo dois tipos, a dermatoscopia manual que utiliza o dermatoscópio para verificar o risco da lesão e a dermatoscopia digital que permite a análise de uma fotografia ampliada da lesão. Porém, o diagnóstico definitivo do melanoma só pode ser feito através da biópsia, que é feito através de amostras de pele enviadas ao patologista para serem analisadas, sendo este o profissional responsável pelo diagnóstico definitivo da doença, como afirma [BRAZIL \(2009\)](#).

9.2.2 Processamento Digital de Imagens

É visto que em [Gonzalez e Woods \(2009\)](#) a área de processamento digital de imagens (PDI) trata da manipulação de imagens por computador de modo que geralmente a entrada e a saída do processo sejam imagens. O interesse nos métodos de processamento digital de imagens provém de duas áreas principais de aplicação: Melhora das informações visuais para a interpretação humana e processamento de dados de imagens para armazenamento, transmissão e representação, considerando a percepção automática por máquinas.

Segundo [Knob \(2011\)](#) o PDI é dividido em 6 etapas, como ilustra a Figura 2. Sendo elas: Aquisição das imagens, pré-processamento, segmentação, representação e descrição, reconhecimento e interpretação.

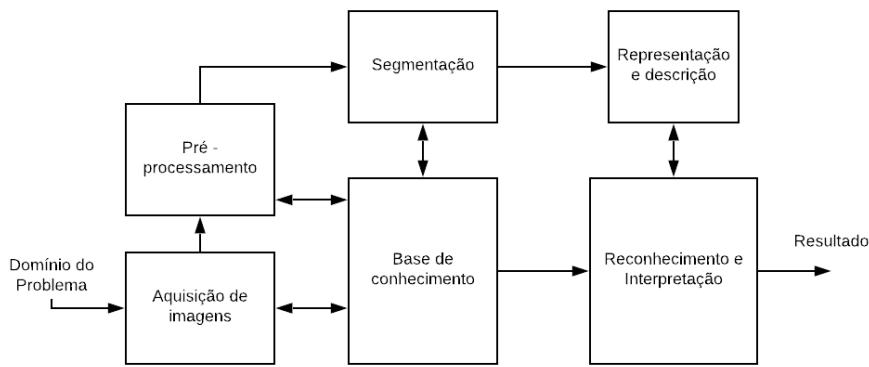


Figura 70 – Etapas do processamento digital de imagens [Knob \(2011\)](#).

O fim de cada etapa do PDI é a entrada para a próxima etapa, no final de todas as etapas, pode ser gerado uma imagem digital ou não.

- A primeira etapa é conhecida como aquisição de imagens, onde o digitalizador converte uma imagem analógica para digital.
- A segunda etapa é o pré-processamento, tendo como objetivo melhorar a imagem, aumentando também os resultados das etapas seguintes. Pode ser usado técnicas como: remoção de ruídos, suavizações de cinza, dentre outros.
- A terceira etapa conhecida como segmentação, tem como objetivo separar a imagem em uma região de interesse. Sendo abordado de forma manual, semi-automática e automática.
- A quarta etapa é a representação e descrição, conhecida como extração de características. Com o objetivo de descrever e representar as imagens através de valores. Cria-se um vetor de característica definindo um padrão, com base nos valores obtidos de cada medida extraída.
- A quinta etapa é conhecida como base de conhecimento, onde é aplicado todo o embasamento teórico e prático na área do PDI, com o objetivo de aplicar as etapas necessárias para a elaboração da metodologia do estudo.
- A última etapa é conhecida como reconhecimento e interpretação, sendo parte do processamento que classifica os objetos a partir de informações encontradas na imagem, atribuindo um rótulo para o objeto baseado na informação fornecida pelos descritores.

9.2.3 Índices de Diversidade

A diversidade é um termo muito utilizado na área da ecologia. O seu objetivo é informar variedade de espécies presentes em uma comunidade ou área ([OLIVEIRA et al., 2013](#)).

Um índice de diversidade é a medida da heterogeneidade de uma comunidade. Ele é utilizado na ecologia para medir a biodiversidade em um ecossistema. Na economia, serve para medir a distribuição sobre os setores de atividades econômicas em uma região. De forma mais geral, os índices de diversidade podem ser utilizados para estimar a diversidade de uma população na qual cada membro pertence a um único grupo ou espécie (SOUSA et al., 2011).

Segundo Pianka (1994), para medir a diversidade necessita de dois componentes: *a riqueza das espécies*, que consiste no número de espécies encontradas em determinada região e a *abundância relativa*, que é o número de indivíduos de uma determinada espécie existentes numa dada área. As medidas de espécies são geralmente úteis para comparar padrões em diferentes áreas.

A forma mais básica de utilizar índices de diversidade em imagens, é quando a comunidade representa uma imagem ou região da mesma, as espécies sendo os níveis de cinza e os indivíduos sendo os pixels OLIVEIRA et al. (2013). A Tabela 56 relaciona a metodologia proposta e a biologia.

Tabela 56 – Relação entre a metodologia e a biologia.

Metodologia Proposta	Biologia
Região de Interesse da Imagem	Comunidade
Níveis de Cinza da Imagem	Espécies
Píxels da Imagem	Indivíduos
Número de arestas entre duas Espécies	Distância Filogenética

9.2.3.1 Índice de Diversidade Filogenética

Diversidade filogenética é uma medida de uma comunidade que incorpora as relações filogenéticas das espécies (MAGURRAN, 2013). A premissa principal dessa medida é que a diversidade é maior em uma comunidade em que as espécies são filogeneticamente mais distintas, em outras palavras, pode-se afirmar que isso representaria uma imagem que possui muita variabilidade de níveis de cinza, ou seja, muitas espécies. A percepção dominante em ecologia evolutiva é que espécies coexistindo devem diferir significativamente e que a maior parte da variação entre espécies aparentadas é uma resposta adaptativa à competição no passado, quando os traços não diferiam (HARVEY; RAMBAUT, 2000). Esses índices consideram a quantidade de indivíduos, calculada a partir da arquitetura da árvore.

Uma árvore filogenética é uma representação gráfica, em forma de árvore, que apresenta relações evolutivas entre várias espécies ou outras entidades que possam ter um ancestral comum. Onde, as folhas representam os organismos e os nós internos representam os supostos ancestrais, as arestas da árvore denotam as relações evolutivas. A Figura 3 apresenta um exemplo de região de imagem onde acontece uma diversidade de espécies em relação aos indivíduos. Sendo possível o uso dos índices de diversidade filogenética.

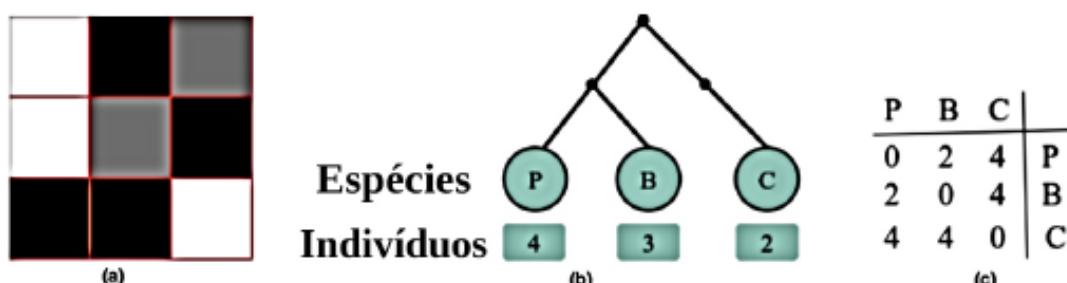


Figura 71 – (a) Representação da região analisada (b) Árvore filogenética extraída (c) Matriz com distâncias levando em consideração a quantidade de arestas da árvore filogenética que as separam as espécies analisadas. Disponível em Carvalho et al. (2018a).

Neste trabalho, são utilizados dois índices de diversidade filogenética para criação dos descritores de texturas, sendo eles: *Phylogenetic Diversity* (PD) e *Sum Of Phylogenetic Distances* (SPD).

O PD é o somatório dos comprimentos dos ramos da filogênica de cada espécie. Onde B é o número de ramificação da árvore, Li é o comprimento do ramo e Ai é a abundância média de espécies que ramo i . Representado pela Equação 9.1.

$$B * \frac{\sum_i^B LiAi}{\sum_i^B Ai}. \quad (9.1)$$

O SPD é a soma das distâncias filogenéticas entre cada par de espécies. Onde d_{mm} é a distância entre as espécies m e n ; a_m , abundância de espécies m ; S , número de espécies no conjunto focal¹. Como mostra a Equação 9.2.

$$SPD = (\frac{S(S - 1)}{2}) * \frac{\sum \sum m < n^a d^{mn} m^a n}{\sum \sum m < n^a m^a n}. \quad (9.2)$$

9.2.4 Reconhecimento de Padrões

Segundo Castro e Prado (2002a) o reconhecimento de padrões pode ser dividido em duas categorias: O estudo de seres humanos e organismos vivos com o objetivo de se estabelecer os modos pelos quais os mesmos desenvolvem e aprimoram suas capacidades de reconhecimento de padrões e o desenvolvimento de teorias e técnicas objetivando a construção de máquinas ou dispositivos capazes de apresentar características semelhantes às dos seres humanos em reconhecerem padrões.

A técnica de reconhecimento de padrões tem como objetivo classificar objetos tendo base um conhecimento prévio ou informações estatísticas extraídas dos padrões (CASTRO; PRADO, 2002a).

O uso do reconhecimento de padrões é recorrente para discriminar e classificar possíveis alterações em imagens médicas, determinando a presença ou ausência da doença. Após a extração de características, geralmente utiliza-se o reconhecimento de padrões para detectar as possíveis alterações nas imagens e classificar as possíveis anomalias ou não da imagem que desencadeiam a doença. Foi utilizado a ferramenta WEKA para realizar a classificação da pesquisa proposta.

Weka é uma coleção de algoritmos de aprendizado de máquina para tarefas de mineração de dados. Ele contém ferramentas para preparação de dados, classificação, regressão, agrupamento, mineração de regras de associação e visualização (WEKA, 2018).

Para o reconhecimento de padrões desta metodologia, baseado em estudos sobre classificadores levando consideração seu desempenho, facilidade de adaptação nos experimentos, eficácia e junção com as ferramentas de extração de características em imagens. Embasado nestes critérios, foram escolhidos os seguintes classificadores: *Random Forest*, *Sequential Minimal Optimization* (SMO), *Multi Layer Perceptron* (MLP) e *Instance Based Learning* (IBK).

O *Random Forest* é um classificador composto por uma coleção de árvores, onde são aleatoriamente independentes e identicamente distribuídas e cada árvore vota na classe mais popular, sendo a escolhida para o objeto (OSHIRO, 2013a).

O SMO é um dos principais métodos atuais para resolver o problema de programação quadrática que surge durante o treinamento de um SVM. Este método, em certa medida, pode diminuir o grau de dificuldade de um problema de programação quadrática através de estratégias de decomposição. Esses problemas são resolvidos analiticamente, permitindo que o SMO manipule um grande conjunto de treinamento e realize a distinção dos padrões entre os conjuntos de dados avaliados (ZENG et al., 2008).

O MLP é uma rede neural perceptron, formada por um conjunto de camadas. Tendo uma arquitetura de uma camada de entrada, que são várias unidades que recebem entradas no mundo real. Com uma ou duas camadas ocultas, elas podem aproximar virtualmente qualquer mapa de entrada-saída. E a camada de saída que retorna os resultados para o mundo real (PANCHAL et al., 2011).

O IBK é um classificador baseado no aprendizado de instâncias, conhecido como o algoritmo do vizinho mais próximo, que possui grandes requisitos de armazenamento. Descrevem-se como os requisitos de armazenamento que podem ser reduzidos significativamente com, no máximo, pequenos sacrifícios na taxa de aprendizado e na precisão da classificação (HALL, 2000).

¹ Comunidade analisada

9.2.5 Métricas de Validação

Para avaliar o desempenho dos classificadores, geralmente são usados métricas para validar a relevância da metodologia, onde são comumente utilizados as seguintes métricas: Acurácia (A), Sensibilidade (S) e Especificidade (E). Essas métricas fazem o uso da matriz de confusão, que faz a classificação correta e incorreta das classes, agrupando os resultados em quatro classes, respectivamente: Verdadeiro Positivo (VP), Verdadeiro Negativo (VN), Falso Positivo (FP) e Falso Negativo (FN).

Acúracia (A): É a proporção de previsões corretas, sem levar em consideração o que é positivo e o que é negativo. Esta medida é altamente suscetível a desbalanceamentos do conjunto de dados e pode facilmente induzir a uma conclusão errada sobre o desempenho do sistema. Representado pela Equação 9.3.

$$A = \frac{VP + VN}{VP + VN + FP + FN}. \quad (9.3)$$

Sensibilidade (S): É a proporção de verdadeiros positivos, isto é, a capacidade do sistema em predizer corretamente a condição para casos que realmente a têm. Representado pela Equação 9.4.

$$S = \frac{VP}{VP + FN}. \quad (9.4)$$

Especificidade (E): É a proporção de verdadeiros negativos, isto é, a capacidade do sistema em predizer corretamente a ausência da condição para casos que realmente não a têm. Representado pela Equação 9.5.

$$E = \frac{VN}{VN + FP}. \quad (9.5)$$

A análise ROC (*Receiver Operating Characteristic*) é uma ferramenta poderosa para medir e especificar problemas no desempenho do diagnóstico em medicina. Como visto em Braga (2001), a área sob curva (AUC) é caracterizado como uma boa forma de avaliar um sistema de classificação binária, onde o sistema de diagnóstico discrimina entre dois estados(especificidade e sensibilidade), verificando-se o melhor desempenho, alcançado pelo método avaliado, quando o valor da AUC for mais próximo de 1 (igual a 100%).

Para avaliar a concordância do experimento é usado o método estatístico de cálculo do Coeficiente Kappa (K). Sendo utilizado para avaliar a reprodutibilidade dos resultados na validação. Este resultado quanto mais próximo de 1, maior será a concordância do experimento. A Tabela 57 representa os níveis de concordância do K.

Tabela 57 – Tabela de Concordância do Coeficiente Kappa.

Valor do Coeficiente Kappa	Nível de Concordância
<0	Não existe concordância
0 - 0.20	Concordância mínima
0.21 - 0.40	Concordância razoável
0.41 - 0.60	Concordância moderada
0.61 - 0.80	Concordância substancial
0.81 - 1.0	Concordância perfeita

No cálculo do K, P_o é a taxa de aceitação relativa e P_e é a taxa hipotética da aceitação. Representado pela Equação 9.6.

$$K = \frac{P_o - P_e}{1 + P_e} \quad (9.6)$$

As métricas citadas acima, tem como finalidade estimar o desempenho da metodologia realizada neste trabalho, para determinar se os resultados obtidos foram relevantes ou não e também apontar os pontos que foram positivos e negativos para que seja aplicado melhorias em trabalhos futuros para essa pesquisa.

9.3 Trabalhos Relacionados

A literatura especializada oferece estudos relacionados a classificação do melanoma, assim como técnicas de extração de características. Nesta seção será abordado um resumo de alguns desses trabalhos.

No trabalho de [Alencar \(2015\)](#) é desenvolvida uma metodologia para categorizar as imagens de pele em melanoma ou não, aplicando a limiarização de *otsu* para segmentar as imagens no canal azul do esquema RGB e extrair os contornos utilizando as funções *FindContours*, *ConvexHull* e *ConvexityDefects* da biblioteca OpenCv. Foi utilizado o classificador *Multi Layer Perceptron* para o reconhecimento de padrões, resultando em 86,6% de acurácia.

Na metodologia de [Rastgoo et al. \(2016\)](#) é proposto o uso de uma técnica que utiliza o *Random Forest* para reconhecer os padrões, juntamente com técnicas de extração de características que são, *Scale-Invariant Feature Transform* (SIFT), Histogramas de Ângulos Opostos e Intensidade de RGB. O treinamento ocorreu através de validação cruzada com 10 divisões com a base de imagens ph2. Obtendo 100% de especificidade e 90,3% de sensibilidade.

No trabalho de [Moura et al. \(2017\)](#) apresenta uma metodologia para detecção automática do melanoma, através de um descritor híbrido para a extração de características, sendo eles, a regra ABCD, *Gray Level Run Length Matrix* (GLRLM), *Histograms of Oriented Gradients* (HOG), *Local Binary Pattern* (LBP), Tamura, *Box-Counting*. Onde foi utilizado o algoritmo Razão de Ganho de Informação (*Gain Ratio Information*) para criar o vetor de características e em seguida, reconhecimento de padrões em melanoma e não melanoma obtendo 95% de acurácia com o classificador MLP.

Na metodologia de [Li e Shen \(2018\)](#) é proposto uma abordagem para segmentar e classificar as imagens de lesões de pele de forma simultânea através da técnica *Lesion Indexing Network* (LIN). Através de 2 CNNs profundas, FCRNs-88, treinadas com diferentes conjuntos de treinamento, para produzir o resultado da segmentação e classificação. Usa-se a técnica *Lesion Indexing Calculation Unit* (LICU) para medir a importância do pixel na decisão da classificação. Resultando em 85,7% de acurácia, 96,6% em especificidade e 49,0% na sensibilidade.

No trabalho de [Bonafini e Perlin \(2018\)](#) mostra uma metodologia para o diagnóstico automático do melanoma, através das regras ABCDE (Assimetria, borda, cor, diâmetro e evolução) são extraídas as características e feito o reconhecimento de padrões com o classificador *Support Vector Machine* (SVM). Resultando em 84,5% de acurácia.

[Maia et al. \(2019\)](#) propõe uma metodologia para o diagnóstico do melanoma através das regras ABCD, com o uso da rede neural VGG16, *Inception* e *Resnet* para a extração de características. Realizando o reconhecimento de padrões com o classificador Random Forest e obtendo uma acurácia de 92%, sensibilidade em 92,5% e especificidade de 91,8%.

A Tabela 4 apresenta o desempenho dos trabalhos relacionados, mencionando a metodologia aplicada, base de imagem e os resultados obtidos (acurácia, sensibilidade e especificidade).

Tabela 58 – Trabalhos Relacionados.

Trabalho	Metodologia	Base	Acurácia (%)	Especificidade (%)	Sensibilidade (%)
(ALENCAR, 2015)	Contornos + MLP	PH2	86,6	-	-
(RASTGOO et al., 2016)	SIFT + Random Forests	PH2	-	100	90,3
(MOURA et al., 2017)	Descritor Híbrido + MLP	PH2	95,5	95,4	95,5
(LI; SHEN, 2018)	LIN CNN	ISIC	85,7	96,6	49
(BONAFINI; PERLIN, 2018)	Regras ABCDE + SVM	ISIC	84,5	-	-
(MAIA et al., 2019)	Inception + Resnet + Random Forest	PH2	92	92,5	91,8

Os trabalhos relacionados acima apresentaram resultados promissores na detecção e diagnóstico do melanoma e lesões de pele, porém alguns apresentaram dificuldades na execução do método. Como nos trabalhos de [Moura et al. \(2017\)](#), [Rastgoo et al. \(2016\)](#) e [Maia et al. \(2019\)](#) que utilizaram uma base de imagens pequena composta por 200 imagens dermatológicas. Na metodologia de [Alencar \(2015\)](#) e [Bonafini e Perlin \(2018\)](#) poderia ter sido aplicado outras métricas para validar suas pesquisas. No projeto de [Li e Shen \(2018\)](#) poderia ser empregado técnicas para melhorar a sensibilidade.

A maioria dos trabalhos citados acima, utilizam descritores baseados em forma, seguindo as regras ABCD e descritores baseados em redes neurais, com isso, nesta pesquisa pretende-se utilizar uma nova abordagem com descritores de textura baseado em índices de diversidade filogenética para verificar a relevância da análise de textura em comparação aos descritores de forma e aos baseados em redes neurais, também aumentar a acurácia e melhorar os valores da sensibilidade e especificidade.

9.4 Método Proposto

A metodologia proposta neste trabalho teve como finalidade realizar a classificação das regions of interest (ROIs) em melanoma ou não. Compreendido em 4 etapas, na primeira, a aquisição do banco de imagens *International Skin Imaging Collaboration* (ISIC) ([ISIC, 2016](#)); na segunda, é realizado a extração de características e de textura, implementados neste trabalho com a biblioteca de manipulação de imagens OpenCv ([PURI; JAIN, 2019](#)); na terceira, utiliza-se algoritmos disponíveis na ferramenta WEKA ([WEKA, 2018](#)) para realizar a classificação das características extraídas; e por fim, é feita a validação dos resultados com as métricas apresentadas na Seção 2.5. A Figura 4 mostra o resumo de cada etapa desta metodologia.

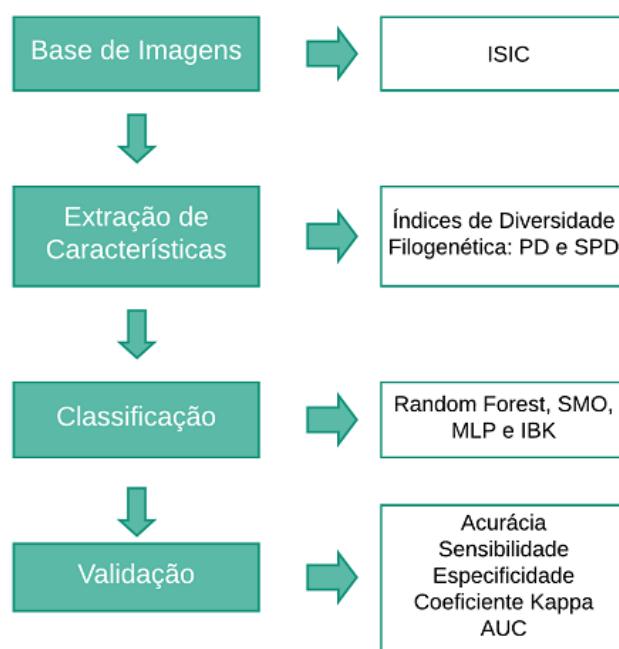


Figura 72 – Metodologia Proposta.

9.4.1 Aquisição das Imagens

A ISIC é um esforço internacional para melhorar o diagnóstico do melanoma, patrocinado pela Sociedade Internacional de Imagem Digital da Pele (ISDIS). A base de imagens ISIC contém a maior coleção publicamente disponível de imagens dermatoscópicas ([ISIC, 2016](#)).

A base de imagens é composta por 900 imagens demartológicas com as marcações feitas por especialistas. A partir dessas marcações são extraídas as regiões de interesse (ROIs), gerando as imagens com as delimitações feitas pelos especialistas e as originais. Com 727 imagens não melanoma e 123 imagens melanoma.

9.4.2 Extração de Características

Após a aquisição das imagens e a extração das ROIs a partir das marcações feitas pelos especialistas, as imagens com as delimitações feitas pelos especialistas foram encaminhadas para a fase de extração de

características de textura. Sendo aplicado a metodologia de extração de características de textura em imagens demartológicas baseado em índices de diversidade filogenética.

Neste trabalho, são utilizados dois índices de diversidade filogenética para criação dos descritores de textura, sendo eles: A diversidade filogenética e a soma das distâncias filogenéticas. Onde, no final do processo de extração de características, esses índices descrevem as texturas das imagens e os atributos fornecidos buscam reconhecer as medidas de homogeneidade ou heterogeneidade da ROI. Servindo como base para categorizar em melanoma e não melanoma.

9.4.3 Classificação e Validação dos Resultados

Para a classificação, foi utilizado a ferramenta Weka, dentre os algoritmos presentes no Weka, foram selecionados: Radom Forest, SMO, MLP e IBK, por apresentarem bons resultados e pelo fato de serem rigorosos na presença de ruídos no processo de classificação, também utilizam descritores que se baseiam em comportamento dentro de comunidades, sendo uma boa escolha para a classificação da metodologia.

Utilizando os parâmetros com os valores padrão juntamente com a validação cruzada (*Cross Validation*) de k -folds, atribuindo o $k = 10$. Esse método tem como procedimento dividir os indivíduos em 10 grupos, separando o treino em 9 grupos e 1 grupo para teste, ocorrendo 10 cruzamentos alterando o grupo de teste, após realizar todos os cruzamentos é gerado uma média que atribue-se como resultado.

Por fim, após a etapa do reconhecimento de padrões é necessário a utilização de métricas para validar os resultados e analisar possíveis melhorias do mesmo. A metologia proposta, utiliza métricas de validação recorrentes na área de processamento digital de imagens, sendo baseadas em estatísticas, respectivamente: Acurácia (A), Sensibilidade (S), Especificidade (E), Coeficiente Kappa (K) e Área Sobre a Curva (ROC).

9.5 Resultados e Discussões

Esta Seção mostra e discute os resultados obtidos da aplicação da metodologia proposta neste trabalho, obtendo o resultado da classificação das ROIs das imagens de pele em melanoma e não melanoma. Foram feitos testes dos descritores de características para imagens em níveis de cinza, do canal vermelho, do canal verde e do canal azul, com o intuito de verificar o potencial dos descritores de textura em cada um desses canais.

As Tabelas 59, 60, 61 e 62 são uma síntese dos principais testes feitos nesta metodologia, através de descritores de textura para a extração de características e utilizando os classificadores: IBK, MLP, Random Forest e SMO para o reconhecimento de padrões das imagens de pele.

A Tabela 59 demonstra os resultados dos testes dos descritores de textura extraídos para as imagens em níveis de cinza.

Tabela 59 – Testes em níveis de Cinza.

Classificadores	Acurácia (%)	Especificidade (%)	Sensibilidade (%)	ROC	Kappa
IBK	98,6	99,2	99,2	0,979	0,957
MLP	92,6	99,2	92,3	0,805	0,732
Random Forest	97,6	98,6	98,5	0,996	0,924
SMO	82,1	99,9	81,9	0,537	0,113

Dante dos resultados apresentados na Tabela 59, o classificador SMO apresentou a menor taxa de predição, obtendo uma acurácia de 82%, especificidade em 99%, sensibilidade de 81,9%, ROC em 0,537 e K de 0,113. O resultado mais relevante foi obtido pelo classificador IBK, com porcentagem de 98% de acurácia, 99% em especificidade, 99% de sensibilidade, ROC em 0,979 e K de 0,957.

A Tabela 60 representa os resultados dos testes dos descritores de textura extraídos do canal vermelho.

Tabela 60 – Testes com o canal vermelho (Red).

Classificadores	Acurácia (%)	Especificidade (%)	Sensibilidade (%)	ROC	Kappa
IBK	99,8	99,9	100	0,999	0,996
MLP	94,7	98,5	95,2	0,941	0,822
Random Forest	99,4	99,7	99,6	0,998	0,982
SMO	81,5	99,9	81,5	0,522	0,070

Entre os resultados obtidos da classificação mostrada na Tabela 60, o que obteve o pior resultado foi o classificador SMO, com 81% na acurácia, 99% em especificidade, 81% de sensibilidade, 0,522 para ROC e 0,070 para o K. A melhor predição foi obtida pelo classificador IBK, alcançando 99% na acurácia, 99% para especificidade, 100% em sensibilidade, 0,999 na ROC e 0,996 no K.

A Tabela 61 apresenta os resultados dos testes dos descritores de textura extraídos do canal verde.

Tabela 61 – Testes com o canal verde (Green).

Classificadores	Acurácia (%)	Especificidade (%)	Sensibilidade (%)	ROC	Kappa
IBK	99,6	99,7	99,9	0,996	0,989
MLP	94,8	97,4	96,3	0,968	0,832
Random Forest	98,7	99,3	99,2	0,996	0,960
SMO	80,6	99,9	80,8	0,499	-0,002

Diante dos resultados atingidos na Tabela 61, o resultado apresentado de menor relevância foi do classificador SMO, com 80% na acurácia, 99% de especificidade, 80% em sensibilidade, 0,499 para ROC e -0,002 para o K. O resultado mais promissor entre os classificadores foi o IBK, obtendo 99% de acurácia, 99% em especificidade, 99% na sensibilidade, 0,996 para ROC e 0,989 no K.

A Tabela 62 mostra os resultados dos testes dos descritores de textura extraídos do canal azul.

Tabela 62 – Testes com o canal azul (Blue).

Classificadores	Acurácia (%)	Especificidade (%)	Sensibilidade (%)	ROC	Kappa
IBK	99,3	99,4	99,7	0,991	0,978
MLP	96,1	96,1	99,6	0,975	0,894
Random Forest	99,2	99,4	99,6	0,996	0,975
SMO	80,6	99,9	80,8	0,499	-0,002

Entre os resultados mostrados na Tabela 62, o classificador com a menor taxa preditiva foi o SMO, tendo 80% na acurácia, 99% de especificidade, 80% em sensibilidade, 0,499 na ROC e -0,002 no K. O de maior taxa preditiva em termos gerais foi o classificador IBK, porém, o classificador Random Forest obteve valores relevantes nas métricas de validação. O IBK com uma ROC de 0,991, K de 0,978 e o Random Forest uma ROC com 0,996, K de 0,975. O IBK com uma maior acurácia e sensibilidade, e a especificidade igual ao Random Forest.

Analisando os resultados apresentados nas Tabelas 59 a 62, com base em todos os experimentos, a Tabela 60 apresentou os melhores resultados, onde, a média de especificidade foi acima de 98%. Também obtendo o melhor resultado dentre todos os experimentos, com 99% de acurácia, 0,999 curva ROC e coeficiente kappa 0,996, com o uso do classificador IBK e os índices de diversidade filogenética, aplicado no canal vermelho (RED) atribuindo os melhores resultados em relação aos testes para as imagens em níveis de cinza e nos canais verde e azul.

Entre todos os classificadores, o IBK mostrou os resultados mais relevantes por ser baseado em aprendizado de instâncias capaz de ter um maior número de acertos com descritores filogenéticos, com taxas de previsões médias de 98% de acurácia em todos os canais, seguido pelo Random Forest que apresentou uma taxa média de 97% de acurácia em todos os canais, após esses, o MLP que apresentou uma taxa média de

92% de acurácia em todos os canais. Por fim, o que apresentou as piores predições o SMO, que apresentou uma taxa média de 80% de acurácia.

Com intuito de averiguar o potencial dos descritores de textura propostos neste trabalho, foram feitos testes com descritores comumente utilizados na literatura. O descritor escolhido foi *Gray-Level-Co-Occurrence Matrix* (GLCM). O GLCM é um método estatístico para descrição de texturas, capaz de armazenar a probabilidade de que dois valores de intensidade cinza estejam envolvidos por uma relação espacial. A partir da matriz de probabilidades, foram usadas as seguintes medidas para extrair as características: Contraste, dissimilaridade, homogeneidade, correlação, energia e o segundo momento angular (ASM). Os testes foram iguais ao da metodologia proposta, feitos com a mesma base de imagens. A Tabela 63 mostra os resultados dos testes do descritor GLCM.

Tabela 63 – Teste com o descritor GLCM.

Classificadores	Acurácia (%)	Especificidade (%)	Sensibilidade (%)	ROC	Kappa
IBK	67,7	80,1	16,1	0,481	-0,037
MLP	81,4	81,3	4,0	0,695	0,061
Random Forest	77,0	80,9	21,0	0,590	0,014
SMO	80,7	100	-	0,500	0

Dante dos resultados apresentados na Tabela 63, o classificador MLP mostrou a melhor taxa preditiva com valores de 81,4 de acurácia, 81,3 para especificidade, 4,0 em sensibilidade, 0,695 na ROC e 0,061 no Kappa. O pior caso mostrou-se no classificador IBK, com taxa de 67,7 na acurácia, 80,1 em especificidade, 16,1 de sensibilidade, 0,481 na ROC e -0,037 no Kappa. Analisando os resultados obtidos com o descritor GLCM e da metodologia proposta, feitos os testes com a mesma base de imagens, nota-se que a pesquisa proposta apresentou uma maior taxa preditiva nas métricas de validação, que determina o nível de relevância do estudo. O melhor resultado do GLCM apresentou 81,4 de acurácia, 81,3 para especificidade, 4,0 em sensibilidade, 0,695 na ROC e 0,061 no Kappa e a metodologia proposta apresentou 99% na acurácia, 99% para especificidade, 100% em sensibilidade, 0,999 na ROC e 0,996 no K.

A Tabela 64 apresenta uma comparação da metodologia deste trabalho com os trabalhos relacionados citados na Seção 3.

Tabela 64 – Comparativo com os Trabalhos Relacionados.

Trabalho	Metodologia	Base	Acurácia (%)	Especificidade (%)	Sensibilidade (%)
(ALENCAR, 2015)	Contornos + MLP	PH2	86,6	-	-
(RASTGOO et al., 2016)	SIFIT + Random Forests	PH2	-	100	90,3
(MOURA et al., 2017)	Descritor Híbrido + MLP	PH2	95,5	95,4	95,5
(LI; SHEN, 2018)	LIN CNN	ISIC	85,7	96,6	49
(BONAFINI; PERLIN, 2018)	Regras ABCDE + SVM	ISIC	84,5	-	-
(MAIA et al., 2019)	Inception + Resnet + Random Forest	PH2	92	92,5	91,8
Metodologia	Descritores Filogenéticos + IBK	ISIC	99,8	99,9	100

A Tabela 64 faz um comparativo entre a metodologia proposta e os trabalhos relacionados, avaliando a acurácia, especificidade e sensibilidade. Como são bases de imagens diferentes deste trabalho, este comparativo visa apenas ilustrar as predições obtidas em cada metodologia. Onde, os resultados da metodologia proposta mostram-se promissores em comparação aos demais trabalhos. Na metodologia de Bonafini e Perlin (2018) foi obtido uma acurácia de 84,5%, sendo a menor taxa entre os trabalhos relacionados. O trabalho de Moura et al. (2017) obteve as melhores valorações preditivas entre os trabalhos relacionados, tendo abaixo só a especificidade em relação ao trabalho de Rastgoo et al. (2016) com o valor de 100. Por fim, baseado nos valores das predições, a metodologia proposta apresentou resultados satisfatórios diante da bibliografia.

9.6 Conclusão

Com o aumento da incidência de doenças, os especialistas necessitam cada vez mais de mecanismos para uma segunda opinião em seus diagnósticos de doenças. O uso de sistemas computadorizados tornou-se recorrente no auxílio do diagnóstico das doenças, por serem precisos em seus resultados.

Este trabalho apresentou uma metodologia para classificação em melanoma ou não, com o uso de índices de diversidade filogenética em imagens, acoplado com o reconhecimento de padrões, capazes de discriminar e classificar, em imagens, as anomalias nas regiões suspeitas que desencadeiam o melanoma, atribuindo uma segunda opinião para os especialistas.

Os índices de diversidade filogenética mostraram-se eficientes na categorização de regiões em melanoma e não melanoma. Foram obtidos resultados promissores nas métricas de validação, com taxas de acertos de 99%.

Como análise nos resultados obtidos, os descritores baseados em textura mostraram resultados promissores na classificação do melanoma, sendo possível proporcionar aos especialistas uma segunda opinião no diagnóstico do melanoma.

Pretende-se, como trabalhos futuros:

- Aumentar as amostras de melanoma e não melanoma utilizando mais bases de imagens;
- Aplicar um método de segmentação nas imagens de pele;
- Aplicar os índices filogenéticos em outras lesões de pele.

Classificação de imagens em COVID-19 e Não-COVID-19 usando CNN para extração de características e múltiplos classificadores

Autores: Antônio Oseas de Carvalho Filho; Edelson Damasceno Carvalho.

Resumo: A doença de coronavírus 2019 (COVID-19) é uma doença respiratória causada pela infecção com o coronavírus da síndrome respiratória aguda grave 2 (SARS-CoV-2). Até Janeiro de 2021, a COVID-19 já infectou mais de 102,8 milhões de pessoas em todo o mundo e é responsável por mais de 2,2 milhões de mortes. Um grande problema enfrentado no diagnóstico da COVID-19 é a ineficiência e a escassez de exames médicos. O uso da tomografia computadorizada (TC) mostrou promissora para a avaliação de pacientes com suspeita de infecção por COVID-19. A análise de TC é complexa e exige esforço do especialista, sendo um processo repetitivo, demorado e cansativo, devido a uma grande quantidade de imagens para serem analisadas, causando fadiga, podendo levar a erros de diagnósticos. O uso de sistemas de apoio ao diagnóstico auxiliado por computador pode minimizar os problemas gerados pela análise de TC pelos especialistas. Recentemente, as redes neurais convolucionais (CNN do inglês *Convolutional Neural Network*) foram empregados no desenvolvimento de sistemas para auxílio ao especialista no diagnóstico médico. Este trabalho apresenta uma metodologia para diagnosticar a COVID-19 usando um extrator de características treináveis usando uma CNN baseada na LeNet-5 e múltiplos classificadores. Primeiro, a qualidade das imagens foi melhorada usando equalização de histograma e CLAHE. Em seguida, a CNN baseada na LeNet-5 é usada para extrair características de 708 TCs, sendo, 312 com COVID-19 e 396 Não-COVID-19. Após a extração das características, foram utilizados múltiplos classificadores para classificação em COVID-19 e Não-COVID-19. A metodologia proposta utilizando pré-processamento com CLAHE, extração de características usando a CNN e o XGboost obteve os melhores resultados, apresentando uma acurácia de 98,5%, recall de 98,5%, precisão de 98,6%, F-score de 0,985, AUC de 0,985 e índice kappa de 0,971. Os resultados obtidos mostram que a metodologia proposta pode ser utilizada como um sistema para auxiliar no diagnóstico da COVID-19.

Keywords: COVID-19. tomografia computadorizada. pré-processamento. características profundas. diagnóstico.

10.1 Introdução

A doença de coronavírus 2019 (COVID-19) é uma doença respiratória causada pela infecção com o coronavírus da síndrome respiratória aguda grave 2 (SARS-CoV-2) (GORBALENYA et al., 2020). Até Janeiro de 2021, a COVID-19 já infectou mais de 102,8 milhões de pessoas em todo o mundo e é responsável por mais de 2,2 milhões de mortes (WHO, 2020). Um grande problema enfrentado no diagnóstico da COVID-19 é a ineficiência e a escassez de exames médicos. Neste sentido, vários esforços foram dedicados para buscar métodos alternativos de diagnóstico da COVID-19.

A tomografia computadorizada (TC) é considerada promissora para a avaliação de pacientes com suspeita de infecção por COVID-19 (BERNHEIM et al., 2020a). A TC manifesta achados radiológicos claros de pacientes com COVID-19, servindo como uma maneira de teste mais eficiente e acessível (HE et al., 2020a). O principal problema desse método é que ele depende do especialista para analisar as imagens de TCs, pois o processo é repetitivo, demorado e cansativo para o especialista, devido a uma grande quantidade de imagens para serem analisadas, causando fadiga, podendo levar a erros de diagnóstico (de Oliveira Torres et al., 2020; LIU et al., 2018; de Carvalho Filho et al., 2018).

Para minimizar os problemas gerados pela análise de imagens pelos especialistas, sistemas de apoio ao diagnóstico (*Computer-Aided Diagnosis - CAD*) surgem como uma alternativa de auxílio ao diagnóstico médico. Sistemas CAD utilizam de técnicas de processamento e análise de imagens, e do poder computacional para analisar as imagens, sendo cruciais para casos em que o diagnóstico é muito difícil para o olho humano (CARVALHO et al., 2018b; de Carvalho Junior et al., 2018).

Recentemente o uso de métodos de aprendizado profundo vem sendo implementado no desenvolvimento de sistemas CAD. As redes neurais convolucionais (*Convolutional Neural Network - CNN*) que são técnicas de aprendizado profundo, tem a capacidade de interpretar automaticamente imagens e prever se são positivas para COVID-19 (GOZES et al., 2020). No entanto, a complexidade do modelo, dificuldade de treinamento, alto custo computacional e necessidade de conjunto grande de dados, dificulta o desenvolvimento de uma metodologia usando CNN com uma aplicação eficaz.

Para o treinamento e a classificação eficiente de um modelo de CNN é necessário um grande conjunto de dados de TCs. Como o conjunto de dados possui apenas 708 imagens de TCs, o treinamento de um modelo de CNN é complexo, sendo assim foi utilizado a CNN como extrator de características das imagens de TCs. Após os dados extraídos, a classificação é feita usando múltiplos classificadores.

O uso de CNN como extrator de características de imagens é bastante utilizada para aplicações de imagens médicas (Zhu; Zhang; Xue, 2015; van Ginneken et al., 2015), reconhecimento de rosto (MORI; MATSUGU; SUZUKI, 2005), reconhecimento de dígitos manuscritos (NIU; SUEN, 2012) e detecção de objetos (OUYANG et al., 2014; GIRSHICK et al., 2014). Os trabalhos mostram que as características da CNN são genéricas e podem ser usados para tarefas de classificação fora do domínio exato para o qual as redes foram treinadas.

10.2 Objetivos Gerais e Específicos

O objetivo geral deste trabalho consiste em desenvolver um método de classificação de imagens médicas de tomografia computadorizada para diagnóstico automático da COVID-19 utilizando CNN para extração de características e múltiplos classificadores, que seja robusta a presença de ruídos, a variações na aquisição das imagens e que necessite da mínima intervenção do usuário ou configuração de parâmetros.

Tendo em vista o objetivo geral deste trabalho, pretende-se:

- Realizar um levantamento dos principais artigos da área de processamento de imagens de TC para classificação da COVID-19.
- Levantamento, descrição e elaboração da base de imagens de TC, que foi utilizada no trabalho.

- Levantamento sobre as técnicas de extração de características usando CNN.
- Levantamento sobre os classificadores comumente empregados na classificação de imagens médicas.
- Estudo, desenvolvimento e implementação de um algoritmo para classificação automática de TCs em COVID-19 e não COVID-19.
- Teste, avaliação e comparação do desempenho do método proposto com outros existentes.

10.2.1 Justificativa

Os sistemas CAD utilizam-se de técnicas de processamento, análise e reconhecimento de padrões de imagens, com o objetivo principal de detectar e diagnosticar anomalias em exames médicos, aproveitando da velocidade e precisão dos sistemas computacionais para fornecer uma segunda opinião ao especialista. Com os avanços nas técnicas de visão computacional e o aumento da velocidade dos computadores, os sistemas CAD para auxílio ao diagnóstico médico vêm se tornando relevantes para um diagnóstico precoce e mais preciso.

O uso de CNN como extrator de características, possui as seguintes vantagens: (a) capacidade de extrair características relevantes através de aprendizado de transformações; (b) depende de menor número de parâmetros de ajustes, dado que, cada unidade de uma camada não é conectada com todas as unidades da camada seguinte, assim, há menos pesos para serem atualizados, facilitando o treinamento; (c) capacidade de obter informações relevantes sem a necessidade da segmentação de estruturas do objeto.

Assim, os dados extraídos com a CNN podem ser utilizado por classificadores comumente empregados na literatura sem a necessidade de treinamentos longos, tornando o processo mais rápido. A classificação de imagens de TC em COVID-19 e Não-COVID-19 usando CNN para extração de características e múltiplos classificadores vai possibilitar ao especialista uma segunda opinião, levando a um diagnóstico mais preciso e de baixo custo.

10.3 Referencial Teórico

Nessa seção é apresentado a fundamentação teórica necessária para a compreensão desse trabalho, abordando os temas da COVID-19, tomografia computadorizada, processamento digital de imagens, extração de características e classificação de imagens.

10.3.1 COVID-19

A doença de coronavírus 2019 (COVID-19) é uma doença respiratória causada pela infecção com o coronavírus da síndrome respiratória aguda grave 2 (SARS-CoV-2) ([GORBALENYA et al., 2020](#)). Os pacientes mais suscetíveis às complicações do novo coronavírus, são idosos acima de 60 anos, grávidas e pessoas que possuem doenças crônicas como diabetes, asma e hipertensão ([WHO, 2020](#)). A COVID-19 teve seus primeiros registros na cidade de Wuhan na China em dezembro de 2019. Em 11 de março de 2020 a COVID-19 foi caracterizada pela Organização Mundial da Saúde (OMS) como uma pandemia mundial.

Para auxiliar no diagnóstico, existem alguns exames que são realizados em laboratórios de confiança, hospitais ou na rede pública de saúde. Atualmente, os exames mais comuns para a detecção da COVID-19 são o proteína c-reativa (PCR) e a Sorologia. O primeiro é o exame que confirma a presença do vírus usando um pedaço do código genético para identificá-lo. O segundo mostra que houve contato com o vírus, pois mede a resposta imune. Entre os exames de imagem, a tomografia computadorizada de tórax é o método de escolha para avaliar comprometimento pulmonar por ter alta sensibilidade e especificidade. Utilizado em pacientes internados, o exame pode indicar que pessoas com forte suspeita epidemiológica e com padrão tomográfico típico podem ser consideradas suspeitas para COVID-19. Um grande problema enfrentado no diagnóstico da COVID-19 é a ineficiência e a escassez de exames médicos. Até Janeiro de 2021 a COVID-19 já infectou mais de 102,8 milhões de pessoas em todo o mundo e é responsável por mais de 2,2 milhões de mortes ([WHO, 2020](#)).

10.3.2 Tomografia computadorizada

Tomografia computadorizada é um exame não invasivo que combina equipamentos especiais de raios-x com computadores programados para produzir imagens dos órgãos internos. A TC é obtida por um feixe de raio-x emitido por uma pequena fenda e tem a forma de leque, onde um tubo de raios-x gira 360 graus em torno da região do corpo a ser estudada e a imagem obtida são “fatias” da região investigada ([VOSPER; GRAHAM; CLOKE, 2011](#)). Em oposição ao feixe de raios-x emitidos, tem-se um detector de fôtons que gira concomitantemente ao feixe. Os fôtons emitidos dependem da espessura do objeto e da capacidade deste de absorver os raios-x. Os detectores de fôtons da TC transformam os fôtons emitidos em sinal analógico e depois digital.

Após a aquisição das seções transversais, as informações obtidas são processadas utilizando técnicas matemáticas. As imagens tomográficas podem ser obtidas em dois planos: axial (perpendicular ao maior eixo do corpo) e o coronal (visão frontal). Após a sua obtenção, recursos computacionais podem permitir reconstruções no plano sagital (paralelo à sutura sagital do crânio) ou reconstruções tridimensionais. A principal vantagem da TC é que permite o estudo de seções transversais do corpo vivo, permitindo a detecção ou o estudo de anomalias que não seria possível senão através de métodos invasivos.

10.3.3 Processamento de imagens digitais

Com a evolução da tecnologia ocorrida nos últimos anos, além do desenvolvimento de novos algoritmos para lidar com sinais bidimensionais, permite cada vez mais aplicações na área de processamento digital de imagens, com isso, desperta o interesse para estudos na área.

Uma imagem pode ser definida como uma função bidimensional, onde x e y são coordenadas espaciais e a amplitude de f no par de coordenadas $f(x, y)$ é denominado intensidade ou nível de cinza. Quando x e y e a intensidade dos valores de f são finitos e discretos, têm-se então uma imagem digital. Esses elementos possuem uma localização e um valor particular, chamados de *pixels* ([SOUZA, 2011](#)).

Processamento digital de imagens é um conjunto de atividades interconectadas, iniciando com a captura da imagem por um sistema de aquisição, em seguida deve ser representada de forma adequada para tratamento computacional, vários processos são necessários para que possa ser analisado e identificado os objetivos de interesse, para isso é necessário que seja extraído características como textura, forma e vizinhanças.

10.3.3.1 Extração de características

A extração de características é o processo mais importante no desenvolvimento de um sistema de classificação de imagens automático ([REN et al., 2017](#)). O desempenho da classificação pode ser influenciado pela qualidade dos dados extraídos, levando a uma perda de desempenho pelo sistema. Nos últimos anos, modelos de *Deep Learning* foram propostos para etapa de extração de características em imagens. A CNN é um modelo de *Deep Learning* que possui uma estrutura hierárquica de aprendizado de características com alta qualidade em suas camadas.

10.3.3.2 CNN

A CNN tem a capacidade de reduzir a complexidade da rede e os números de parâmetros através de campos receptivos locais, operação de compartilhamento e compartilhamento de peso. O ajuste dos *kernels* de convolução são feitos pelo algoritmo de retropropagação ([ZIPSER; ANDERSEN, 1988](#)), que é baseado no algoritmo estocástico de descida de gradiente, utilizado para reduzir o espaço entre os dados de saída da rede e os rótulos de treinamento. Uma CNN consiste em camadas alternadas de convolução e subamostragem, transformando depois em camadas totalmente conectadas ao se aproximar da camada de saída.

As CNNs conseguem extrair características geralmente úteis de dados, detectar e remover redundâncias de entrada e preservar apenas aspectos essenciais dos dados em representações robustas e discriminatórias ([MASCI et al., 2011](#)). Suas camadas semi-conectadas e as totalmente conectadas fornecem um ambiente razoável para o avanço do processo de treinamento e aprendizado ([LIU, 2018](#)). Assim, as camadas de

convolução servem como extrator de características eficiente, especializada na redução da dimensão dos dados e na produção de um conjunto de dados menos redundante.

10.3.3.3 Métodos de classificação

O processo de classificação pode ser realizado considerando as classes previamente definidas. Nas classes previamente definidas, cada parâmetro que caracteriza cada classe é obtido através de treinamento, executado antes da aplicação do algoritmo de classificação. A classificação consiste em reconhecer a qual de um conjunto de categorias pertence uma nova observação, baseado em um treinamento prévio num conjunto de dados que tenha observações cuja categoria é conhecida (RIPLEY, 1996). Para o desenvolvimento deste trabalho, a classificação foi realizada usando os classificadores *XGBoost*, *random forest*, *multilayer perceptron* e *support vector machine*.

10.3.3.4 XGBoost

No aprendizado de máquina, o aumento de árvores é um método altamente eficaz e amplamente utilizado. O *Extreme Gradient Boosting* (*XGBoost*) é um sistema de aprendizado de máquina escalável e eficaz para aumento de árvores, proposto por Chen e Guestrin (2016). O aumento de árvore é um algoritmo de aprendizado que faz com que classificadores fracos se tornem fortes na classificação de um conjunto de dados. O *XGBoost* é um método de árvore que aplica o princípio de impulsionar aprendizado fraco usando a arquitetura de gradiente descendente. No entanto, o *XGBoost* melhora a estrutura básica do *Gradient Boosting Machines* por meio da otimização de sistemas e aprimoramentos algorítmicos (CHEN; GUESTRIN, 2016). O *XGBoost* têm a capacidade de classificar problemas usando uma quantidade mínima de características. Os parâmetros usados no *XGBoost* foram os seguintes: max depth = 7, learning rate = 0.1, ite = 1000, gama = 0, max delta step = 1, and objective = “multi:softmax”

10.3.3.5 Random forest

Random forest (RF) é a combinação aleatória de múltiplas árvores de decisão, combinadas para obter uma predição mais estável e com maior acurácia (WRIGHT; WAGER; PROBST, 2016). O RF divide cada nó usando o melhor dentre um subconjunto de indicadores escolhidos aleatoriamente naquele nó. Esta estratégia, apesar de um tanto contraditória, funciona adequadamente em comparação com muitos outros classificadores, além de ser robusto a superajuste nos parâmetros. Além disso, é de fácil utilização, pois possui apenas dois parâmetros: o número de variáveis no subconjunto aleatório em cada nó e o número de árvores da floresta. Os parâmetros usados foram: bag size percent = 100, batch size = 100, number of execution slots = 1, max depth = 0 (unlimited), number of randomly chosen attributes = 0, number of iterations to be performed = 100, minimum number of instances per leaf = 1.0, minimum variance for split = 0.001, and random number seed to be used = 1.

10.3.3.6 Multilayer perceptron

Multilayer perceptron (MLP) é uma rede neural com várias camadas de neurônios ligados por meio de sinapses ponderadas, que aprende com a retro-propagação do erro de saída e atualização dos pesos (RUSSELL; NORVIG, 2009). Um MLP consiste em pelo menos três camadas de nós: uma camada de entrada, uma camada oculta e uma camada de saída. Exceto os nós de entrada, cada nó é um neurônio que usa uma função de ativação não linear. O MLP utiliza a retro-propagação como técnica de aprendizado durante o treinamento. As múltiplas camadas e ativação não linear distinguem o MLP de um perceptron linear, conseguindo classificar dados que não são linearmente separáveis. Os parâmetros usados no MLP foram: learning rate = 0.3, momentum = 0.2, the number of epochs to train through = 500, validation set size = 0 (the network will train for the specified number of epochs), seed = 0, validation threshold = 20, and hidden layers = (number of attributes + classes)/2.

10.3.3.7 Support vector machine

O *Support Vector Machine* (SVM) é um método de aprendizado supervisionado que analisa os dados e reconhece padrões, usado para classificação e análise de regressão (VAPNIK, 1998). O SVM tem como princípio básico a construção de um hiperplano como superfície de decisão, cujas as margens de separação entre as classes seja máxima (SCHÖLKOPF et al., 2002). O hiperplano é responsável pela separação de uma superfície em um espaço multidimensional em duas dimensões, podendo essa dimensão ser até infinita (Ushizima; Lorena; de Carvalho, 2005). Os parâmetros usados no SVM foram: C = 1.0, kernel = 'rbf', degree = 3, gamma = 'scale', coef0 = 0.0, shrinking = True, probability = False, tol = 0.001, cache_size = 200, class_weight = None, verbose = False, max_iter = -1, decision_function_shape = 'ovr', break_ties = False, random_state = None.

10.4 Trabalhos Relacionados

O desenvolvimento de sistemas CAD para auxílio no diagnóstico médico usando CNN tem se mostrado bastante promissor. As CNNs podem ser implementadas em sistemas CAD para extração de características e classificação, ou extração de características. A eficiência de um sistema CAD está relacionado as técnicas que os compõem. Nesse sentido, a literatura mostra estudos usando CNN para diagnóstico da COVID-19.

O trabalho desenvolvido por Ozkaya, Ozturk e Barstugan (2020a) usa técnicas de fusão e de classificação na análise de imagens de TC da COVID-19 através de características profundas. O uso de técnicas de tomografia computadorizada (TC), fornece um diagnóstico precoce da doença por meio de resultados rápidos e precisos. Porém radiologistas no uso de suas atribuições declara que a COVID-19 apresenta comportamentos diferentes nas tomografias já realizadas. O trabalho apresentado fornece uma proposta de dois modelos de análise, o primeiro método apresentado usa *patches* 16x16 (subconjunto-1) com uso de 150 imagens de TC para geração do conjunto de dados. Foram rotuladas 3000 imagens de *patches* de COVID-19 e nenhum resultado foi alcançado na fase inicial de treinamento do teste. Já o segundo método alcançou bons resultados, onde foi utilizada a mesma metodologia e quantidade de imagens, porém a quantidades de *patches* foram aumentadas para 32x32 (Subconjunto-2), com uso de fusão de características para classificação e aumento do desempenho. O modelo de dados foi processado por uma Máquina de Vetor de Suporte (SVM) com uso de outros modelos pré-treinados através da Rede Neural Convolutional (CNN) para o aprendizado por transferência. Alcançando assim resultados com acurácia de 98,27%, sensibilidade de 98,93%, especificidade de 97,60%, precisão de 97,63%, pontuação F1 de 0,982 e métricas de coeficiente de correlação de Matthews (MCC) de 96,54%.

Em Wang et al. (2020a), é utilizada uma metodologia para rastrear o vírus em imagens de Tomografia Computadorizada (TC) por meio de um algoritmo de aprendizado profundo. Para alcançar os objetivos foram coletados 1.065 imagens de TC de casos da COVID-19. Destas 1065, foram testadas positivas para o novo coronavírus 325 imagens e 740 imagens foram diagnosticadas com pneumonia viral. Onde usou a modificação do modelo de aprendizado do *Inception* para organização do algoritmo, ou seja para a validação interna e externa. O resultado obtido pela validação interna alcançou uma acurácia total de 89,5%, com especificidade de 0,88, sensibilidade de 0,87, F-score de 0,77 e um índice kappa de 0,69. Já o dados de teste externo mostrou uma acurácia total de 79,3%, com especificidade de 0,83, sensibilidade de 0,67, F-score de 0,63 e um índice kappa de 0,48. Além do mais foram testadas 54 imagens da COVID-19 e os dois primeiros resultados do teste de ácido nucleico foram negativos, 46 foram previstos como COVID-19 positivos pelo algoritmo, com precisão de 85,2%.

A eficiência da aprendizagem profunda para o diagnóstico da COVID-19 com base em tomografias computadorizadas destacado por He et al. (2020a). Onde foi proposto e desenvolvido um conjunto de dados de algumas centenas de tomografia que testaram positivas para a COVID-19. A fim de que pudessem criar métodos de aprendizado profundo a partir de imagens de tomografias para o diagnóstico da COVID-19. Foi usada uma abordagem de *Self-Trans* para integração de aprendizagem auto-supervisionada e com transferência de aprendizagem para redução do risco de sobreajuste. A partir das diversas experiências realizadas a abordagem proposta alcançou um desempenho superior com uso da *Self-Trans* em comparação com outras bases e com isso apresentou uma F1 de 0,850 e uma AUC de 0,940 no diagnóstico de COVID-19 a partir de tomografias.

Em [Wang et al. \(2020\)](#), desenvolveu um método para o diagnóstico automático da COVID-19 em TC do tórax baseado em aprendizado profundo usando uma GPU dedicada. Utilizaram uma estrutura de aprendizado profundo supervisionado por fraco usando volumes de TC 3D para classificação da COVID-19 e localização de lesões. Foi realizado a segmentação da região pulmonar usando uma UNet pré-treinado, em seguida, foi inserida em uma rede neural profunda 3D para classificação. Foram utilizados 499 volumes de TC para treinamento e 131 volumes de TC para teste. A metodologia obteve uma área sob a curva ROC de 0,959 e área sob a curva de precisão-recall 0,976. Ao usar um limiar de probabilidade de 0,5 para classificar COVID-positivo e COVID-negativo, o algoritmo obteve uma acurácia de 90,1%, um valor preditivo positivo de 0,840 e um valor preditivo negativo muito alto de 0,982.

O modelo proposto por [Zhao et al. \(2020a\)](#), apresenta um conjunto de dados de tomografia computadorizada para detecção da COVID-19, pois as mesmas apresentam resultados promissores no fornecimento de exames e testes com grande grau de precisão, rápido e baratos para o diagnóstico da COVID-19. Foi proposto um conjunto de dados contendo 275 exames de Tomografias Computadorizadas (TC) positivas para a COVID-19. Com isso foi desenvolvido e treinado métodos de aprendizado profundo para detectar se uma pessoa possui o vírus ou não analisando apenas imagens de TCs. Após o treinamento de uma rede neural convolucional foi alcançado um F1 de 0,850, que é um desempenho promissor.

[Wang, Liu e Dou \(2020\)](#) propôs o uso de uma *Redesign COVID-Net* para diagnóstico da COVID-19, usando o conjunto de dados *SARS-CoV-2 CT-scan*. A metodologia obteve uma acurácia de 90,8%, precisão de 95,7% e um F-Score de 0,908. [Pathak, Shukla e Arya \(2020\)](#) propôs uma rede de memória de curto prazo bidirecional profunda com modelo de rede de densidade de mistura (DBM) com ajuste de hiperparâmetros do modelo DBM usando um algoritmo *Memetic Adaptive Differential Evolution* (MADE). A metodologia MADE-DBM proposta foi testada no conjunto de dados *SARS-CoV-2 CT-scan*, apresentando uma acurácia de 98,3%, recall de 98,7% e um F-Score de 0,981.

A metodologia que [Farid, Selim e Khater \(2020\)](#) propôs esta focado em apresentar uma nova abordagem para análise de características de imagens de tomografia computadorizada, para rastreamento do coronavírus (COVID-19) em imagens analisadas. Aborda uma nova técnica para classificação e previsão no diagnóstico das imagens de tomografias computadorizada mais importantes em relação ao COVID-19. A principal vantagem em utilizar este método consiste em prever a probabilidade de recorrências em casos sem recorrência (detecção pela primeira vez) na aplicação da abordagem proposta para a extração de características. É aplicado às ferramentas convencionais de estatística e aprendizado de máquina para a extração de características de imagens de TC por meio de quatro filtros de imagens em combinação com a extração de características híbridos compostos (CHFS). As características selecionados foram classificados pelo sistema de classificação híbrida de pilha (SHC). O processo de classificação reduziu a taxa de falso-negativos com alta precisão ao usar um *Naïve Bayes* como um classificador em um método de classificação híbrida com 96,07% em comparação com 48,66% de uso individual. O modelo alcançou resultados que mostram a classificação precisa das imagens TC-COVID-19.

Como pode ser observado, resolver o problema de classificação de TC em COVID-19 e Não-COVID-19 não é uma tarefa simples. O uso de CNN para extração e classificação de características requer um número grande de imagens e treinamento de diversos parâmetros para criar modelos. O uso de conjuntos pequenos de imagens pode levar ao ajuste excessivo do modelo, tendo um bom desempenho nos dados de treinamento, mas generalizando mal os dados de teste. Assim, é proposto uma abordagem para diagnóstico da COVID-19 em imagens de TC usando CNN para extração de características e múltiplos classificadores.

10.5 Metodologia

A metodologia proposta neste trabalho consiste em desenvolver um método computacional para classificar imagens de TC em COVID-19 e Não-COVID-19. O método proposto é apresentado na Figura 73.

10.5.1 Aquisição de imagem

COVID-CT é um conjunto de imagens de TC desenvolvido por Zhao et al. ([ZHAO et al., 2020a](#)) para classificação binária de COVID-19. O conjunto é composto por 708 TCs, sendo 312 COVID-19 e 396 Não-

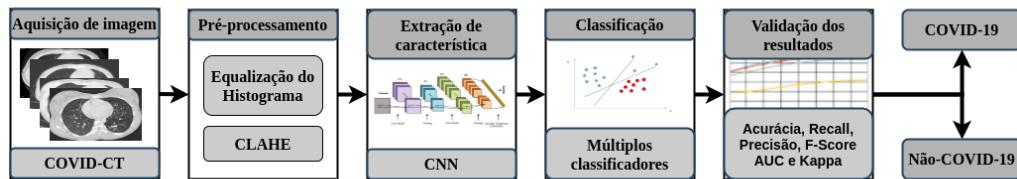


Figura 73 – Metodologia proposta.

COVID-19. Figura 74 mostra um exemplo de imagens do COVID-CT, onde, em (a) tem-se um exemplo de TC COVID-19 e em (b) tem-se um exemplo de TC Não-COVID-19.

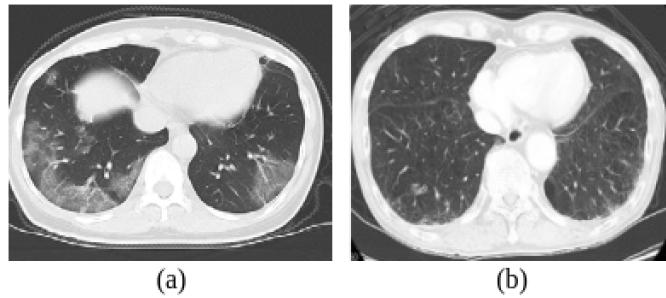


Figura 74 – Exemplos de imagens do COVID-CT, (a) TC COVID-19 e (b) TC Não-COVID-19.

10.5.2 Pré-processamento de imagem

O pré-processamento de imagem visa melhorar a qualidade da imagem. COVID-CT é um conjunto de imagens coletadas de artigos que podem apresentar perda de qualidade. Nesse sentido, a equalização do histograma e o CLAHE foram usados para melhorar a qualidade das imagens do COVID-CT.

- Equalização de histograma é um método para melhorar o contraste geral de uma imagem (Yadav; Maheshwari; Agarwal, 2014). Dada uma imagem em tons de cinza x e o número de ocorrências de nível de cinza n_i , a probabilidade de ocorrência de um pixel i na imagem x é dado pela Equação 10.1.

$$p_x(i) = p(x = i) = \frac{n_i}{n}, 0 \leq i < L \quad (10.1)$$

Onde, L é o número total de níveis de cinza, n o número total de pixels e $p_x(i)$ o histograma da imagem para o valor do pixel i , normalizado para 0 ou 1. Figura 75(b) apresenta uma imagem de COVID-CT com o histograma equalizado.

- O contraste de equalização adaptativa de histograma limitada (CLAHE) é um método que melhora o contraste local de uma imagem (PIZER et al., 1987; ZUIDERVELD, 1994). O CLAHE realiza a equalização do histograma de subáreas não sobrepostas da imagem, corrigindo inconsistências por interpolação. Neste método, uma função de aprimoramento é aplicada sobre todos os pixels vizinhos e a função de transformação é derivada (Park; Cho; Choi, 2008). Figura 75(c) mostra uma imagem de COVID-CT aplicada a CLAHE.

10.5.3 Extração de característica

Após a aquisição das imagens, e aplicado o pré-processamento, foi realizado a extração das características usando CNN. Neste trabalho, foi utilizado uma CNN simples para extração das características. A estrutura da rede consiste em: uma camada de entrada (Entrada), duas camadas de convolução (C1, C2), duas camadas

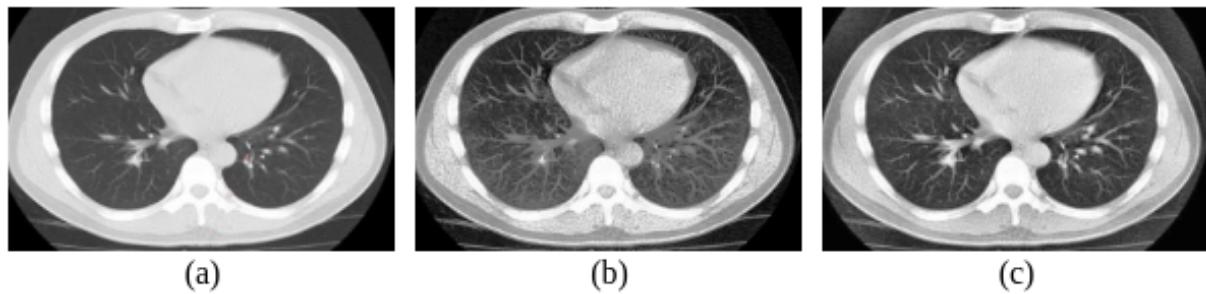


Figura 75 – Exemplo de imagem pré-processada, (a) original, (b) histograma equalizado e (c) CLAHE.

pooling (S_1, S_2), uma camada totalmente conectada e uma camada de saída (Saída). A estrutura da CNN para extração de características é apresentado na Figura 76. A Tabela 65 apresenta um resumo das camadas da CNN. Como um núcleo de convolução da camada de convolução pode extrair apenas uma característica dos mapas de recursos de entrada, ele exige vários *kernels* de convolução para extrair diferentes características.

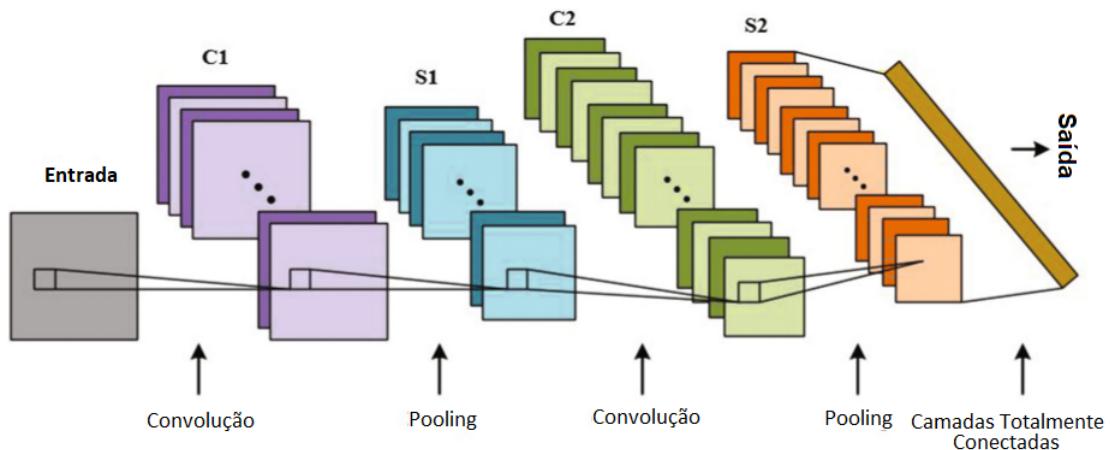


Figura 76 – Estrutura da CNN para extração de características.

Tabela 65 – Resumo das camadas da CNN.

Camada	Nº de Kernels	Tamanho do Kernel	Ativação
Entrada	1	128 × 128	-
Convolução2D	32	5 × 5	ReLU
Maxpooling2D	-	2 × 2	-
Convolução2D	32	5 × 5	ReLU
Maxpooling2D	-	2 × 2	-
Totalmente Conectado	1024	-	Dropout (0.25)
Saída	2	-	SoftMax

Para utilizar a CNN como extrator de características, foi removido a última camada totalmente conectada da rede (a camada que computa a probabilidade da imagem de entrada pertencer a umas das classes predeterminadas) e utilizada a última camada totalmente conectada da rede como um vetor de características que descrevem a imagem de entrada. As características extraídas das imagens da base podem ser utilizadas juntamente com um classificador que requeira menos dados para o treinamento que uma CNN. Essa estratégia de extração de características é bastante utilizada para aplicações de imagens médicas (Zhu; Zhang; Xue, 2015; van Ginneken et al., 2015).

10.5.4 Classificação

Com as características extraídas usando a CNN, foi realizado a classificação usando múltiplos classificadores. Neste trabalho utilizou-se os classificadores *XGBoost*, *Random Forest*, *MLP* e *SVM*, definidos na Seção 10.3.3.3, juntamente com a validação cruzada de *k-fold*, onde $k = 5$. Esses classificadores foram escolhidos por serem comumente empregados na literatura para classificação de imagens médicas.

10.5.5 Validação dos resultados

Para validar o modelo, foi utilizado métricas de avaliação estatísticas comumente empregadas na literatura. Essas métricas são calculadas com base na matriz de confusão, dado o número de verdadeiros positivos (VP), falsos positivos (FP), verdadeiros negativos (VN) e falsos negativos (FN), as medidas são matematicamente expressas da seguinte forma:

$$A = \frac{VP + VN}{VP + VN + FP + FN} \quad (10.2)$$

$$P = \frac{VP}{VP + FP} \quad (10.3)$$

$$R = \frac{VP}{VP + FN} \quad (10.4)$$

$$F = 2 \times \frac{R \times P}{R + P} \quad (10.5)$$

Area under ROC curve (AUC) é uma medida de desempenho para o problema de classificação em várias configurações de limites. A AUC representa o quanto o modelo é capaz de distinguir entre classes. Por analogia, quanto maior a AUC, melhor o modelo para distinguir entre pacientes com doença e sem doença.

Índice Kappa mede o acordo entre os resultados apresentados pela metodologia desenvolvida e a verdade do terreno humano rotulada pelos patologistas. A escala da interpretação do valor Kappa é apresentado na Tabela 66.

Tabela 66 – Níveis de precisão da classificação de acordo com o índice Kappa.

Índice Kappa (k)	Qualidade
$K < 0.2$	Ruim
$0.2 \leq K < 0.4$	Razoável
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito bom
$K \geq 0.8$	Excelente

10.6 Resultados

Para extraer características do modelo da CNN, é necessário treinar a rede. Primeiro, os dados da imagem de entrada são normalizados e transferidos para a camada de entrada da CNN. Foi realizado um treinamento por 200 épocas, para obter uma estrutura adequada para extração de características mais robustas. Como nosso modelo requer apenas as características, podemos extraer características em qualquer camada densa intermediária com a dimensão necessária. Para este trabalho, foi extraído as características usando um modelo intermediário até a primeira camada totalmente conectada. Um conjunto de 1.024 características foi extraído de cada imagem. Após extraer as características das imagens de 708 TCs, a classificação foi realizada em COVID-19 e Não-COVID-19 usando múltiplos classificadores, com validação cruzada *k-fold*, com $k = 5$. A tabela 67 mostra os resultados obtidos com as imagens originais. Nas Tabelas 68 e 69 é apresentado os resultados obtidos com aplicação do pré-processamento nas imagens.

Tabela 67 – Resultados da metodologia proposta usando imagem normal.

Classificador	Acurácia	Recall	Precisão	F-Score	AUC	Kappa
Random Forest	95,76%	98,05%	98,05%	0,980	0,980	0,961
MLP	96,69%	96,68%	96,52%	0,966	0,966	0,932
XGBoost	95,07%	95,09%	94,99%	0,950	0,950	0,900
SVM	95,10%	95,31%	94,82%	0,950	0,953	0,900

Como pode ser visto na tabela 67, utilizando as imagens originais, as características extraídas utilizando a CNN obtiveram resultados bastante promissores. Os melhores resultados foram obtidos usando o *random forest* e o MLP. A variação dos resultados apresentado entre as métricas de avaliação é muito pequeno, mostrando a eficiência na categorização das imagens de TC em COVID-19 e Não-COVID-19. Os resultados apresentados foram bastante eficientes de acordo com o índice kappa.

Tabela 68 – Resultados da metodologia proposta usando pré-processamento com equalização de histograma.

Classificador	Acurácia	Recall	Precisão	F-Score	AUC	Kappa
Random Forest	97,18%	97,37%	96,69%	0,970	0,973	0,940
MLP	97,19%	96,98%	96,98%	0,969	0,969	0,939
XGBoost	97,18%	96,92%	97,34%	0,971	0,969	0,942
SVM	96,47%	96,80%	95,84%	0,962	0,968	0,925

Aplicando a equalização do histograma, o MLP obteve a melhor acurácia. O *random forest* e o XGBoost obtiveram resultados melhores nas outras métricas. A aplicação da equalização do histograma ajudou a melhorar a qualidade das imagens, fazendo com que a arquitetura da CNN utilizada pudesse extrair características mais robustas, que melhor discriminasse as imagens em COVID-19 e Não-COVID-19.

Tabela 69 – Resultados da metodologia proposta usando pré-processamento com CLAHE.

Classificador	Acurácia	Recall	Precisão	F-Score	AUC	Kappa
Random Forest	97,88%	97,74%	97,94%	0,978	0,977	0,956
MLP	97,19%	95,90%	97,09%	0,963	0,959	0,927
XGBoost	98,50%	98,55%	98,66%	0,985	0,985	0,971
SVM	97,88%	97,74%	97,94%	0,978	0,977	0,956

Aplicando o CLAHE, o XGBoost obteve os melhores resultados, com uma acurácia de 98,50%, recall de 98,55%, precisão de 98,66%, F-score de 0,985, AUC de 0,985 e um índice kappa de 0,971. Os demais classificadores obtiveram resultados semelhantes. Pode-se observar que os resultados foram bastante eficientes na categorização das imagens de TC para o diagnóstico da COVID-19. O índice kappa mostra que os classificadores conseguem diferenciar os dados com eficiência, possuindo uma perfeita correspondência com os rótulos estabelecidos pelos especialistas.

Pode-se verificar que o uso do pré-processamento auxiliou a CNN a obter características que melhor discriminam as imagens de TC em COVID-19 e Não-COVID-19. A metodologia usando CLAHE, CNN e XGBoost obteve os melhores resultados, com uma acurácia de 98,50%, recall de 98,55%, precisão de 98,66%, F-score de 0,985, AUC de 0,985 e um índice kappa de 0,971. O índice kappa apresentado pela metodologia proposta de acordo com a Tabela 66 mostra que os classificadores obtiveram excelentes resultados na categorização da COVID-19. O uso do CLAHE melhorou a qualidade das imagens, fazendo com que a CNN obtivesse essas características mais robustas para diagnosticar COVID-19 em imagens de TC. As características extraídas com a CNN são bastante robustas para categorizar imagens em COVID-19 e Não-COVID-19. Assim, a metodologia proposta apresentou resultados bastante promissores.

A Tabela 70 compara o resultado obtido com o método proposto com os apresentados nos trabalhos relacionados para a classificação da COVID-19. O estágio de comparação de resultados é muito complexo,

visto que muitos fatores podem influenciar em uma comparação confiável. Assim, apresentamos um resumo dos resultados obtidos alcançados no método proposto com os disponíveis nos trabalhos relacionados.

Tabela 70 – Comparação dos resultados da metodologia proposta com os trabalhos relacionados

Trabalhos	Acurácia	Recall	Precisão	F-Score
(OZKAYA; OZTURK; BARSTUGAN, 2020a)	98,2%	-	97,6%	0,982
(Wang et al., 2020)	90,1%	-	-	-
(HE et al., 2020a)	-	-	-	0,850
(WANG et al., 2020a)	89,5%	87,0%	-	0,770
(ZHAO et al., 2020a)	-	-	-	0,850
(Wang; Liu; Dou, 2020)	90,8%	85,8%	95,7%	0,908
(Pathak; Shukla; Arya, 2020)	98,3%	98,7%	-	0,981
(FARID; SELIM; KHATER, 2020)	-	-	96,07%	-
Método Proposto	98,5%	98,5%	98,6%	0,985

Como pode ser visto na Tabela 70, a metodologia proposta apresenta resultados promissores na classificação de imagens de TC em COVID-19 e Não-COVID-19, quando comparados aos apresentados em trabalhos relacionados. A metodologia proposta por Ozkaya, Ozturk e Barstugan (2020a), Pathak, Shukla e Arya (2020) obtiveram resultados semelhantes aos apresentados na metodologia proposta. A metodologia proposta apresentou resultados melhores do que usar a rede Inception Wang et al. (2020), transferência de aprendizagem He et al. (2020a) e CNN supervisionado por fraco Wang et al. (2020a). A metodologia proposta obteve resultados melhores que os apresentados por Zhao et al. (2020a), Wang, Liu e Dou (2020), Farid, Selim e Khater (2020). A metodologia proposta não requer um treinamento muito grande para a extração de características e os resultados ficaram próximo aos melhores apresentado nos trabalhos relacionados. O uso de CNN para extração de características e o XGBoost como classificador apresentou resultados bastante eficientes na categorização de imagens de TC com COVID-19 e Não-COVID-19. A utilização de diferentes abordagens torna muito complexa a comparação eficiente dos resultados da metodologia proposta com os trabalhos relacionados.

10.7 Conclusão

Neste trabalho, é apresentado uma metodologia para o diagnóstico da COVID-19 em imagens de TC. Inicialmente, o pré-processamento foi aplicado às imagens utilizando a equalização do histograma e CLAHE, em seguida foi extraído um conjunto de 1024 características de cada imagem utilizando uma CNN baseada na LeNet-5. Com o conjunto de características extraídos, é realizada a classificação utilizando múltiplos classificadores. O uso de pré-processamento ajudou a melhorar a qualidade das imagens. As características extraídas com a arquitetura de CNN utilizada foram bastante robustos, obtendo uma acurácia de 98,5%, recall de 98,5%, precisão de 98,6%, F-score de 0,985, AUC de 0,985 e índice kappa de 0,971. Assim, a metodologia proposta pode ser utilizada por um especialista no diagnóstico da COVID-19, proporcionando uma segunda opinião no diagnóstico do paciente.

A metodologia proposta ainda pode ser aprimorada, pois em trabalhos futuros, pretende-se utilizar outros conjuntos de imagens da COVID-19, para o desenvolvimento de uma metodologia completa utilizando a CNN. Além disso, usar outras arquiteturas de CNN mais robustas para extrair características das imagens de TCs, como ResNet-50, VGG16 e VGG19. Assim, pretende-se obter uma metodologia mais eficiente no diagnóstico da COVID-19, que possa auxiliar o especialista no diagnóstico final.

Uma Abordagem Baseada em *Capsule Network* para a Identificação de Glaucoma em Imagens de Retina

Autores: Antônio Oseas de Carvalho Filho; Vitória de Carvalho Brito.

Resumo: O glaucoma é uma doença ocular responsável pela segunda causa mais comum de cegueira no mundo. A necessidade de detectar a doença em seus estágios iniciais é imprescindível, visto que o tratamento tardio pode acarretar na perda de visão. Nesse contexto, métodos computacionais estão sendo desenvolvidos para auxiliar os especialistas na detecção do glaucoma através da análise de imagens oculares, a fim de conferir maior precisão ao diagnóstico e aumentar as chances do paciente receber tratamento em tempo hábil, evitando a perda da visão. Neste trabalho, apresenta-se uma metodologia para a classificação automática do glaucoma em imagens de retina utilizando a *Capsule Network* (*CapsNet*), um modelo recente de aprendizado profundo que analisa as relações espaciais hierárquicas entre as características para representar as imagens, de modo a requerer menos amostras de treinamento do que as *Convolutional Neural Networks* (*CNNs*) tradicionais para obter um desempenho eficiente na classificação. Antes da execução da *CapsNet*, aplicou-se uma etapa de pré-processamento nas imagens, com o intuito de destacar as características da retina. Os resultados mostraram-se promissores, atingindo *accuracy* de 91,89%, *recall* de 88,52%, *precision* de 96,42%, *f1-score* de 92,30%, *AUC* de 0,92, e *kappa* de 0,83. A principal contribuição desta abordagem está na obtenção de resultados promissores sem a necessidade da aplicação de técnicas para aumentar os dados e segmentar a região do disco óptico. Assim, este estudo comprovou o potencial das cápsulas em identificar as relações entre as características, mesmo diante de um conjunto de dados com apenas 556 imagens de retina.

Keywords: *Capsule Network*, Glaucoma, Imagens de Retina, *Convolutional Neural Networks*.

11.1 Introdução

O glaucoma é uma doença ocular caracterizada pela danificação gradual do nervo óptico, podendo causar a perda da visão. Depois da catarata, o glaucoma é a causa mais comum de cegueira no mundo ([KINGMAN](#),

2004b). Estima-se que, até 2040, 111,8 milhões de pessoas serão portadoras desta doença (THAM et al., 2014). No caso da catarata, é possível reverter a perda da visão através de cirurgia, por outro lado, o glaucoma provoca cegueira permanente se tratado tarde. A elevação da Pressão Intraocular (PIO) é um dos principais fatores que acarretam no desenvolvimento do glaucoma, pois, quando essa pressão aumenta, pode causar a morte das células ganglionares da retina (JUNIOR et al., 2018c). Quando isso acontece, ocorre um aumento na escavação do nervo óptico, que é um espaço central do nervo que não contém fibras nervosas, sendo assim, o aumento desta escavação se torna um indício da presença do glaucoma.

Por ser uma doença assintomática em seus estágios iniciais, o prognóstico favorável do glaucoma está relacionado à detecção precoce, bem como ao tratamento em tempo hábil (JUNIOR et al., 2018c). Uma forma comum de diagnosticar o glaucoma é verificando a PIO. No entanto, podem existir casos de glaucoma onde a PIO não é alta. Alguns exames permitem detectar o glaucoma, como a Tomografia de Coerência Óptica (OCT, do inglês *Optical Coherence Tomography*) e a Tomografia Retiniana de Heidelberg (HRT, do inglês *Heidelberg Retinal Tomography*), que fornecem imagens de alta resolução das estruturas da retina, e a Retinografia, que inclui imagens coloridas do fundo do olho. As imagens geradas por meio desses exames são analisadas por especialistas que farão o diagnóstico da doença. Contudo, essa análise demanda tempo e é um processo cansativo, tendo em vista a necessidade de analisar com cautela as alterações morfológicas no nervo óptico. Dessa forma, métodos computacionais estão sendo desenvolvidos para auxiliar os especialistas na tomada de decisão durante o diagnóstico do glaucoma, a fim de conferir uma maior precisão aos resultados e aumentar as chances do paciente receber tratamento em tempo hábil, evitando a perda da visão.

Em comparação às abordagens tradicionais, a utilização de métodos computacionais combinada à análise dos especialistas possibilita uma maior precisão na sensibilidade e especificidade dos resultados, mesmo no diagnóstico de doenças assintomáticas, como o glaucoma (ABDEL-ZAHER; ELDEIB, 2016; TSOCHATZIDIS et al., 2017). Dentre as técnicas computacionais utilizadas no auxílio ao diagnóstico do glaucoma, está o *Deep Learning*, uma sub-área do *Machine Learning*, onde os modelos computacionais identificam padrões nas imagens, para que, no processo de classificação, possam discriminá-las corretamente as classes em questão (??). As *Convolutional Neural Networks* (CNNs) são exploradas em vários domínios de imagem, inclusive, é o estado da arte em reconhecimento de imagens (??).

Um estudo recente, proposto por Sabour, Frosst e Hinton (2017), apresenta um novo modelo de *Deep Learning*, a *Capsule Network* (*CapsNet*), visando solucionar algumas desvantagens trazidas pelas CNNs tradicionais, como a grande quantidade de imagens necessárias para um bom treinamento e as operações de *pooling*, que, durante a redução da dimensionalidade das características, podem acabar perdendo informações importantes. Na *CapsNet*, os principais blocos são as cápsulas, essas estruturas analisam a relação espacial hierárquica das características para tentar representar melhor os dados. Dessa forma, ao contrário das CNNs que apenas identificam a presença das características em um objeto, as cápsulas identificam relacionamentos espaciais entre elas. Estudos recentes que aplicam a *CapsNet* apontam que é possível treinar esse modelo com menos amostras do que as CNNs tradicionais e ainda assim obter um bom desempenho (JIMÉNEZ-SÁNCHEZ; ALBARQOUNI; MATEUS, 2018; AFSHAR; MOHAMMADI; PLATANIOTIS, 2018). Além disso, Gaddipati et al. (2019) sugere que a *CapsNet* tem um comportamento mais robusto em relação ao desbalanceamento das classes, sendo esse um problema recorrente em muitas bases de imagens.

No contexto das imagens glaucomatosas, a análise das relações espaciais entre as características da retina, como por exemplo tamanho e orientação do disco óptico e distância da escavação para o disco, que é um dos fatores que indicam a presença do glaucoma, podem ser relevantes para a classificação das imagens em glaucomatosas ou saudáveis. Sendo assim, este trabalho visa explorar tais aspectos para a classificação das imagens de retina avaliando a arquitetura *CapsNet* diante de um conjunto de dados público com apenas 556 imagens de retina.

11.1.1 Objetivos

Diante do contexto apresentado, o propósito deste trabalho é explorar a *CapsNet* para a detecção automática de glaucoma em imagens da retina, de forma que a relação espacial hierárquica entre os atributos que caracterizam as imagens possa contribuir para a classificação dos dados, além de utilizar um número menor de imagens do que outras abordagens durante o treinamento para realizar esta tarefa.

De maneira específica, pretende-se:

1. Avaliar a viabilidade do uso da arquitetura *CapsNet* para a identificação da doença, utilizando bases públicas de imagens de retina, sem a necessidade de uma segmentação prévia do disco óptico;
2. Aplicar técnicas de pré-processamento nas imagens que possam realçar as características da doença; e
3. Construir uma metodologia para classificação automática do glaucoma, que possa ser aplicada no auxílio ao diagnóstico da doença.

11.1.2 Organização do capítulo

O trabalho está organizado da seguinte maneira: a Seção 11.2 aborda os conceitos fundamentais para o entendimento da metodologia proposta; a Seção 11.3 discute os trabalhos relacionados ao tema abordado neste estudo; a Seção 11.4 apresenta a metodologia aplicada para alcançar os objetivos apontados neste trabalho; a Seção 11.5 apresenta e discute os resultados obtidos com a execução da metodologia; e, finalmente, a Seção 11.6 conclui este trabalho.

11.2 Referencial Teórico

Este capítulo apresenta os conceitos fundamentais para a compreensão da proposta deste trabalho. A Seção 11.2.1 relata sobre o Glaucoma, abordando as formas atuais de diagnóstico da doença. A Seção 11.2.2 discorre sobre a etapa de pré-processamento de imagens, onde são apresentados procedimentos capazes de proporcionar um melhoramento nos aspectos visuais e estruturais da imagem. A Seção 11.2.3 contextualiza o Deep Learning, abordando sobre as CNNs e adentrando no modelo investigado neste trabalho para a detecção do glaucoma nas imagens de retina. Finalmente, a Seção 11.2.4 explana sobre as métricas de validação utilizadas para avaliar os resultados da classificação das imagens.

11.2.1 Glaucoma

O glaucoma é uma doença ocular crônica que danifica progressivamente o nervo óptico, causando perda irreversível da visão periférica se não for tratada. A detecção precoce do glaucoma é muito importante, pois permite que os pacientes busquem tratamento adequado rapidamente, o que pode minimizar os danos e a perda da visão (ARAÚJO et al., 2019). Atualmente, o glaucoma é a segunda maior causa de cegueira no mundo. Estima-se que, até 2040, 111,8 milhões de pessoas sofrerão desta doença (THAM et al., 2014).

Um exame comum para a detecção do glaucoma é a verificação da PIO. Contudo, o aumento da PIO não necessariamente diz respeito a presença de glaucoma, assim como a PIO normal também não exclui a possibilidade da existência da doença. Loewen e Tanna (2014) afirmam que, do ponto de vista da gestão de pacientes, isso significa que o diagnóstico do glaucoma deve ser baseado principalmente na análise do disco óptico, sendo realizada por meio da análise de imagens médicas do fundo da retina.

Além da medição da PIO, o diagnóstico do glaucoma pode ser feito através de exames periódicos, como o OCT e o HRT, ou através da avaliação manual do disco óptico por meio de imagens digitais do fundo do olho. A doença causa alterações morfológicas na cabeça do nervo óptico e na camada de fibras nervosas que podem ser examinadas para fornecer um diagnóstico (ARAÚJO et al., 2019). A Figura 77 ilustra exemplos de retina saudável e glaucomatosa, onde é possível observar que a elevação da PIO vai danificando progressivamente as células ganglionares da retina, que formam o nervo óptico. Dessa maneira, como essas células constituem a única saída de informações visuais da retina, essa danificação provoca a perda da visão.

O diagnóstico a partir dos exames de imagens é feito através da análise visual e interpretação dos especialistas. Sendo assim, diferentes especialistas podem fornecer interpretações distintas para um mesmo exame. Além disso, a interpretação é uma tarefa repetitiva, requerendo um grande nível de atenção sobre os detalhes presentes na imagem. Diante disso, a análise desses exames através de sistemas computacionais tem sido de grande valia para monitorar o desenvolvimento natural da doença e fornecer uma segunda opinião sobre o diagnóstico.

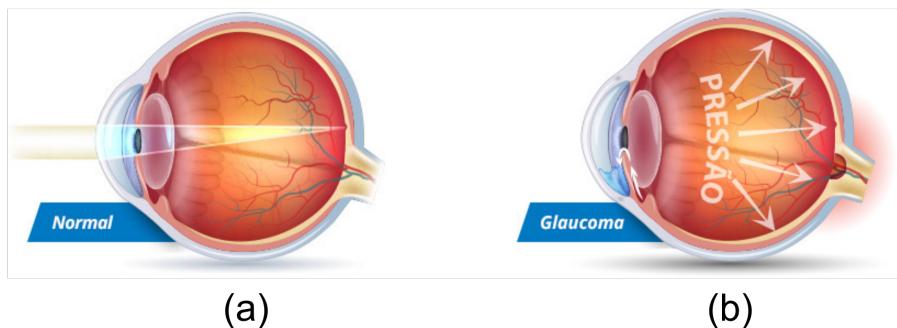


Figura 77 – Exemplos de (a) retina saudável e (b) glaucomatosa. Fonte: [Ghanem \(2020\)](#).

11.2.2 Pré-processamento de imagens

O pré-processamento de imagens é uma etapa do sistema de Visão Computacional que aplica procedimentos capazes de proporcionar um melhoramento nos aspectos visuais e estruturais da imagem ([FILHO et al., 2016b](#)). Dessa maneira, consegue-se aumentar contraste, eliminar ruídos, etc. Nos problemas de *Machine Learning*, é comum a utilização de conjuntos de dados que não estão padronizados, o que muitas vezes pode ser resultado da aquisição dos dados provenientes de fontes distintas. Dessa forma, é crucial que os dados sejam tratados antes de serem aplicados na construção do modelo.

No âmbito das imagens, a etapa de pré-processamento pode ser uma tarefa simples, como o redimensionamento de imagens. Para alimentar uma CNN com um conjunto de dados de imagens, por exemplo, é necessário que todas as imagens possuam a mesma dimensão. Outras tarefas de pré-processamento podem incluir transformações geométricas, aplicação de filtros, conversão dos canais da imagem, normalização dos pixels, entre outras.

Neste trabalho, aplicou-se um método de pré-processamento denominado equalização de histograma ([KRUTSCH; TENORIO, 2011](#)), que proporciona uma distribuição uniforme no histograma, de forma a melhorar o contraste da imagem. Isso acontece porque a equalização distribui os valores de intensidade mais frequentes, possibilitando que regiões que antes apresentavam um contraste menor, agora obtenham um maior contraste. Assim, pode-se evitar que a variação acentuada no contraste das imagens influencie na análise das características ([BISNETO; FILHO; MAGALHÃES, 2020](#)). Na Figura 78, tem-se um exemplo de uma imagem de retina antes e depois da aplicação da equalização de histograma, onde é possível observar que o contraste foi realçado e que algumas regiões, principalmente a escavação do disco óptico, ficaram bem destacadas.

11.2.3 Deep Learning

O *Deep Learning* é um ramo do *Machine Learning* na qual modelos computacionais compostos por várias camadas de processamento aprendem representações de dados com vários níveis de abstração. Esses métodos melhoraram o estado da arte em reconhecimento de fala, reconhecimento visual de objetos, detecção de objetos e vários outros domínios, como descoberta de medicamentos e genômica (??).

O *Deep Learning* busca identificar uma estrutura complexa em grandes conjuntos de dados usando o algoritmo de *backpropagation* para indicar como uma rede deve alterar seus parâmetros internos para diminuir a taxa de erro durante o aprendizado. As CNNs são algoritmos de *Deep Learning* que trouxeram avanços no processamento de imagens, vídeo, fala e áudio, enquanto as redes recorrentes se destacaram em dados sequenciais, como texto e fala (??).

Uma CNN é uma arquitetura baseada no processamento de dados visuais, capaz de aplicar filtros nesses dados, mantendo a relação de vizinhança entre os pixels da imagem ao longo do processamento da rede (??). Essa operação é conhecida como convolução, onde ocorre a somatória do produto ponto a ponto entre os valores de um filtro e cada posição da vizinhança do pixel de entrada.

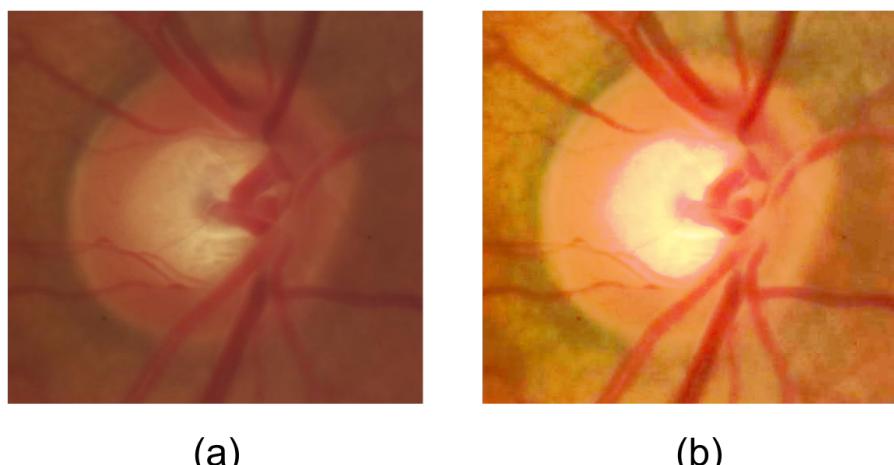


Figura 78 – Exemplos de uma imagem de retina (a) antes e (b) depois da equalização de histograma. Imagens disponíveis na base RIM-ONE v2 ([FUMERO et al., 2011b](#)).

Um dos primeiros projetos de CNNs foi a *LeNet*, proposta por ??), que contribuiu para impulsionar a área de *Deep Learning*. Inicialmente, essa arquitetura foi utilizada para reconhecimento de caracteres, tais como código postal e dígitos numéricos. Novas arquiteturas foram propostas nos últimos anos como forma de melhoria da *LeNet*, embora as versões de CNNs melhoradas compartilhem os conceitos fundamentais, como observado na Figura 79.

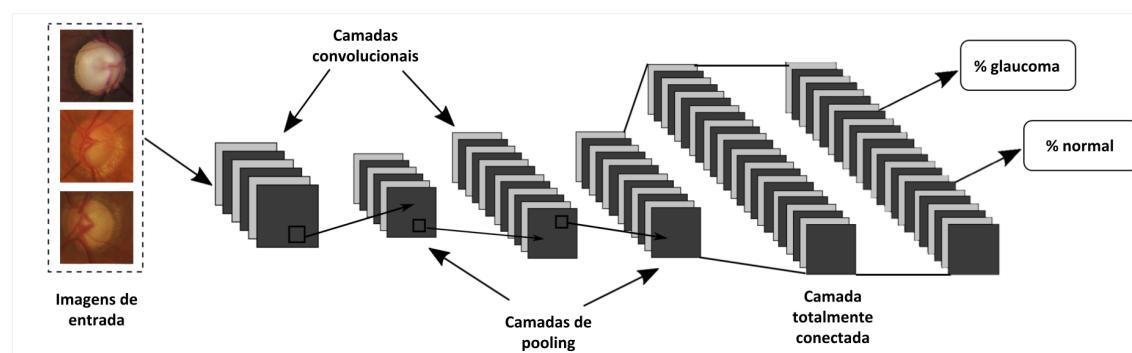


Figura 79 – Ilustração da arquitetura *LeNet* para classificação de imagens de retina, com suas três principais camadas: convolucionais, de *pooling* e totalmente conectadas. Adaptado de [Araújo et al. \(2017b\)](#).

Após uma camada convolucional, geralmente existe uma camada de *pooling*. O objetivo dessa camada é reduzir progressivamente a dimensionalidade das imagens de entrada, diminuindo o custo computacional da rede e evitando o *overfitting*. A saída das camadas convolucionais e de *pooling* representam as características extraídas da imagem de entrada. O objetivo das camadas totalmente conectadas é utilizar essas características para classificar a imagem em uma classe previamente especificada ([ARAÚJO et al., 2017b](#)).

Novas arquiteturas de CNNs que surgiram na última década são basicamente formadas pelas mesmas camadas presentes na *LeNet*, mas variam em relação ao propósito da classificação, à quantidade de camadas convolucionais, entre outros fatores. Em meio a diversidade dessas arquiteturas, este projeto investigou a utilização da *CapsNet*, que será detalhada a seguir.

11.2.3.1 Capsule Network

A *CapsNet* foi sugerida por [Sabour, Frosst e Hinton \(2017\)](#) como um novo modelo de rede neural para a tarefa de classificação de imagens. De acordo com [Sabour, Frosst e Hinton \(2017\)](#), este modelo tem como principal vantagem a preservação das relações espaciais hierárquicas dos atributos presentes na imagem, de forma a identificar não somente a presença de uma determinada característica, mas também a sua relação espacial com as demais características do objeto. Isso é feito por meio de vetores que carregam consigo informações como posição, tamanho e orientação do objeto ([SHAHROUDNEJAD et al., 2018](#)). Dessa forma, os autores afirmam que a *CapsNet* pode ser treinada com menos amostras e ainda assim alcançar um desempenho satisfatório.

Como visto em [Moraes \(2018\)](#), a identificação de elementos de interesse nas redes neurais tradicionais é feita através das ativações neuronais. Essas ativações são derivadas das somas ponderadas dos valores de suas arestas incidentes e o resultado desta soma é um único valor escalar, o que permite apenas informar a confiança da ocorrência de um elemento na região coberta pelo neurônio. Em contrapartida, a *CapsNet* utiliza vetores que permitem descrever mais do que somente a confiança de ocorrência dos elementos, eles carregam consigo informações de instanciação (posição, tamanho, orientação, etc.) das entidades identificadas. Assim, a *CapsNet* propõe eliminar o conceito de invariância com o de equivariância, ou seja, se a entrada sofre transformação espacial, a rede se adapta de acordo com a modificação ([SHAHROUDNEJAD et al., 2018](#)).

Na figura 80, tem-se uma representação de comparação entre as previsões da *CapsNet* e da CNN diante de uma imagem antes e depois da aplicação de modificações espaciais. Como a *CapsNet* é equivariante, se a imagem for transformada de alguma forma, os vetores de ativação também mudam de acordo. Por esse motivo, a imagem deformada não foi identificada como um panda, pois as cápsulas não concordaram que as características estão posicionadas corretamente para que a imagem seja classificada como tal. Já em relação à imagem rotacionada, as características estão posicionadas corretamente na imagem, independente da orientação, por conseguinte, a *CapsNet* identificou a entrada como panda.

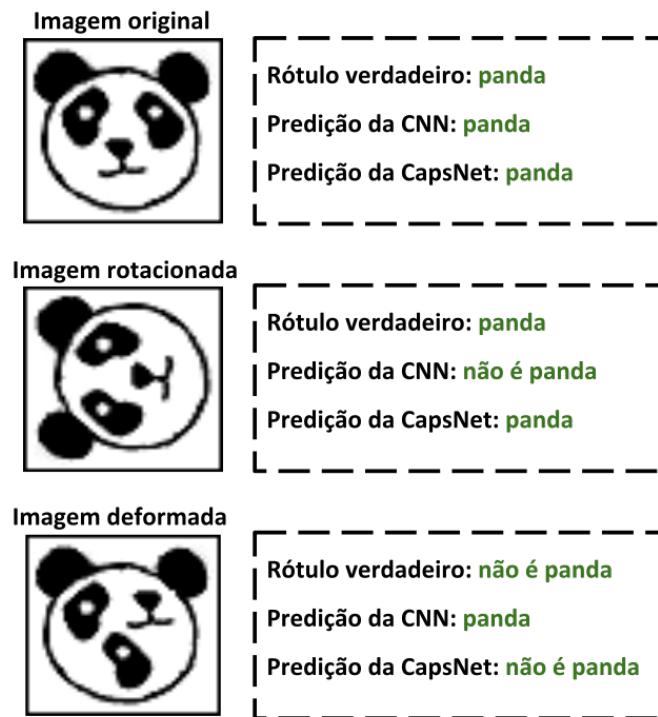


Figura 80 – Comparação representativa entre a *CapsNet* e a CNN na classificação de uma imagem de panda antes e depois de modificações. Adaptado de [Kumar \(2019\)](#).

A arquitetura original da *CapsNet* foi utilizada para a classificação de dígitos da base MNIST, que contém imagens de dimensão 28x28. O principal bloco desse modelo é a cápsula, que pode ser definida como um conjunto de neurônios organizados de forma aninhada dentro de uma camada. Esse grupo de neurônios pode ser representado como um vetor que carrega informações de instanciação, como por exemplo posição, escala, orientação, textura e cor. Esse vetor de informações passa por uma função *squashing*, onde seu cumprimento é ajustado para um valor entre 0 e 1, de modo que esse valor representará a probabilidade de que a entidade expressa pela cápsula esteja presente na atual entrada da rede.

A arquitetura original, representada pela Figura 81, utiliza uma camada convolucional, uma camada de cápsulas primárias e uma camada totalmente conectada. A primeira camada convolucional é composta por 256 filtros com convoluções de 9x9, passo 1 e função de ativação *Rectified Linear Unit* (*ReLU*). Já a segunda camada convolucional é a *PrimaryCaps*, composta de 32 cápsulas convolucionais com 8 dimensões cada, sendo que cada uma dessas dimensões se trata de uma cápsula convolucional com kernel de 9x9 e passo 2. A última camada é a *DigitCaps*, composta por uma cápsula de 16 dimensões por cada classe de dígito, sendo que cada cápsula dessas dimensões é conectada à todas as cápsulas da camada inferior. Por fim, o vetor de probabilidades das classes é calculado através da norma L2, uma regularização onde o somatório de todos os parâmetros ao quadrado é adicionado ao somatório da diferença quadrada entre a saída real e as previsões. O roteamento dinâmico é feito entre as cápsulas da *PrimaryCaps* e *DigitCaps*, é ele que irá possibilitar que uma cápsula que está em um determinado nível receba ativações de algumas cápsulas do nível abaixo e ignore outras.

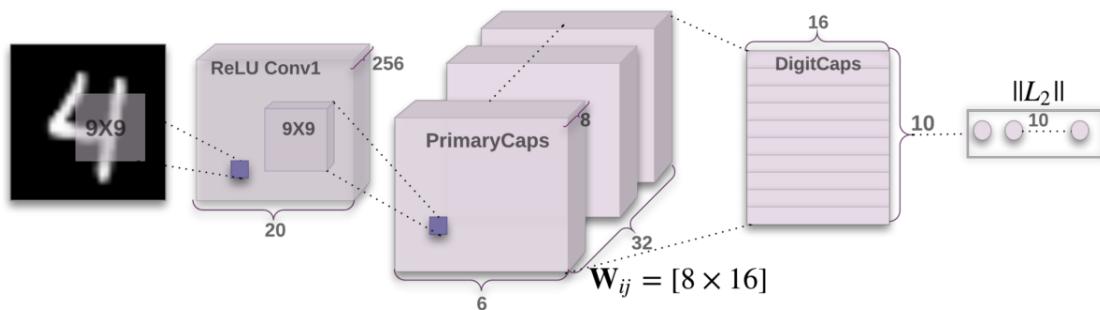


Figura 81 – Arquitetura da *CapsNet*. Fonte: [Sabour, Frosst e Hinton \(2017\)](#).

11.2.4 Métricas de Validação

Para avaliar as previsões fornecidas pelo modelo proposto neste trabalho, foram aplicadas as seguintes métricas de validação: *accuracy* (*Acc*), *recall* (*Rec*), *precision* (*Prec*), *f1-score* (*FI*), *Area Under ROC Curve* (*AUC*) ([ERKEL; PATTYNAMA, 1998](#)) e índice *kappa* ([COHEN, 1968b](#)). O cálculo dessas métricas requer valores fornecidos pela matriz de confusão, que organiza os resultados de predição em quatro classes: Verdadeiro Positivo (*VP*), Falso Positivo (*FP*), Falso Negativo (*FN*) e Verdadeiro Negativo (*VN*). A seguir, está uma breve explicação sobre as fórmulas para calcular as métricas citadas.

A *accuracy*, representada pela Equação 11.1, calcula a capacidade de um teste diagnóstico identificar verdadeiros positivos e verdadeiros negativos em uma amostra, ou seja, a proporção de acertos nos dados classificados.

$$Acc = \frac{VN + VP}{VN + VP + FN + FP}. \quad (11.1)$$

Em um teste diagnóstico, o *recall* (também conhecido como sensibilidade) mede a capacidade de identificar verdadeiros positivos, ou seja, indivíduos que têm a doença. É uma métrica fundamental na análise,

pois indica a eficácia do método. A equação 11.2 apresenta a fórmula para o cálculo do *recall*.

$$Rec = \frac{VP}{VP + FN}. \quad (11.2)$$

A *precision*, expressa pela Equação 11.3, mede a proporção de amostras positivas corretamente classificadas entre todas as classificadas como positivas.

$$Prec = \frac{VP}{VP + FP}. \quad (11.3)$$

É possível combinar *precision* e *recall* para medir o desempenho da classificação, e é isso que a *f1-score* faz. Como mostra a Equação 11.4, esta métrica calcula uma média harmônica entre os valores de *precision* e *recall*, sendo uma medida comumente utilizada em problemas de classificação onde as classes são desbalanceadas.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \quad (11.4)$$

A métrica *AUC* mede a capacidade do modelo de distinguir as classes usando o gráfico de desempenho fornecido pela curva *ROC* (*Receiver Operating Characteristic*). A curva *ROC* é calculada a partir de diferentes limiares, analisando duas métricas calculadas usando a matriz de confusão: a Taxa de Verdadeiro Positivo (TVP), equivalente ao *recall*, e a Taxa de Falso Positivo (TFP), equivalente a $1 - especificidade$, sendo que a especificidade é a métrica que calcula a proporção de verdadeiros negativos corretamente classificados. Quanto mais próximo de 1 é o valor da *AUC*, melhor é o desempenho do método na distinção das classes.

Por fim, também foi utilizado o coeficiente *kappa* para avaliar o modelo de classificação. O *kappa* mede o nível de concordância do experimento, sendo que, quanto mais próximo de 1, melhor é o resultado. A fórmula para calcular o *kappa* está representada pela Equação 11.6, onde *RandomAccuracy* é definida como a soma dos produtos de probabilidade de referência e de resultado para cada classe dividida pelo quadrado da quantidade total de amostras, como observado na Equação 11.5.

$$RandomAccuracy = \frac{(VN + FP) * (VN + FN) + (FN + VP) + (FP + VP)}{Total * Total} \quad (11.5)$$

$$Kappa = \frac{Accuracy - RandomAccuracy}{1 - RandomAccuracy} \quad (11.6)$$

De acordo com o Landis e Koch (1977c), há uma categorização dos níveis de desempenho de classificação pelo índice *kappa*, que pode ser encontrada na Tabela 71.

Tabela 71 – Categorização do desempenho da classificação, de acordo com o índice *kappa*.

Kappa (K)	Qualidade
$K < 0.2$	Ruim
$0.2 \leq K < 0.4$	Razoável
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito bom
$K \geq 0.8$	Excelente

11.3 Trabalhos Relacionados

A *CapsNet* vem sendo aplicada em muitos problemas de classificação de imagens, desde que foi introduzida como um novo modelo de aprendizado profundo, em 2017 (PATRICK et al., 2019). Neste sentido, a literatura dispõe de estudos recentes sobre a aplicação da *CapsNet* no contexto da classificação de imagens médicas. No caso do glaucoma, o trabalho de Gaddipati et al. (2019) propõe o uso da *CapsNet* para a classificação de imagens 3D de OCT em glaucomatosas e normais, avaliando o método em uma base privada composta por 233 exames. O próprio autor afirma que a análise desses exames não é uma tarefa simples, uma vez que o

modelo se torna computacionalmente complexo por lidar com as imagens tridimensionais, além do fato de existir uma baixa disponibilidade de conjuntos de dados de OCT.

Por outro lado, arquiteturas tradicionais de *Deep Learning* continuam sendo aplicadas no auxílio ao diagnóstico do glaucoma. O trabalho de [Phan et al. \(2019\)](#) comparou 3 modelos de *CNNs* na tarefa de classificação de imagens de retina: a VGG19, a ResNet152 e a DenseNet201. Neste experimento, [Phan et al. \(2019\)](#) utilizou um conjunto de 3312 imagens, adquiridas de duas bases distintas, e, durante os testes, os autores perceberam que a qualidade das imagens afetou negativamente o desempenho dos métodos na distinção das classes. Já o estudo de [Bajwa et al. \(2019\)](#) teve os experimentos divididos em dois estágios: um estágio de segmentação do disco óptico, usando uma *Regions with Convolutional Neural Network* (RCNN); e um estágio de classificação das regiões em glaucomatosas e saudáveis, usando uma *Deep Convolutional Neural Network* (DCNN). Para tanto, [Bajwa et al. \(2019\)](#) utilizou 780 imagens provindas de três bases distintas.

Pelo fato de ter usado um conjunto de dados com apenas 455 imagens, o trabalho de [Maheshwari, Kanhangad e Pachori \(2020\)](#) aplicou uma técnica de aumento de dados baseada no *Local Binary Pattern* (LBP), a fim de evitar o problema de *overfitting*. Em seguida, os autores realizaram a classificação das imagens de retina usando um modelo pré-treinado de *CNN*, a *AlexNet*. Outro estudo recente na detecção automática do glaucoma foi proposto por [Bisneto, Filho e Magalhães \(2020\)](#), onde o mesmo aplicou uma *Generative Adversarial Network* (GAN) para segmentar o disco óptico nas imagens e, em seguida, utilizou descritores de textura associados à classificadores tradicionais na literatura para classificar as regiões em glaucomatosas e normais. Os métodos propostos por [Bisneto, Filho e Magalhães \(2020\)](#) foram avaliados em um conjunto de 556 imagens, derivadas de duas bases diferentes que, inclusive, são as mesmas aplicadas neste trabalho.

A Tabela 72 aponta um resumo dos estudos relacionados, onde é possível observar que o problema de classificação do glaucoma foi explorado pela *CapsNet* em apenas um dos trabalhos. Todavia, o modelo proposto por [Gaddipati et al. \(2019\)](#) é uma arquitetura 3D que utiliza volumes de OCT, onde o mesmo aponta que há uma baixa disponibilidade de amostras desse tipo de exame, fator que muitas vezes dificulta a utilização dessas imagens em abordagens computacionais. Em comparação à abordagem descrita, o método proposto neste trabalho utiliza imagens de fundo do olho, por meio de bases publicamente disponíveis.

Em relação aos outros estudos levantados, percebe-se que as *CNNs* são amplamente aplicadas na detecção do glaucoma, com uma extensiva variedade de abordagens. Contudo, é possível observar que dos métodos que não aplicaram segmentação, um deles precisou utilizar aumento de dados no conjunto de imagens, enquanto o outro não aplicou, mas utilizou o maior conjunto de imagens entre todos os trabalhos apresentados. Isso evidencia o fato que as *CNNs* precisam de muitas amostras de treinamento para alcançar uma capacidade discriminativa satisfatória. Em comparação à esses métodos, a abordagem proposta neste trabalho utilizou um conjunto de apenas 556 imagens, todavia, sem a necessidade de uma segmentação prévia do disco óptico ou técnicas de aumento de dados.

11.4 Metodologia

Este capítulo descreve as etapas envolvidas neste trabalho para a classificação automática de glaucoma em imagens de retina usando a *CapsNet*, como levantado na Seção 11.1.1. O primeiro passo foi a aquisição das imagens de retina através de duas bases de dados publicamente disponíveis: a RIM-ONE v2 ([FUMERO et al., 2011b](#)) e a Drishti-GS ([SIVASWAMY et al., 2014b](#)). Em seguida, as imagens passaram por uma etapa de pré-processamento, onde foi aplicado um método de realce para destacar as características da retina. A etapa seguinte foi a classificação das imagens através da arquitetura *CapsNet*. Por fim, os resultados foram avaliados através de métricas comumente utilizadas na literatura. Os algoritmos implementados neste trabalho estão publicamente disponíveis no [GitHub](#). A Figura 82 ilustra o fluxo da metodologia proposta, que terá cada etapa detalhada no decorrer deste capítulo.

11.4.1 Aquisição de Imagens

Para avaliar o modelo de classificação, foram utilizadas as bases de imagens de retina RIM-ONE v2 e Drishti-GS, onde ambos os conjuntos contêm imagens de disco óptico glaucomatosas e saudáveis. De acordo com

Tabela 72 – Resumo dos trabalhos relacionados ao método proposto.

Trabalho	Método	Objetivo	Bases	Tipo de Imagem	Amostra	Métricas de Validação
Gaddipati et al. (2019)	CapsNet	Classificação	Privada	OCT	233	Accuracy = 94% Sensitivity = 89% Specificity = 96% AUC = 0,973
Phan et al. (2019)	VGG19, ResNet152 e DenseNet201	Classificação	YMU e KOSEI	Imagens de fundo do olho	3312	AUC = 0,9
Bajwa et al. (2019)	RCNN e DCNN	Segmentação e Classificação	ORIGA, HRF e OCT & CFI	Imagens de fundo do olho	780	Sensitivity = 71,17% AUC = 0,874
Maheshwari, Kanhangad e Pachori (2020)	LBP e AlexNet pré-treinada	Classificação	RIM-ONE	Imagens de fundo do olho	455	Accuracy = 98,9% Sensitivity = 100% Specificity = 97,5%
Bisneto, Filho e Magalhães (2020)	GAN e características de textura	Segmentação e Classificação	RIM-ONE e Drishti-GS	Imagens de fundo do olho	556	Accuracy = 100% Sensitivity = 100% Specificity = 100% AUC = 1
Método Proposto	CapsNet	Classificação	RIM-ONE e Drishti-GS	Imagens de fundo do olho	556	Accuracy = 91,89% Recall = 88,52% Precision = 96,42% F1-score = 92,30% AUC = 0,92 Kappa = 0,83

Bisneto, Filho e Magalhães (2020), essas bases são comumente utilizadas na comunidade científica, sendo aplicadas em diversos métodos de classificação do glaucoma.

A RIM-ONE v2 consiste em imagens fornecidas por três hospitais: Hospital Universitário de Canarias, Hospital Clínico San Carlos e Hospital Miguel Servet. Ela possui 455 imagens oculares, sendo 200 imagens de retina com glaucoma em diferentes estágios e 255 imagens de retinas saudáveis. Já a Drishti-GS é uma base destinada à validação de métodos computacionais com o mesmo propósito deste trabalho e é composta de apenas 101 imagens, onde 70 delas apresentam retinas glaucomatosas e 31 apresentam retinas saudáveis. As imagens da base Drishti-GS foram coletadas e anotadas pelo Aravind Eye Hospital, Madurai, Índia e são de uma única população (SIVASWAMY et al., 2014b). Ambos os conjuntos possuem imagens de diferentes resoluções e dimensões. Todas as imagens são de bits e estão no modelo de cores RGB (do inglês *red, green, blue*).

11.4.2 Pré-processamento

Como visto na Seção 11.2.2, antes do treinamento com a CapsNet, foi aplicado o método de equalização de histograma (KRUTSCH; TENORIO, 2011) para realçar o contraste das imagens, aprimorando aspectos estruturais e visuais das imagens de retina. De acordo com Bisneto, Filho e Magalhães (2020), como a equalização distribui os valores de intensidade mais frequentes, pode-se evitar que a variação acentuada no contraste das imagens influencie na análise das características. Para a implementação deste método, foi utilizado o módulo *exposure* da biblioteca *skimage* (WALT et al., 2014).

11.4.3 Classificação

Como indicado na Seção 11.2.3.1, este trabalho explorou um modelo de *Deep Learning* denominado CapsNet para a classificação das imagens de retina em glaucomatosas e saudáveis. O modelo implementado neste trabalho seguiu as configurações originais da arquitetura, logo, foram utilizadas uma camada convolucional, uma camada de cápsulas primárias e uma camada totalmente conectada, que possui 16 dimensões por classe, isto é, glaucoma e normal. Na Figura 83, tem-se uma representação ilustrativa da arquitetura utilizada, onde é possível notar que há uma camada convolucional (*Conv2D*), uma camada de cápsulas primárias (*Primary Capsule*) e uma camada totalmente conectada (*Class Capsule*).

As duas bases de imagens mencionadas na Seção 11.4.1 foram utilizadas na execução da metodologia, resultando em um conjunto de 556 imagens (270 glaucomatosas e 286 normais). Esse conjunto foi dividido aleatoriamente em treino, validação e teste, com uma proporção de 70%, 10% e 20%, respectivamente. Sendo

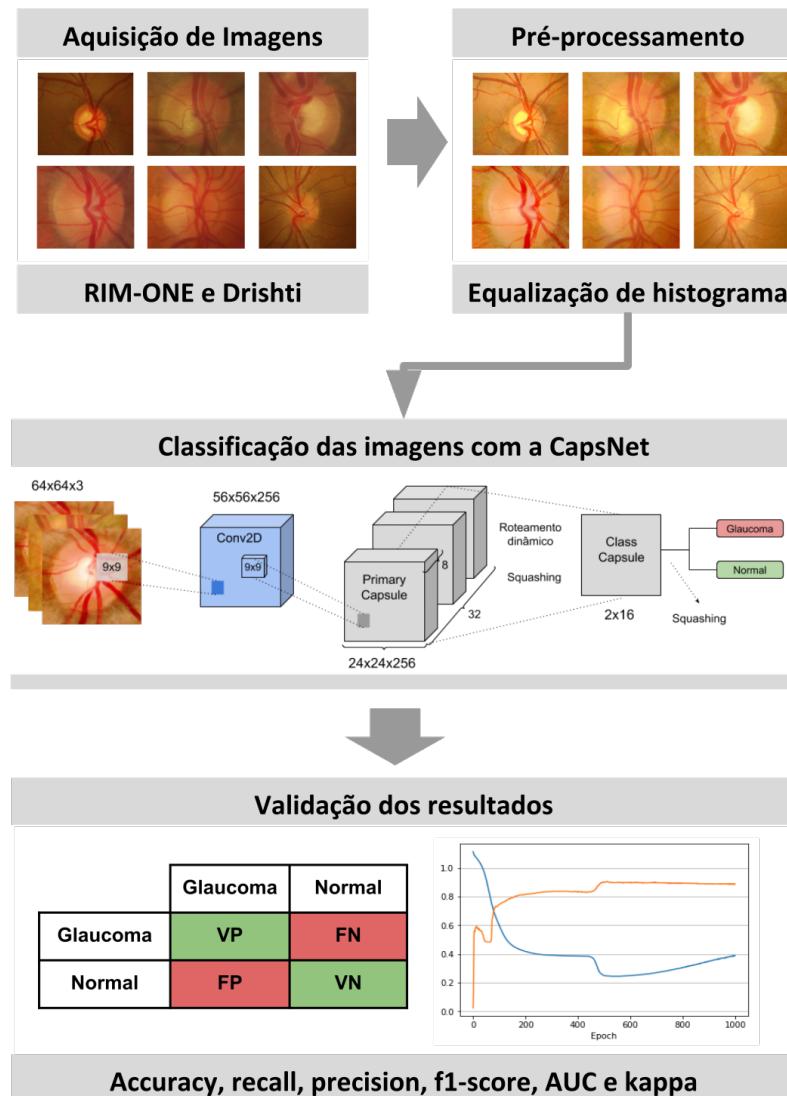


Figura 82 – Fluxo da metodologia proposta. Autoria própria.

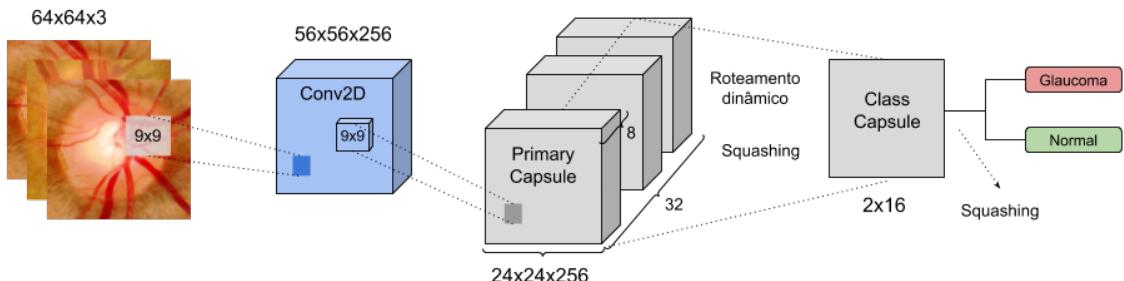


Figura 83 – Arquitetura da CapsNet aplicada neste trabalho. Autoria própria.

assim, utilizou-se 390 imagens para o treinamento, 55 para a validação e 111 para o teste. Uma vez dividido, o mesmo conjunto foi utilizado em todos os testes realizados.

A entrada da rede foi adequada para receber as imagens das bases mencionadas na Seção 11.4.1, mas antes todas as imagens foram padronizadas para uma mesma dimensão através da função de redimensionamento disponível na biblioteca *pillow* (CLARK, 2015). A próxima Seção deste trabalho apresenta os resultados obtidos com a execução do modelo em dimensões variadas das imagens, mas o melhor resultado foi alcançado quando o formato das imagens foi definido como 64x64. Em relação ao modelo de cores das imagens, além do RGB, também foram testados os canais R, G e B, separadamente, e o canal *gray*, sendo esta conversão implementada através do método *rgb2gray* da biblioteca *skimage* (WALT et al., 2014). Após a aplicação do método de pré-processamento, uma normalização das imagens foi realizada, deixando a distribuição dos valores de pixel entre 0 e 1.

Nos testes iniciais da *CapsNet*, o número de iterações do roteamento dinâmico foi definido como 3, mas a Seção de resultados mostra a execução do modelo com a atribuição de outros valores para este parâmetro. O valor 3 foi definido como base pelo fato de ter sido o melhor valor encontrado no artigo original da *CapsNet* (SABOUR; FROSST; HINTON, 2017). Contudo, é válido variar este parâmetro, tendo em vista que o problema é diferente do que foi abordado no artigo original. Os outros parâmetros definidos nos testes estão detalhados na Tabela 73. Em relação ao número de épocas, o valor 200 foi selecionado como um ponto de partida por ser um número maior de épocas do que o que normalmente tem sido aplicado nos estudos com a *CapsNet* (GADDIPATI et al., 2019; BUTUN et al., 2020; TSUJI et al., 2020; AFSHAR et al., 2020; JIANG et al., 2020; AFSHAR; PLATANIOTIS; MOHAMMADI, 2019), no entanto, convém analisá-lo nos resultados, através dos gráficos de desempenho, para comprovar se o valor é suficiente para o aprendizado. Os outros parâmetros também foram baseados na literatura.

Tabela 73 – Principais parâmetros usados no modelo.

Hiperparâmetros	Valores aplicados
Número de épocas	200
Otimizador	<i>Adam optimizer</i>
Taxa de aprendizado	0,0001
Tamanho do lote	64
Número do roteamento dinâmico	3

11.4.4 Validação

O processo de avaliação da classificação é essencial para a análise do desempenho do modelo e possíveis melhorias. Para avaliar as previsões da *CapsNet* nas imagens de retina, foram aplicadas as métricas *accuracy*, *recall*, *precision*, *f1-score*, *AUC* e *kappa*, calculadas a partir da matriz de confusão gerada, como discutido na Seção 11.2.4. Na implementação dessas métricas, foram utilizados os métodos do módulo *metrics* da biblioteca *sklearn* (PEDREGOSA et al., 2011). Essas métricas são amplamente adotadas na literatura para avaliar modelos de *Machine Learning*.

11.5 Resultados e Discussões

Este capítulo apresenta e discute os resultados alcançados através dos experimentos realizados para a classificação das imagens do fundo da retina em glaucomatosas e saudáveis utilizando a arquitetura *CapsNet*, de acordo com a metodologia apontada na Seção 11.4. A Seção está dividida em duas partes: na primeira, são apresentados os resultados obtidos com a execução da *CapsNet* em vários casos de teste, a fim de encontrar a dimensão e a representação de cores ideais para as imagens, analisar o impacto do realce de contraste e encontrar o melhor valor para o roteamento dinâmico entre as cápsulas; na segunda, o desempenho da metodologia proposta foi comparado com outras abordagens da literatura.

11.5.1 Resultados Obtidos pelo Método Proposto

11.5.1.1 Experimento 1

Como exposto na Seção 11.4, as imagens de retina passaram por um redimensionamento antes de serem impostas como entrada da rede, isso porque a arquitetura requer uma padronização na dimensão das imagens. Para encontrar a melhor dimensão das imagens para a rede, o primeiro experimento foi a execução do modelo com três dimensões diferentes: 64x64, 128x128 e 256x256. Com o aumento da altura e largura das imagens, o modelo vai ficando mais custoso computacionalmente, sendo assim, o cenário ideal seria aquele onde a dimensão da entrada não seja tão extensa mas ainda assim consiga representar bem as características do problema.

A Tabela 74 aponta os resultados da execução do modelo com as três dimensões de entrada mencionadas. Para esse primeiro teste, optou-se por utilizar o canal *gray* das imagens, que consegue carregar a importância dos três canais em apenas uma matriz. Em relação ao número de iterações do roteamento dinâmico, para esse teste foi definido o valor 3, pelo fato de ter sido o melhor valor encontrado no artigo original da CapsNet (SABOUR; FROSST; HINTON, 2017). Quanto ao número de épocas, utilizou-se 200. Por fim, nessa execução, o método de pré-processamento ainda não foi aplicado, visto que o propósito desse primeiro teste é apenas encontrar a melhor dimensão para que, posteriormente, outros parâmetros possam ser variados.

Tabela 74 – Experimento 1: avaliação do modelo em decorrência da variação da dimensão das imagens de entrada.

Dimensão	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC	Kappa
64x64	86,48	81,96	92,59	86,95	0,86	0,73
128x128	76,57	72,13	83,01	77,19	0,77	0,53
256x256	76,57	73,77	81,81	77,58	0,76	0,53

Analizando a Tabela 74, é possível observar que o modelo alcançou melhores resultados quando executado com as imagens na dimensão 64x64. Além de conseguir representar melhor as características da doença dentre todas as dimensões testadas, o valor 64x64 proporcionou ao modelo um treinamento mais rápido, uma vez que o custo computacional é intensificado conforme o aumento na dimensionalidade das imagens.

11.5.1.2 Experimento 2

Em posse da dimensão ideal para a entrada do modelo, o segundo experimento realizado foi um conjunto de testes para encontrar o melhor canal para a representação das imagens. Ademais, nesse experimento também foi avaliada a aplicação do método de pré-processamento nas imagens, a fim de comprovar se o realce das características impacta positivamente os resultados. Assim como no experimento anterior, a quantidade de épocas foi definida como 200 e o roteamento dinâmico como 3. Os resultados alcançados neste experimento podem ser visualizados através da Tabela 75.

Visualizando os resultados expostos na Tabela 75, observa-se que o RGB mostrou-se a melhor representação de cores para o reconhecimento das imagens de retina pela CapsNet. Além disso, os melhores resultados dessa representação foram alcançados, na maioria das métricas, com a aplicação do realce de contraste, obtendo *accuracy* de 88,28%, *recall* de 83,60%, *precision* de 94,44%, *f1-score* de 88,69%, AUC de 0,88 e *kappa* de 0,76. Nesse teste, o modelo apresentou resultados superiores aos demais na maioria das métricas de validação, inclusive, obteve um *kappa* muito bom, de acordo com a categorização de Landis e Koch (1977c). Diante do exposto, a aplicação do pré-processamento no canal RGB, utilizando a dimensão 64x64 nas imagens, mostrou-se a melhor configuração de entrada para a CapsNet.

11.5.1.3 Experimento 3

Até o momento, os experimentos realizados permitiram a descoberta da dimensão e canal ideais para as imagens de entrada, além da aplicação do pré-processamento, que mostrou-se relevante para o modelo.

Tabela 75 – Experimento 2: avaliação do modelo em virtude da variação dos canais da imagem executados com e sem a equalização de histograma na etapa de pré-processamento.

	Canal	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC	Kappa
Sem pré-processamento	RGB	86,48	88,52	87,09	87,80	0,86	0,72
	R	81,98	80,32	85,96	83,05	0,82	0,63
	G	81,08	67,21	97,61	79,61	0,82	0,63
	B	82,88	77,04	90,38	83,18	0,83	0,65
	Gray	86,48	81,96	92,59	86,95	0,86	0,73
Com pré-processamento	RGB	88,28	83,60	94,44	88,69	0,88	0,76
	R	74,77	73,77	78,94	76,27	0,74	0,49
	G	87,38	83,60	92,72	87,93	0,87	0,74
	B	82,88	80,32	87,5	83,76	0,83	0,65
	Gray	84,68	86,88	85,48	86,17	0,84	0,69

Outro aspecto importante para ser avaliado no modelo é o impacto do roteamento dinâmico. Como mencionado anteriormente, os testes realizados utilizaram 3 iterações no roteamento dinâmico, por ter sido o melhor valor encontrado nos experimentos do artigo original da CapsNet. No entanto, este trabalho trata-se da exploração de um problema totalmente diferente do contexto do artigo original, tornando-se necessária uma nova avaliação desse parâmetro para a obtenção de melhores resultados.

A Tabela 76 apresenta os resultados da execução do modelo diante da variação do número de iterações do roteamento dinâmico. Os resultados presentes no roteamento 3 são os melhores resultados obtidos no experimento anterior. Os valores do roteamento foram variados para menos e para mais de 3.

Tabela 76 – Experimento 3: avaliação do modelo em decorrência da variação do número de iterações do roteamento dinâmico.

Iterações do Roteamento Dinâmico	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC	Kappa
1	87,38	81,96	94,33	87,71	0,87	0,74
2	90,09	86,88	94,64	90,59	0,90	0,80
3	88,28	83,60	94,44	88,69	0,88	0,76
4	87,38	80,32	96,07	87,50	0,88	0,74
5	85,58	85,24	88,13	86,66	0,85	0,70
6	88,28	85,24	92,85	88,88	0,88	0,76
7	88,28	81,96	96,15	88,49	0,88	0,76
8	91,89	88,52	96,42	92,30	0,92	0,83
9	87,38	88,52	88,52	88,52	0,87	0,74
10	88,28	81,96	96,15	88,49	0,88	0,76

Analisando os resultados apontados na Tabela 76, é possível observar que, de fato, modificar o parâmetro do roteamento dinâmico influencia os resultados de predição do modelo. Contudo, os resultados foram bastante variados, não houve um desempenho crescente ou decrescente em função do incremento do número de iterações do roteamento. Quando o roteamento foi definido como 8, o modelo alcançou uma melhor capacidade discriminatória, atingindo *accuracy* de 91,89%, *recall* de 88,52%, *precision* de 96,42%, *f1-score* de 92,30%, AUC de 0,92 e *kappa* de 0,83, considerado excelente, consoante a categorização de Landis e Koch (1977c).

Em face do exposto, observa-se que o melhor cenário de teste encontrado foi na execução do modelo usando 8 iterações no roteamento dinâmico e recebendo como entrada as imagens RGB, redimensionadas

para 64x64 e realçadas através da equalização de histograma. Com o intuito de facilitar a compreensão dos resultados, as Figuras 84(a) e 84(b) apresentam os gráficos de *accuracy* e *loss* do treinamento e validação do modelo que alcançou os melhores resultados neste experimento, enquanto a Figura 85 representa a matriz de confusão gerada no teste.

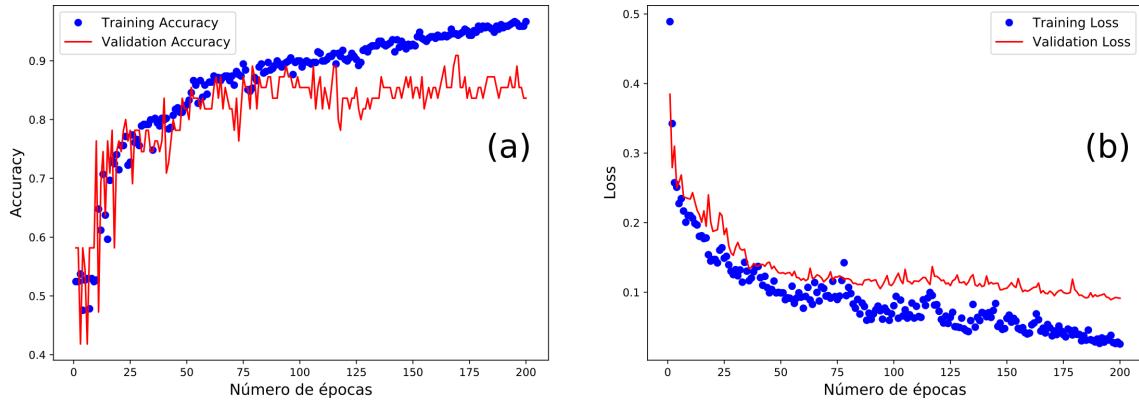


Figura 84 – Gráfico de (a) *accuracy* e (b) *loss* de treino e validação do modelo que obteve o melhor resultado no experimento 3.

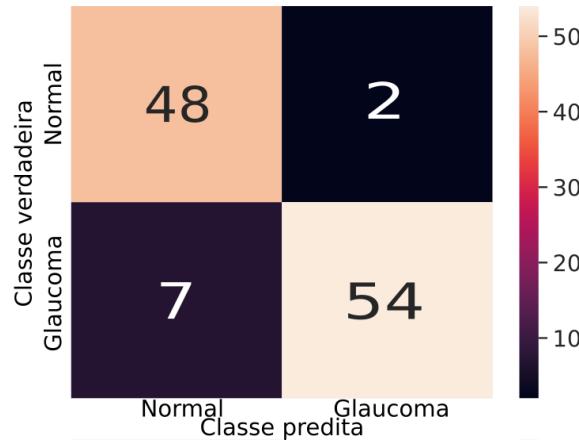


Figura 85 – Matriz de confusão gerada no teste do melhor modelo do experimento 3.

Nas Figuras 84(a) e 84(b) é possível observar que, embora as curvas de aprendizado no treino e validação tenham se distanciado um pouco por volta da época 100, a rede conseguiu aprender, já que errou apenas 8,1% dos dados de teste, com alguns falsos negativos a mais que falsos positivos, como mostra a Figura 85. Para comprovar esta afirmação de que a rede, de fato, está aprendendo, a Figura 86 apresenta as características extraídas da camada *Class Capsule* na predição do conjunto de teste. Com o propósito de facilitar a visualização das características resultantes das cápsulas, foi utilizado o algoritmo *Principal Component Analysis* (PCA) para reduzir a dimensionalidade dos dados sem perder informações importantes. O redimensionamento recebeu como entrada a saída das cápsulas, que possuem formato 2x16, resultando posteriormente ao PCA um vetor com dois componentes principais que carregam 99,92% das informações presentes nas cápsulas.

Analisando a Figura 86, com o auxílio do algoritmo PCA para o redimensionamento das características, é possível notar a capacidade de distinção das classes pela *CapsNet*. Os resultados mostram que, apesar de alguns exemplos mais difíceis (principalmente no caso do glaucoma), o treinamento foi capaz de separar as classes de forma que seja possível visualizar uma classificação binária nos dados. Dessa maneira, observa-se que a *CapsNet* mostrou-se promissora para o problema abordado, tendo em vista que, diante das imagens

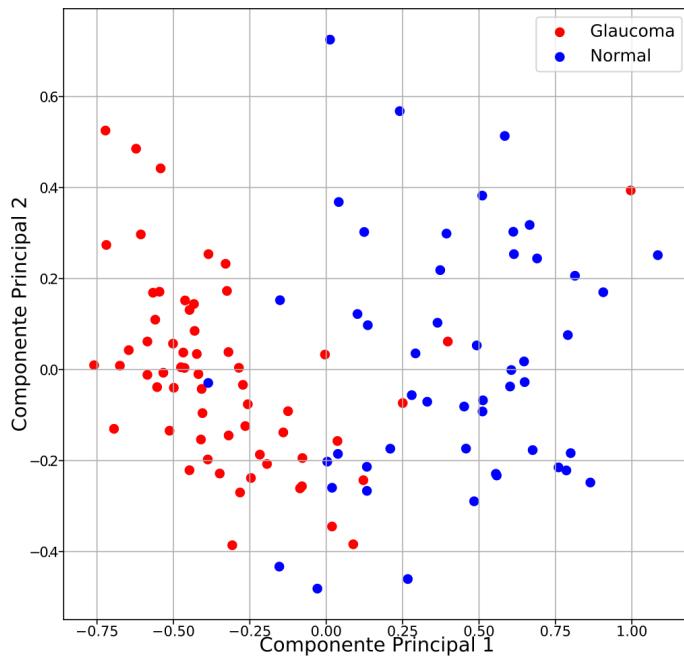


Figura 86 – Representação dos dois componentes principais gerados pelo PCA para o conjunto de características extraídas pela CapsNet no conjunto de teste do experimento 3.

realçadas e redimensionadas para 64x64 e com um roteamento dinâmico de 8, o modelo alcançou resultados consistentes e satisfatórios em todas as métricas de validação.

O número de épocas não foi variado neste trabalho pelo fato de que, como observado nas Figuras 84(a) e 84(b), por volta da época 100, a curva de validação passou a se distanciar da curva de treino. Dessa maneira, caso o treinamento continuasse com mais épocas, é provável que esse distanciamento aumentasse, pois o modelo passaria a se ajustar demais aos dados de treinamento, causando o *overfitting*. No cenário atual, percebe-se, através dos resultados apresentados, que a rede obteve um aprendizado satisfatório, uma vez que alcançou uma capacidade discriminatória das imagens de teste.

11.5.2 Comparação com Outras Abordagens

Esta seção compara o desempenho do método proposto neste trabalho com outras abordagens aplicadas ao problema de classificação do glaucoma. Na primeira comparação, foi avaliado o desempenho de três arquiteturas de CNN comumente aplicadas na literatura. Em seguida, os resultados obtidos neste trabalho foram postos em comparação com os resultados dos estudos apresentados na Seção 11.3.

11.5.2.1 Testes com Outras Arquiteturas

Os estudos apresentados na Seção 11.3 usam diferentes modelos de CNNs para a categorização de imagens de retina em saudáveis e glaucomatosas. Todavia, a intenção de realizar testes com outras arquiteturas é comprovar a eficácia desses modelos diante do mesmo conjunto de imagens utilizado neste trabalho e sem o uso de técnicas de segmentação e/ou aumento de dados.

Nesse contexto, foram realizados testes com três arquiteturas de CNN comumente aplicadas na literatura: a ResNet-50 (HE et al., 2015b), a DenseNet-121 (HUANG et al., 2017b) e a VGG16 (SIMONYAN; ZISSERMAN, 2014b). A ResNet-50 é estruturada em 50 camadas de profundidade e aplica blocos residuais para otimizar o treinamento. A DenseNet-121 é composta por camadas estreitas, geralmente 12 filtros por camada, e o classificador final toma a decisão com base em todos os mapas de características que a rede produziu. Por fim, a VGG16 é um modelo de 16 camadas que apresenta-se como uma evolução da arquitetura AlexNet, utilizando filtros de tamanho 3x3.

Os modelos foram executados em dois cenários de teste: usando transferência de aprendizado e ajuste fino. A transferência de aprendizado foi aplicada pelo fato de que as arquiteturas exploradas nesta seção são amplamente aplicadas com a inicialização de pesos treinados em outro conjunto de dados. Nesse teste, foram utilizados os modelos pré-treinados com a ImageNet (DENG et al., 2009c), um conjunto de dados com mais de 1 milhão de imagens divididas em 1.000 classes de objetos. A predição dos modelos foi feita nas bases de imagens utilizadas neste trabalho.

Utilizar apenas os modelos pré-treinados pode não proporcionar um bom desempenho na classificação das imagens nesse problema. Em razão disso, optou-se por testar também o ajuste fino, que consiste em continuar o treinamento das imagens a partir dos pesos ajustados para outro problema, como é o caso da ImageNet. Nesse segundo teste, alguns parâmetros aplicados foram os mesmos utilizados na metodologia deste trabalho, como: número de épocas, taxa de aprendizado e otimizador. É importante ressaltar que os mesmos conjuntos de treinamento, validação e teste foram utilizados tanto na metodologia proposta quanto nos algoritmos de comparação, com a diferença apenas na dimensão das imagens, já que as arquiteturas dessa comparação possuem um formato de entrada diferente da CapsNet. Ademais, as imagens foram utilizadas no canal RGB e realçadas pela etapa de pré-processamento, uma vez que esse mostrou-se o melhor cenário neste trabalho. Os resultados podem ser conferidos na Tabela 77.

Tabela 77 – Resultados obtidos com a execução de outras arquiteturas de CNNs.

	Arquitetura	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC	Kappa
Transferência de aprendizado	ResNet-50	48,64	50,00	24,32	32,00	0,50	0,00
	DenseNet-121	54,95	55,04	55,00	54,86	0,55	0,09
	VGG16	47,74	47,95	47,96	47,00	0,50	0,00
Ajuste fino	ResNet-50	54,95	50,00	27,47	35,00	0,50	0,00
	DenseNet-121	50,45	50,40	50,40	50,00	0,50	0,00
	VGG16	54,95	50,00	27,47	35,00	0,50	0,00

Na Tabela 77, é possível perceber que as CNNs tradicionais não conseguiram distinguir com êxito as classes do problema em questão. Nenhum dos resultados mostrou-se eficiente na categorização das imagens, inclusive, todos os modelos apresentaram um *kappa* ruim, de acordo com a categorização de Landis e Koch (1977c). Em relação à transferência de aprendizado, os resultados mostraram que apenas os pesos da ImageNet não foram suficientes para que a rede identificasse corretamente as classes do problema. Quanto ao ajuste fino, acredita-se que os resultados ruins são motivados pelo fato de que esses modelos requerem uma quantidade maior de amostras de treinamento para classificar as imagens corretamente. A DenseNet-121, por exemplo, possui muitos pesos, sendo assim, seu aprendizado acaba se tornando difícil em decorrência dos poucos exemplos de treinamento. Por esse motivo, as técnicas de aumento de dados são amplamente aplicadas quando o conjunto de treinamento não é vasto. Em contrapartida, diante de um mesmo cenário com poucas imagens de treinamento, o método proposto neste trabalho mostrou-se promissor na tarefa de classificação de imagens, enfatizando a capacidade das cápsulas de representar melhor as entidades das imagens, sem a necessidade de um amplo conjunto de treinamento.

11.5.2.2 Comparação com os Trabalhos Relacionados

A Tabela 78 resume a comparação dos resultados obtidos neste trabalho com os estudos descritos na Seção 11.3. Como *sensitivity* e *recall* apresentam a mesma fórmula, os resultados de *sensitivity* dos trabalhos foram considerados como *recall* na tabela, para padronizar a comparação.

Analisando os resultados da Tabela 78, nota-se que o método proposto superou os resultados apenas dos estudos de Phan et al. (2019) e Bajwa et al. (2019). Entretanto, o método proposto obteve um resultado satisfatório sem a necessidade de uma etapa de segmentação, ao contrário das abordagens de Bajwa et al. (2019) e Bisneto, Filho e Magalhães (2020), e sem a necessidade de uma técnica de aumento de dados, ao contrário da metodologia proposta por Maheshwari, Kanhangad e Pachori (2020).

Tabela 78 – Comparação do método proposto com os trabalhos relacionados.

Trabalho	Método	Bases	Amostra	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC	Kappa
Gaddipati et al. (2019)	CapsNet	Privada	233	94,00	89,00	-	-	0,97	-
Phan et al. (2019)	VGG19, ResNet152 e DenseNet201	YMU e KOSEI	3312	-	-	-	-	0,90	-
Bajwa et al. (2019)	RCNN e DCNN	ORIGA, HRF e OCT & CFI	780	-	71,17	-	-	0,87	-
Maheshwari, Kanhangad e Pachori (2020)	LBP e Alexnet pré-treinada	RIM-ONE	455	98,90	100	-	-	-	-
Bisneto, Filho e Magalhães (2020)	GAN e características de textura	RIM-ONE e Drishti-GS	556	100	100	-	-	1	-
Método Proposto	CapsNet	RIM-ONE e Drishti-GS	556	91,89	88,52	96,42	92,30	0,92	0,83

Outro ponto importante é que apenas o trabalho de Gaddipati et al. (2019) aplica a CapsNet para a classificação de glaucoma em um conjunto menor de imagens, se comparado ao utilizado neste trabalho. Contudo, esse estudo avalia imagens 3D de OCT em sua metodologia, sendo que, conforme mencionado anteriormente, a disponibilidade desse tipo de imagem não é tão ampla quanto no caso das imagens de fundo da retina, além do fato de que o método torna-se computacionalmente complexo por lidar com muitas camadas de imagens.

11.5.3 Discussões

O objetivo deste trabalho em aplicar a arquitetura CapsNet foi avaliar se as cápsulas poderiam identificar as relações entre as características da retina de forma que um conjunto reduzido de imagens de treinamento fosse suficiente para que o modelo atingisse uma classificação satisfatória dos dados. Conforme observado na Seção 11.5.1, a CapsNet, em união ao método de pré-processamento aplicado às imagens, apresentou resultados promissores, mas com espaço para ajustes adicionais a fim de melhorar a separabilidade entre as características. A seguir, destaca-se alguns pontos essenciais sobre este trabalho:

- A abordagem proposta neste trabalho faz uso de um modelo de *Deep Learning* proposto com o intuito de solucionar as desvantagens que as CNNs tradicionais carregam, mas ainda é um modelo pouco explorado no contexto das imagens oculares, de acordo com os estudos levantados na Seção 11.3;
- Durante os testes, observou-se que a CapsNet, no problema das imagens oculares, é sensível ao canal de cor utilizado;
- Uma etapa de pré-processamento foi aplicada nas imagens de retina. Os resultados mostraram que o realce de contraste afetou positivamente a capacidade discriminatória das cápsulas;
- Com base nos resultados obtidos, nota-se que as relações entre as características das retinas são relevantes para a ativação das cápsulas;
- Os testes comparativos apresentados na Seção 11.5.2.2 mostraram que a CapsNet obteve resultados superiores às demais arquiteturas aplicadas ao mesmo problema;
- Em relação à dimensão das imagens, acredita-se que a dimensão 64x64 proporcionou à rede melhores resultados em decorrência do fato de que, por ser uma arquitetura composta por poucas convoluções e nenhuma camada de *pooling*, imagens com dimensões maiores podem desfavorecer o aprendizado da CapsNet.

11.6 Conclusão

Este trabalho propôs a aplicação da arquitetura CapsNet para a classificação de imagens de retina em glaucomatosas e saudáveis. Na abordagem proposta, foram utilizadas duas bases de imagens da imagem

do fundo da retina, que passaram por uma etapa de pré-processamento, onde aplicou-se a equalização de histograma para melhorar o contraste das imagens, a fim de destacar as estruturas da retina. Em conjunto com a etapa de pré-processamento aplicada ao canal RGB das imagens, a CapsNet alcançou *accuracy* de 91,89%, *recall* de 88,52%, *precision* de 96,42%, *f1-score* de 92,30%, AUC de 0,92 e *kappa* de 0,83, considerado excelente, consoante a categorização de Landis e Koch (1977c).

Os experimentos realizados neste trabalho provaram a relevância da análise das relações espaciais hierárquicas entre as características da retina para a separabilidade das classes, utilizando menos amostras de treinamento do que outras abordagens, e sem a necessidade de segmentar a região do disco óptico. Ademais, este estudo apresenta contribuições tanto no campo computacional quanto na área da saúde, em virtude da exploração de técnicas de *Deep Learning* e processamento de imagens no contexto de imagens médicas. Os resultados alcançados neste trabalho demonstram o potencial da metodologia para auxiliar o especialista na tomada de decisão sobre o diagnóstico dos exames, contribuindo para um diagnóstico precoce do paciente, consequentemente, evitando a perda da visão.

Os resultados mostraram-se promissores, não obstante, ainda há espaço para ajustes no método, a fim de alcançar uma classificação ainda melhor. Como trabalhos futuros, pretende-se: i) adicionar mais convoluções à arquitetura original para destacar regiões importantes das imagens antes de entrar nas cápsulas; e, ii) modificar o número de cápsulas no modelo para avaliar o impacto das mesmas na classificação dos dados.

Índices COVID: Uma Abordagem Baseada em Textura para Classificar Lesões Pulmonares

Autores: Antônio Oseas de Carvalho Filho; Patrick Ryan Sales dos Santos.

Resumo: A COVID-19 é uma doença infecciosa causada por um tipo recém descoberto de coronavírus, denominado SARS-CoV-2. Desde a descoberta da doença, no final de 2019, a COVID-19 tornou-se uma preocupação mundial, principalmente pelo seu elevado grau de contágio. Em outubro de 2020, o número de casos confirmados de COVID-19 reportados à *World Health Organization* já ultrapassava 36 milhões no mundo, enquanto o número de mortes ultrapassa 1 milhão. Em virtude dos impactos causados pela doença, a literatura tem intensificado seus esforços no estudo de abordagens voltadas à detecção de COVID-19, visando auxiliar e facilitar o processo de diagnóstico da doença. Este trabalho propõe a aplicação de descritores de textura baseados nas relações filogenéticas entre as espécies para a caracterização de volumes segmentados de exames de *Computed Tomography* (CT) e posterior classificação das regiões em COVID-19, sólidas ou não nódulos. Para avaliar o método proposto, foram utilizadas imagens provindas de três conjuntos de dados distintos. Os resultados mostraram-se promissores, com uma acurácia de 99,25%, *recall* de 99,25%, *precision* de 99,24%, F1 de 99,24% e AUC de 0,96. O trabalho apresenta um método robusto, simples e eficiente, que pode ser facilmente aplicado em imagens 2D e/ou 3D, sem limitações em relação à dimensionalidade das imagens.

Keywords: COVID-19; Tomografia Computadorizada; Análise de textura 3D; Diversidade Filogenética.

12.1 Introdução

Desde a descoberta do novo coronavírus (COVID-19) na China, no final de 2019, a doença tornou-se uma preocupação mundial, principalmente pela sua rápida disseminação. Em março de 2020, 80,565 casos do novo vírus foram confirmados na China e 95,333 no mundo ([SHAN et al., 2020](#)). Em janeiro de 2021, o número de casos confirmados reportados à *World Health Organization* (WHO) já ultrapassava 86 milhões, enquanto o número de mortes ultrapassa 1,8 milhão ([WHO, 2020](#)). A COVID-19 é uma doença infecciosa causada por um tipo de vírus descoberto recentemente, denominado SARS-CoV-2. A maioria das pessoas infectadas com esse vírus tem doença respiratória leve a moderada e se recupera sem tratamento especial. No entanto, idosos e pessoas com doenças pré-existentes, como doenças cardiovasculares, diabetes, doenças

respiratórias crônicas e câncer têm maior probabilidade de desenvolver formas graves de COVID-19 (WHO, 2020; GORBALENYA et al., 2020; REMUZZI; REMUZZI, 2020). Devido à indisponibilidade de medicamentos específicos para COVID-19, o diagnóstico precoce é fundamental para a cura e controle da doença (SINGH et al., 2020).

Os testes de laboratório aplicados na detecção do COVID-19 atualmente baseiam-se, principalmente, na *reverse transcription polymerase chain reaction* (RT-PCR) (JAISWAL et al., 2020). Contudo, diante do cenário de pandemia, é comum a escassez desses testes em alguns lugares, o que pode atrasar o diagnóstico e tratamento clínico e, consequentemente, favorecer a propagação da doença. Além disso, esse tipo de exame pode produzir uma alta taxa de falsos negativos (SHAN et al., 2020). Como alternativa aos testes laboratoriais, outros tipos de exames têm sido utilizados pelos hospitais para o diagnóstico da doença, como por exemplo a *Computed Tomography* (CT) (SHAN et al., 2020).

A CT proporciona insumos radiológicos claros do paciente acometido pela COVID-19, portanto é considerada eficiente para a avaliação dos casos suspeitos da doença (BERNHEIM et al., 2020b). Contudo, o diagnóstico através da CT depende da análise das imagens pelo especialista, sendo este um processo demorado, repetitivo e propenso a erros, uma vez que analisar muitas imagens pode ser exaustivo, principalmente no cenário pandêmico, onde há vários exames a serem diagnosticados (FILHO et al., 2018). Normalmente, os casos de COVID-19 compartilham algumas características nas imagens tomográficas, como a presença de *ground-glass opacities* (GGO) em estágios iniciais e *pulmonary consolidation* (PC) em estágios avançados (WANG et al., 2020b). O *pleural effusion* (PE) também pode ocorrer em casos de COVID-19, mas é menos comum do que as demais lesões. No entanto, embora essas características sejam frequentes na maior parte dos casos, as imagens de algumas pneumonias virais também apresentam tais características, o que pode acabar confundindo o diagnóstico, ocultando outras doenças mais graves (WANG et al., 2020b).

Devido à importância do diagnóstico precoce da COVID-19, várias técnicas computacionais vêm sendo desenvolvidas para serem integradas aos sistemas de detecção auxiliado por computador (CAD, do inglês *Computer-Aided Detection*) e diagnóstico auxiliado por computador (CAD-X do inglês *Computer-Aided Diagnosis*). O objetivo desses sistemas é melhorar a precisão do diagnóstico e auxiliar os especialistas em tomada de decisão, dando aos pacientes maiores chances de receber tratamento antes que a doença progride.

Neste trabalho, é proposto a aplicação de um método eficiente para extrair características importantes dos volumes de CT, sendo essa etapa denominada em um sistema de visão computacional de, extração de características. Como as lesões provocadas pela COVID-19 não apresentam uma forma definida, a textura pode ser a propriedade mais relevante para fornecer informações da doença que sejam eficientes para o processo de classificação. Um dos motivos da despadronização na forma das lesões é que na segmentação manual, os próprios especialistas costumam delimitar uma região maior que o volume de interesse (VOI), do inglês *Volume of Interest*, a fim de extrair mais detalhes que possam contribuir para o diagnóstico da doença. Por outro lado, muitos sistemas CAD aplicam métodos de segmentação que produzem regiões candidatas com formatos semelhantes (FILHO et al., 2014; NETTO et al., 2012), podendo acarretar uma classificação errônea. Em virtude de tais fatores, foi empregado apenas a análise por textura na solução.

Para medir essa textura, foram utilizados conceitos da biologia referentes aos índices de diversidade filogenética, aplicados para descrever as relações filogenéticas entre as espécies, a fim de distinguir as VOIs entre COVID-19, sólidas e tecido saudável. Reunimos VOIs de lesões provindas de três conjuntos de dados diferentes, a fim de formar um conjunto heterogêneo que possa ser utilizado para avaliar a robustez e confiabilidade da nossa solução.

A seguir, destaca-se as principais contribuições deste trabalho:

- O presente método leva em consideração as VOIs das CTs, sem a necessidade de dividi-las em slices;
- O método trabalha com as VOIs em sua dimensão original, dispensando a aplicação de quaisquer técnicas de redimensionamento;
- Utiliza-se técnicas de outras áreas do conhecimento, adaptando-as ao contexto das imagens pulmonares. Especificamente, é proposto a caracterização da textura das imagens com base nas relações filogenéticas entre as espécies;

- Desenvolve-se um método escalável, uma vez que pode ser facilmente empregado em imagens 2D ou 3D, sem restrições em relação à quantização das imagens;
- Emprega-se uma avaliação do método em conjuntos de dados distintos, realizando um processo de classificação multiclasse para a distinção das lesões provocadas pela COVID-19 e por outros tipos de doenças.

12.1.1 Objetivos

Diante do contexto apresentado, o propósito deste trabalho é explorar e analisar a textura na sua viabilidade da utilização das características filogenéticas, então assim classificar os volumes de tecidos pulmonares, entre solida, COVID-19 e não nódulo em imagens de CT, provindas de conjuntos de imagens distintos.

De maneira específica, pretende-se:

1. Avaliar a viabilidade do uso dos índices para a identificação da doença, utilizando bases públicas de imagens de CT, com e sem balanceamento das classes;
2. Construir uma metodologia para classificação automática do covid-19, que possa ser aplicada no auxílio ao diagnóstico da doença.

12.1.2 Organização do capítulo

Este trabalho está organizado da seguinte forma: a Seção 12.2 explana sobre os conceitos fundamentais para o entendimento da metodologia proposta neste trabalho; a Seção 12.4 aborda os trabalhos da literatura relacionados à detecção de COVID-19; a Seção 12.5 discorre sobre o método proposto; a Seção 12.6 aponta os resultados alcançados com a execução da metodologia; a Seção 12.7 discute os resultados obtidos; e, finalmente, a Seção 12.8 mostra a conclusão deste trabalho.

12.2 Referencial Teórico

Este capítulo apresenta os principais conceitos para a compreensão da proposta deste trabalho. A Seção 12.2.1 relata sobre o COVID-19, abordando as formas atuais de diagnóstico da doença. A Seção 12.2.2 discorre sobre a extração de características por textura, onde é detalhando sobre como é possível extrair informações de uma imagens analisando a textura da mesma. A Subseção 12.2.2.1 Descreve os índices de diversidade filogenética, extraídos dos exames de CTs, a Subseção 12.2.3 exemplifica de forma simples a utilização e os cálculos realizados pelos índices de diversidade filogenética, a Seção 12.2.4 discorre como a classificação foi realizada para as classes solida, COVID-19 e não nódulo, e por fim a Seção 12.3 descreve como a validação dos resultados foi obtida.

12.2.1 COVID-19

Em dezembro de 2019, a COVID-19 foi descoberto na cidade de Wuhan, China. Esse surto se espalhou exponencialmente por todo o mundo, sendo declarada como uma pandemia. Os mais prevalentes dos sintomas clínicos de pacientes com COVID-19 são febre, seguida por tosse, fadiga e dispneia. Essa doença pode levar a problemas respiratórios agudos, síndrome de angústia, insuficiência renal aguda, choque e morte. (ZHAO et al., 2020b).

Os critérios de diagnósticos da COVID-19 são avaliação laboratorial de secreções respiratórias adquiridas do aspirado endotraqueal, lavagem broncoalveolar ou swab nasofaríngeo / orofaríngeo. Atualmente, exames laboratoriais, como a RT-PCR tornou-se a avaliação padrão para o diagnóstico de infecção por COVID-19. No entanto, os resultados do teste RT-PCR podem ser falsamente negativos devido a insuficiência amostra ou erro de laboratório (OZKAYA; OZTURK; BARSTUGAN, 2020b).

A COVID-19 é uma doença inflamatória causada por quadro agudo grave de síndrome respiratória, que pode se manifestar como um amplo espectro de sintomas que variam de poucos ou nenhum sintoma para

pneumonia grave que pode evoluir para aguda síndrome de dificuldade respiratória e morte (MILLET et al., 2021). Embora os mecanismos moleculares que conduzem a gravidade da doença permaneçam obscuros, a associação clínica de mediadores com casos graves sugerem que a inflamação excessiva é fundamental para um resultado clínico ruim (TAN et al., 2021).

A indução de processos inflamatórios na célula hospedeira muitas vezes requer o envolvimento de inflamassomas, que são plataformas de proteínas que se agregam no citosol em resposta a diferentes estímulos (RODRIGUES et al., 2021).

12.2.2 Extração de Características por Textura

Uma das tarefas mais complexas na análise de imagens está na definição de um conjunto de características que possam descrever de maneira concreta cada região contida em uma imagem, de modo a serem utilizados em processos de mais alto nível (PEDRINI; SCHWARTZ, 2008b). Em outras palavras, a etapa de caracterização e representação da imagem consiste em uma etapa de fundamental importância no modelo clássico do processamento de imagem digital proposto por (CAREY et al., 2006). Nesta etapa, as propriedades da imagem podem ser representadas através de modelos matemáticos para servirem de entrada para a análise de reconhecimento e classificação de padrões.

Em linhas gerais, a etapa de extração de características pode ser dividida em duas categorias de análises, sendo elas por: textura e forma. Em uma análise por textura, o objetivo geral é descrever aspectos da imagem no que diz respeito a suavidade, rugosidade e regularidade (CAREY et al., 2006). Já em uma análise baseada na forma da imagem, o intuito é extrair informações que mensuram sobre propriedades morfológicas da imagem.

Segundo (HARALICK; SHANMUGAM; DINSTEIN, 1973), textura é definida como a informação de uma região relacionada a coeficientes de uniformidade, densidade, aspereza, regularidade, intensidade, dentre outras características da imagem. A análise de textura é relevante em imagens digitais, uma vez que possibilita distinguir regiões da imagem que apresentam características semelhantes (AZEVEDO; CONCI; VASCONCELOS, 2018).

12.2.2.1 Índices de diversidade filogenética

Nesta Seção, apresenta-se a justificativa para os índices propostos para caracterização de textura. Cada índice corresponde a uma característica, totalizando oito características que serão extraídas de cada imagem analisada.

A filogenética é um ramo da biologia que estuda as relações evolutivas entre as espécies, e é possível descrever a semelhança entre elas. Em árvores filogenéticas, as folhas representam espécies e os nós representam ancestrais comuns. A figura 87 ilustra um exemplo de árvore filogenética, representando a relação genética entre a espécie de macaco e a espécie humana, onde é possível observar que geneticamente, homem e chimpanzé estão mais próximos que os outros pares existentes na árvore (BAXEVANIS; OUELLETTE, 2004)

A combinação de árvores filogenéticas e índices de diversidade filogenética é usada para analisar as relações evolutivas entre as espécies e pode medir a variação das espécies presentes em uma comunidade (CIANCIARUSO; SILVA; BATALHA, 2009a). Para aplicar esses conceitos à caracterização de imagens de tomografia computadorizada, é necessário compreender a correspondência entre as definições da biologia e as utilizadas neste trabalho. A tabela 79 descreve esta correspondência.

Neste trabalho, é apresentado os índices de Diversidade Filogenética (PD), do inglês *Phylogenetic Diversity* (FAITH, 1992), Soma das Distâncias Filogenéticas (SPD), do inglês *Sum of Phylogenetic Distances* (HELMUS et al., 2007), Distância Média do Vizinho Mais Próximo (MNND), do inglês *Mean Nearest Neighbor Distance* (WEBB; LOSOS, 2000), Variabilidade Filogenética das Espécies (PSV), do inglês *Phylogenetic Species Variability* (HELMUS et al., 2007), Riqueza de Espécies Filogenéticas (PSR), do inglês *Phylogenetic Species Richness* (HELMUS et al., 2007), Distância Filogenética Média (MPD), do inglês *Mean Phylogenetic Distance* (WEBB; LOSOS, 2000), índice de diversidade taxonômica (Δ) (CLARKE; WARWICK, 1998b) e índice de distinção

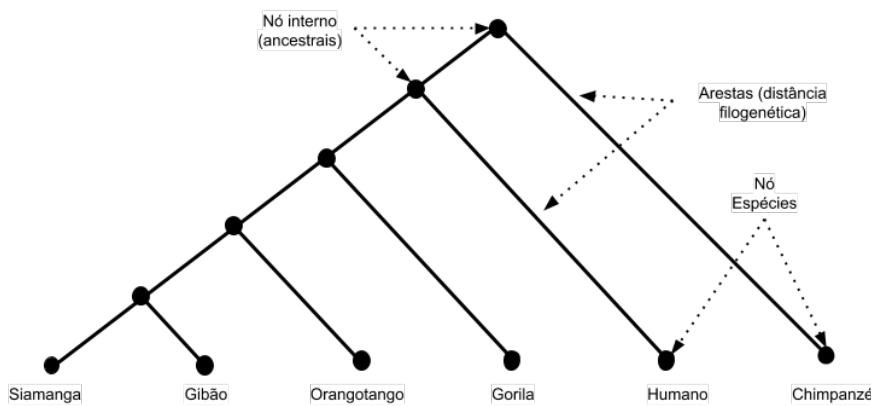


Figura 87 – Exemplo de árvore filogenética para alguns primatas adaptados de (BAXEVANIS; OUELLETTE, 2004).

Tabela 79 – Correspondência entre os termos da biologia e o método proposto.

Biologia	Método
Comunidade	Volume segmentado
Espécies	Valor de voxel
Indivíduos	Número de voxels em uma espécie
Características fenotípicas das espécies	Relação de textura entre espécies

taxonômica (Δ^*) (CLARKE; WARWICK, 1998b). Cada índice corresponde a um descritor. Portanto, para cada imagem analisada, teremos oito feições compondo o vetor de feições.

O índice PD (FAITH, 1992) é uma medida responsável por fornecer a soma das distâncias dos ramos filogenéticos da árvore. Assim, quando o comprimento do ramo é longo, a espécie possivelmente corresponde a grupos mais distintos. A equação (12.1) apresenta a fórmula para o cálculo do PD, onde B identifica o número de ramos da árvore, L_i representa a extensão do ramo i (o número de bordas no ramo i), e A_i refere-se à abundância média de espécies que compartilham o ramo i .

$$PD = B \times \frac{\sum_i^B L_i A_i}{\sum_i^B A_i}. \quad (12.1)$$

O SPD é um índice filogenético responsável por calcular a soma das distâncias entre os pares de espécies presentes na árvore (HELMUS et al., 2007). A equação (12.2) pode calcular este índice, onde S representa o número de espécies na comunidade, d_{nm} representa a distância entre as espécies m e n , e a_m e a_n correspondem à abundância das espécies m e n , respectivamente.

$$SPD = \left(\frac{S(S - 1)}{2} \right) * \frac{\sum \sum_{m < n} d_{mn} a_m a_n}{\sum \sum_{m < n} a_m a_n}. \quad (12.2)$$

O MNND calcula a média ponderada da distância filogenética do vizinho mais próximo de cada espécie (WEBB; LOSOS, 2000). Os pesos são a abundância de espécies. A equação (12.3) apresenta a fórmula para o cálculo deste índice, onde S representa o número de espécies na comunidade, d_{nm} representa a distância entre as espécies m e n , e a_m corresponde à abundância de espécies de m . No caso de d_{nm} , n se refere ao parente mais próximo da espécie m .

$$MNND = \sum_m^S \min(d_{mn}) a_m, \quad (12.3)$$

O índice PSV mede a variação entre duas espécies em uma comunidade a fim de quantificar a relação filogenética entre elas. O PSV pode ser calculado usando a Equação (12.4), onde C é uma matriz e trC é a soma dos valores diagonais desta matriz, $\sum c$ representa a soma de todos os valores da matriz, n é o número total de espécies e \bar{c} é a média dos valores fora da diagonal da matriz.

$$PSV = \frac{n(trC) - \sum c}{n(n - 1)} = 1 - \bar{c}, \quad (12.4)$$

O índice PSR calcula a riqueza de espécies presentes em uma comunidade com base em sua variabilidade (HELMUS et al., 2007). Portanto, conforme mostrado pela Equação (12.5), o cálculo é feito multiplicando o número de espécies (n) pelo PSV.

$$PSR = nPSV. \quad (12.5)$$

O MPD calcula a média das distâncias filogenéticas analisando a combinação de todos os pares de espécies na comunidade (WEBB; LOSOS, 2000). Para isso, utiliza o número total de espécies, indicado por N , a distância filogenética entre cada par de espécies, denotada por d_{ij} , e também uma variável $p_i p_j$, que terá valor 1 se a espécie está presente e 0 se não estiver presente. A equação (12.6) apresenta a fórmula para o cálculo do MPD.

$$MPD = \frac{\sum_{i=1}^{N-1} \sum_{j=i+1}^N d_{ij} p_i p_j}{\sum_{i=1}^{N-1} \sum_{j=i+1}^N p_i p_j}, \quad (12.6)$$

O valor de Δ fornece a distância filogenética média entre indivíduos de espécies (CLARKE; WARWICK, 1998b). Para isso, esse índice leva em consideração o número de indivíduos da espécie e a relação taxonômica entre eles. A fórmula para calcular Δ é definida pela Equação (12.7), onde x_i ($i = 1, \dots, S$) representa a abundância das espécies i , x_j ($j = 1, \dots, S$) representa a abundância das espécies j , S indica o número total de espécies, n denota o número total de indivíduos, e w_{ij} expressa a distância taxonômica entre as espécies i e j .

$$\Delta = \frac{\sum \sum_{i < j} w_{ij} x_i x_j}{[n(n - 1)/2]}, \quad (12.7)$$

Finalmente, o índice Δ^* , definido pela Equação (12.8), expressa a distância taxonômica média entre dois indivíduos de espécies distintas (CLARKE; WARWICK, 1998b). Neste cálculo, x_i ($i = 1, \dots, S$) é a abundância das espécies i , x_j ($j = 1, \dots, S$) é a abundância das espécies j , S indica o número total de espécies, n indica o número total de indivíduos e w_{ij} expressa a distância taxonômica entre as espécies i e j .

$$\Delta^* = \frac{\sum \sum_{i < j} w_{ij} x_i x_j}{\sum \sum_{i < j} x_i x_j}, \quad (12.8)$$

12.2.3 Exemplo de cálculo de índice para uma imagem

As equações descritas na Subseção 12.2.2.1 são apresentadas em relação à biologia. Para facilitar a compreensão do cálculo dos índices nas imagens, apresentamos nesta subseção um exemplo de imagem bidimensional, da qual foi extraída a árvore filogenética e calculada as distâncias e os oito índices. Foi usado uma pequena imagem para que os cálculos não ficassem extensos no papel. A figura 88 mostra a imagem que foi usada como exemplo, seguida da árvore filogenética extraída. É possível ver que a imagem possui diversidade de espécies em relação aos pixels. É importante ressaltar que a imagem ilustrada na Figura 88 é apenas um exemplo e, portanto, não consideramos os verdadeiros valores de pixel das cores utilizadas.

Para calcular as distâncias filogenéticas com base na árvore filogenética, consideramos as seguintes equações:

$$D_{ij} = j - i + 1, \text{ se } i = 0, \text{ e}$$

$$D_{ij} = j - i + 2, \text{ se } i \neq 0,$$

onde i e j são duas espécies diferentes.

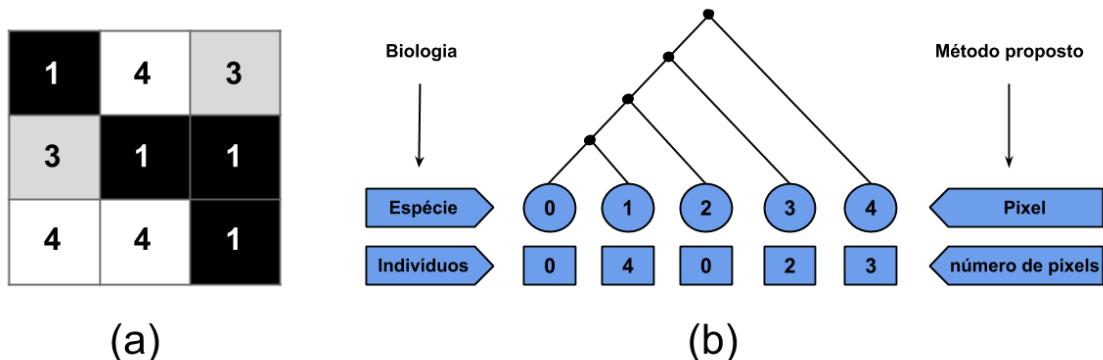


Figura 88 – (a) Exemplo de uma região analisada, (b) Representação da árvore filogenética extraída do exemplo. Autoria própria.

Tabela 80 – Distâncias calculadas para a árvore filogenética mostrada na Figura 88.

i	j	D_{ij}
0	1	$D_{01} = 1 - 0 + 1 = 2$
0	2	$D_{02} = 2 - 0 + 1 = 3$
0	3	$D_{03} = 3 - 0 + 1 = 4$
0	4	$D_{04} = 4 - 0 + 1 = 5$
1	2	$D_{12} = 2 - 1 + 2 = 3$
1	3	$D_{13} = 3 - 1 + 2 = 4$
1	4	$D_{14} = 4 - 1 + 2 = 5$
2	3	$D_{23} = 3 - 2 + 2 = 3$
2	4	$D_{24} = 4 - 2 + 2 = 4$
3	4	$D_{34} = 4 - 3 + 2 = 3$

Com base nas equações acima, observa-se na Tabela 80 as distâncias obtidas para a árvore filogenética da Figura 88.

Nesta implementação dos índices, apresenta-se a árvore filogenética para uma estrutura de histograma. Assim, cada posição do histograma se refere às espécies, que são as intensidades da imagem, e cada valor se refere à abundância, que é o número de *pixels* de cada intensidade. Assim, é possível calcular as distâncias usando o histograma. Com base na imagem, na árvore filogenética e nas distâncias calculadas, calcula-se os índices descritos na Subseção 12.2.2.1.

O PD definido na Equação (12.1) considera o número de ramos da árvore (B), a extensão de cada ramo (L_i) e a média da abundância de espécies em cada ramo (A_i). Em relação às imagens, B equivale ao número de espécies menos um, L_i representa a distância entre as espécies do ramo i e A_i refere-se à abundância média das espécies conectadas ao ramo i pelo número de espécies também conectadas a esse ramo. Os valores de L_i são os mesmos da Tabela 80, enquanto a abundância das espécies pode ser vista na Figura 88. Como o cálculo do PD é mais extenso que os outros, foi descrito usando a Tabela 81.

Com base nos valores calculados na Tabela 81, o valor PD para o exemplo de imagem resultou em:

Tabela 81 – Cálculo dos termos L_i e A_i do índice PD.

i	j	L_i	A_i	$L_i * A_i$
0	1	2	$(0 + 4) / 2 = 2$	$2 * 2 = 4$
0	2	3	$(0 + 4 + 0) / 3 = 1,33$	$3 * 1,33 = 3,99$
0	3	4	$(0 + 4 + 0 + 2) / 4 = 1,5$	$4 * 1,5 = 6$
0	4	5	$(0 + 4 + 0 + 2 + 3) / 5 = 1,8$	$5 * 1,8 = 9$
1	2	3	$(4 + 0) / 2 = 2$	$3 * 2 = 6$
1	3	4	$(4 + 0 + 2) / 3 = 2$	$4 * 2 = 8$
1	4	5	$(4 + 0 + 2 + 3) / 4 = 2,25$	$5 * 2,25 = 11,25$
2	3	3	$(0 + 2) / 2 = 1$	$3 * 1 = 3$
2	4	4	$(0 + 2 + 3) / 3 = 1,66$	$4 * 1,66 = 6,64$
3	4	3	$(2 + 3) / 2 = 2,5$	$3 * 2,5 = 7,5$
Sum			18,04	65,38

$$PD = B * \frac{\sum_i^B L_i A_i}{\sum_i^B A_i} = 4 * \frac{65,38}{18,04} = 14,49$$

Para calcular o SPD, representado pela Equação (12.2), precisamos dos valores de: distância entre espécies (d_{mn}); número de espécies (S), que nesse caso é o número de intensidades do histograma; e abundância de espécies (a), que representa o número de pixels da intensidade em análise. Substituindo os valores em cada equação, tem-se:

$$\begin{aligned} \Sigma \Sigma_{m < n} d_{mn} a_m a_n &= (2 * 0 * 4) + (3 * 0 * 0) + (4 * 0 * 2) + (5 * 0 * 3) + (3 * 4 * 0) + \\ &(4 * 4 * 2) + (5 * 4 * 3) + (3 * 0 * 2) + (4 * 0 * 3) + (3 * 2 * 3) = 110 \end{aligned}$$

$$\begin{aligned} \Sigma \Sigma_{m < n} a_m a_n &= (0 * 4) + (0 * 0) + (0 * 2) + (0 * 3) + (4 * 0) + (4 * 2) + (4 * 3) + \\ &(0 * 2) + (0 * 3) + (2 * 3) = 26 \end{aligned}$$

$$\left(\frac{S * (S - 1)}{2} \right) = \left(\frac{5 * (5 - 1)}{2} \right) = 10$$

$$SPD = 10 * \frac{110}{26} = 42,30$$

O MNND, representado pela Equação (12.3), requer apenas a distância de uma espécie ao parente mais próximo. Portanto, S indica o número de espécies (número de intensidades no histograma), m denota a espécie em questão (intensidade m), n representa o parente mais próximo de m (intensidade n , que se refere à intensidade após m), e a_m denota a abundância da espécie m (número de pixels com intensidade m). Como nossa árvore filogenética tem apenas um caminho de uma espécie para outra, o caminho mínimo é o único caminho entre as espécies. Assim, para a imagem, o MNND resulta no seguinte valor:

$$MNND = \sum_m^S \min(d_{mn}) a_m = (2 * 0) + (3 * 4) + (3 * 0) + (3 * 2) = 18,00$$

Para calcular o PSV, representado pela Equação (12.4), é considerado as operações aplicadas à imagem, que é a matriz C . Assim, trC se refere à soma da imagem diagonal, $\sum c$ representa a soma de todos os *pixels* da imagem e \bar{c} se refere à média dos *pixels* fora da diagonal. Além disso, n representa o número de espécies da comunidade que possuem indivíduos, ou seja, a quantidade de intensidades que estão presentes na imagem.

$$\begin{aligned} PSV &= \frac{n(trC) - \sum c}{n(n-1)} = \frac{3*3 - 22}{3*(3-1)} = \frac{-13}{6} = -2,16 \\ 1 - \bar{c} &= 1 - 3,16 = -2,16 \end{aligned}$$

O PSR, representado pela Equação (12.5), é apenas o PSV multiplicado pelo número de espécies da comunidade que possuem indivíduos, ou seja, a quantidade de intensidades de imagem.

$$PSR = n * PSV = 3 * (-2,16) = -6,48$$

Em relação ao MPD, definido pela Equação (12.6), considera-se a soma das distâncias entre as espécies (d_{ij}) multiplicada pelas variáveis p_i e p_j , que recebem o valor 0 se a espécie não estiver presente ou 1 se a espécie estiver presente. Assim, quando existe uma intensidade do histograma na imagem, p recebe 1, caso contrário p recebe 0.

$$\begin{aligned} \sum_{i=1}^{N-1} \sum_{j=i+1}^N d_{ij} p_i p_j &= (2 * 0 * 1) + (3 * 0 * 0) + (4 * 0 * 1) + (5 * 0 * 1) + (3 * 1 * 0) + \\ &(4 * 1 * 1) + (5 * 1 * 1) + (3 * 0 * 1) + (4 * 0 * 1) + (3 * 1 * 1) = 12 \end{aligned}$$

$$\begin{aligned} \sum_{i=1}^{N-1} \sum_{j=i+1}^N p_i p_j &= (0 * 1) + (0 * 0) + (0 * 1) + (0 * 1) + (1 * 0) + (1 * 1) + \\ &(0 * 1) + (0 * 1) + (1 * 1) = 3 \end{aligned}$$

$$MPD = \frac{12}{3} = 4.00$$

No cálculo de Δ , definido pela Equação (12.7), w_{ij} representa a distância entre as espécies (intensidades) i e j , x refere-se ao abundância de espécies (número de *pixels* nas intensidades), e n é o número de indivíduos na comunidade (número de *pixels* no histograma). Dessa forma, podemos calcular Δ da seguinte maneira:

$$\begin{aligned} \Sigma \Sigma_{i < j} w_{ij} x_i x_j &= (2 * 0 * 4) + (3 * 0 * 0) + (4 * 0 * 2) + (5 * 0 * 3) + (3 * 4 * 0) + \\ &(4 * 4 * 2) + (5 * 4 * 3) + (3 * 0 * 2) + (4 * 0 * 3) + (3 * 2 * 3) = 110 \end{aligned}$$

$$[n * (n - 1)/2] = [9 * (9 - 1)/2] = 36$$

$$\Delta = \frac{110}{36} = 3,05$$

Finalmente, o cálculo de Δ^* , definido pela Equação (12.8), é semelhante ao de *triangulo*, com a diferença que no caso de Δ^* o denominador é a soma da multiplicação das abundâncias das espécies. Substituindo os valores, encontra-se:

$$\begin{aligned} \Sigma \Sigma_{i < j} w_{ij} x_i x_j &= (2 * 0 * 4) + (3 * 0 * 0) + (4 * 0 * 2) + (5 * 0 * 3) + (3 * 4 * 0) + \\ &(4 * 4 * 2) + (5 * 4 * 3) + (3 * 0 * 2) + (4 * 0 * 3) + (3 * 2 * 3) = 110 \end{aligned}$$

$$\begin{aligned}\Sigma \Sigma_{i < j} x_i x_j &= (0 * 4) + (0 * 0) + (0 * 2) + (0 * 3) + (4 * 0) + (4 * 2) + (4 * 3) + \\ &(0 * 2) + (0 * 3) + (2 * 3) = 26\end{aligned}$$

$$\Delta^* = \frac{110}{26} = 4,23$$

12.2.4 Classificação

Para o reconhecimento do padrão das características extraídas, foram utilizados dois classificadores comuns na literatura: RF ([WRIGHT; WAGER; PROBST, 2016](#)) e reforço de gradiente extremo (XGBoost) ([CHEN et al., 2015](#)). Para validar os modelos, foi aplicado a técnica de validação cruzada. Essa técnica envolve a divisão aleatória do conjunto de imagens em k folds, de tamanho aproximadamente igual. A primeira fold é tratada como um conjunto de validação, e o método é treinado nas outras $k - 1$ folds. É importante ressaltar que cada imagem na amostra de dados é atribuída a um grupo individual e permanece nesse grupo durante o procedimento. Isso significa que cada amostra tem a oportunidade de ser usada no conjunto de validação 1 vez e usada para treinar o modelo $k - 1$ vezes. Cada fold tem uma média de 10% do número de imagens em cada classe. A seguir está uma breve descrição dos classificadores usados.

- RF é um algoritmo de regressão e classificação desenvolvido por [Wright, Wager e Probst \(2016\)](#). Nesse método, as previsões são feitas a partir de árvores de decisão. Cada árvore no RF fornece uma previsão de classe, então a classe com mais votos torna-se a saída do modelo para a amostra em questão.
- XGBoost é uma técnica de aprendizado de máquina otimizada desenvolvida por ([CHEN et al., 2015](#)), que é baseada em árvores de decisão e usa uma estrutura gradiente crescente. Este algoritmo foi projetado para ser flexível e eficiente, por isso seus parâmetros podem ser facilmente alterados ([CARVALHO et al., 2020](#)). O XGBoost pode ser aplicado a problemas de regressão e classificação.

12.3 Métricas de Validação

Para avaliar os modelos de classificação obtidos na etapa anterior, foram utilizadas as seguintes métricas: acurácia, *recall*, precisão, pontuação F1 e AUC. Para calcular essas métricas, é necessário analisar a matriz de confusão, que é calculada com base em quatro valores: verdadeiro positivo (TP), falso positivo (FP), falso negativo (FN) e verdadeiro negativo (TN). Esses valores servem para indicar o número de amostras classificadas corretamente e incorretamente.

A acurácia (Acc) calcula a capacidade de um teste de diagnóstico de identificar tanto os verdadeiros positivos quanto os verdadeiros negativos em uma amostra, ou seja, a proporção de acertos nos dados classificados. A equação (12.9) mostra a fórmula desta métrica.

$$Acc = \frac{TN + TP}{TN + TP + FN + FP}. \quad (12.9)$$

Em um teste diagnóstico, a *recall* (rec), também conhecida como sensibilidade, mede a capacidade de identificar verdadeiros positivos, ou seja, indivíduos que têm a doença. É uma métrica fundamental na análise, pois indica a eficácia do método. A equação (12.10) apresenta a fórmula para o cálculo de *recall*.

$$Rec = \frac{TP}{TP + FN}. \quad (12.10)$$

A precisão (prec), expressa pela Equação (12.11), mede a proporção de positivos corretamente classificados entre todos aqueles classificados como positivos.

$$Prec = \frac{TP}{TP + FP}. \quad (12.11)$$

É possível combinar precisão e *recall* para medir o desempenho da classificação, e é isso que pontuação F1 (F1) faz. Conforme visto na Equação (12.12), esta métrica calcula uma média harmônica de precisão e *recall*, sendo uma medida comumente usada em problemas de classificação desequilibrada.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \quad (12.12)$$

Também foi usado a análise da área sob a curva ROC (AUC) para avaliar o método. A curva ROC (*receiver operating feature*) (ERKEL; PATTYNAMA, 1998) fornece um gráfico de desempenho de um método de classificação a partir de diferentes limiares, analisando duas métricas calculadas por meio da matriz de confusão: a taxa de verdadeiro positivo (TPR) e a taxa de falso positivo (FPR). A partir deste gráfico, a AUC pode medir quão bem o método de classificação foi capaz de distinguir os dados; portanto, quanto mais próximo de 1 for o resultado, melhor será o método de separação de classes.

12.4 Trabalhos Relacionados

Desde a declaração da pandemia do COVID-19, pela WHO, os esforços no campo científico têm sido intensificados em prol do desenvolvimento de metodologias para auxiliar o diagnóstico da doença, baseando-se em imagens médicas, como exames de raio-X e tomografias computadorizadas. É destacado, nesta Seção, alguns trabalhos relevantes sobre o tema.

Os trabalho de Abbas, Abdelsamea e Gaber (2020) e Narin, Kaya e Pamuk (2020), apresentaram metodologias baseadas em imagens de raio-X e Deep Learning para a classificação de imagens em COVID-19 e non-COVID-19. Os resultados alcançados foram promissores, com acurácia acima de 90%. Devido à escassez na disponibilidade de dados de CT públicos da COVID-19, o trabalho de Zhao et al. (2020b) e He et al. (2020b) construíram seus próprios conjunto de dados, chamado COVID-CT, composto por 349 imagens de CT de COVID-19 e 463 de non-COVID-19. Já no estudo de Ozkaya, Ozturk e Barstugan (2020b), dois subconjuntos foram extraídos de 150 imagens de CT, sendo que cada subconjunto contém 3000 patches de COVID-19 e 3000 patches de non-COVID-19. No estudo de Wang et al. (2020b) também aplicou imagens de CT e abordagens de aprendizado profundo para a tarefa de classificação das imagens em casos positivos e negativos para a COVID-19.

No trabalho de Barstugan, Ozkaya e Ozturk (2020), foram utilizadas 150 imagens de CT, divididas em 4 subconjuntos, que abrangem patches de dimensões 16x16, 32x32, 48x48, 64x64, respectivamente. A abordagem empregada no trabalho de Barstugan, Ozkaya e Ozturk (2020), aplica técnicas como Grey Level Co-occurrence Matrix (GLCM), Grey Level Run Length Matrix (GLRLM), Grey-Level Size Zone Matrix (GLSZM), Local Directional Pattern (LDP), e Discrete Wavelet Transform (DWT) para extrair características das imagens e, em seguida, classificá-las em COVID-19 ou non-COVID-19 utilizando o Support-Vector Machine (SVM). Na abordagem proposta por Jaiswal et al. (2020), um modelo pré-treinado da arquitetura DenseNet201 é aplicado para realizar a classificação de imagens de CT em COVID-19 ou non-COVID-19, aplicando os pesos da ImageNet.

Diferente das abordagens anteriores, o trabalho de Zheng et al. (2020) propõe uma metodologia para a detecção de COVID-19 utilizando volumes 3D de CTs. Nessa abordagem, cada volume é segmentado através de uma UNet pré-treinada e posteriormente classificado por meio de uma rede neural profunda 3D *weakly-supervised*, denominada DeCoVNet. As CTs utilizadas nos experimentos foram coletadas de 540 pacientes (313 com COVID-19 e 229 sem COVID-19).

A Tabela 82 aponta um resumo dos trabalhos descritos.

Como observado no decorrer desta seção, desde o surto da COVID-19, várias abordagens vêm sendo exploradas para a detecção da doença através da análise de imagens médicas. Nos estudos correlatos, é possível observar a crescente aplicação de métodos de *deep learning* na detecção de lesões provocadas pela COVID-19, inclusive, as Convolutional Neural Network (CNN) são exploradas em diversos domínios de imagem, sendo o estado da arte no reconhecimento de imagens (??).

No entanto, alguns fatores tornam as CNNs um método complexo e muitas vezes limitado a um problema específico. Um desses fatores é a parametrização, visto que as CNNs requerem uma ampla quantidade de parâmetros, que precisam ser ajustados durante o treinamento até que o modelo se torne adequado

Tabela 82 – Resumo do trabalho relacionado.

Trabalho	Método	Tipo de exame	Amostra	Resultados
Abbas, Abdelsamea e Gaber (2020)	Decompose, Transfer, and Compose (DeTraC)	X-ray	206	Acurácia: 95,12% Sensibilidade: 97,91% Especificidade: 91,87%
Narin, Kaya e Pamuk (2020)	Pre-trained CNNs	X-ray	100	Acurácia: 98,00% Recall: 96,00% Especificidade: 100% Precisão: 100% F1: 98,00%
Zhao et al. (2020b)	Multi-task learning and contrastive self-supervised learning	CT	812	Acurácia: 89,00% F1: 90,00% AUC: 0,98
Ozkaya, Ozturk e Barstugan (2020b)	Pre-trained CNNs + SVM	CT	150	Acurácia: 98,27% Sensibilidade: 98,93% Especificidade: 97,60% Precisão: 97,63% F1: 98,28%
Wang et al. (2020b)	Selection of ROIs + Inception	CT	453	Acurácia: 82,90% Sensibilidade: 84,00% Especificidade: 80,50%
He et al. (2020b)	Self-supervised transfer learning	CT	746	Acurácia: 86,00% F1: 85,00% AUC: 0,94
Barstugan, Ozkaya e Ozturk (2020)	GLCM + GLRLM + GLSZM + LDP + DWT + SVM	CT	150	Acurácia: 98,71% Sensibilidade: 97,56% Especificidade: 99,68% Precisão: 99,62% F1: 98,58%
Jaiswal et al. (2020)	Pre-trained DenseNet201	CT	2492	Acurácia: 96,25% Recall: 96,29% Especificidade: 96,21% Precisão: 96,29% F1: 96,29%
Zheng et al. (2020)	3D UNet + 3D DeCoVNet	CT	540	Sensibilidade: 90,70% Especificidade: 91,10% AUC: 0,959

ao problema a que se propõe (RAKHLIN et al., 2018; GOLATKAR; ANAND; SETHI, 2018; NAWAZ et al., 2018). Para contornar esse problema, alguns estudos aplicam a transferência de aprendizado, que permite a utilização de pesos pré-definidos; entretanto, essa abordagem pode não proporcionar um resultado válido em todos os casos, uma vez que esses pesos, sendo treinados para outra finalidade, podem não ser adequados para o problema em questão.

Outro fator que limita o uso de CNNs é o número de amostras necessárias no treinamento para melhorar o desempenho da classificação. Inclusive, por este motivo os estudos Jaiswal et al. (2020), Zheng et al. (2020), Abbas, Abdelsamea e Gaber (2020), Zhao et al. (2020b), He et al. (2020b) aplicam técnicas de aumento de dados. Além disso, dependendo da arquitetura, é necessário modificar o tamanho das imagens para adaptá-las à entrada da rede, como empregado nos estudos Narin, Kaya e Pamuk (2020), Zhao et al. (2020b), He et al. (2020b). Contudo, o redimensionamento das imagens pode ocasionar a perda de informações valiosas para a classificação.

Dos trabalhos mencionados anteriormente, apenas Zheng et al. (2020) utilizou volumes de CT em seu método. Em contrapartida, apresentamos uma abordagem simples e eficiente para a detecção de COVID-19 em imagens de CT, utilizando volumes 3D das lesões. A extração de características com os índices de diversidade filogenética não requer uma parametrização complexa, não se limita à quantidade de bits das imagens, não necessita de padronização no tamanho das entradas e é escalável, já que é possível ser aplicada em imagens 2D ou 3D. Além disso, em união aos classificadores Random Forest (RF) e XGBoost, os índices mostraram-se promissores mesmo sem a aplicação de técnicas de aumento de dados.

12.5 Metodologia

Esta seção descreve a metodologia proposta para a classificação de volumes de CT em COVID-19, sólidas ou não nódulos. As imagens utilizadas neste estudo foram adquiridas utilizando a *Lung Image Database*

Consortium Image Collection (LIDC-IDRI) (III et al., 2011c) e do repositório *MedSeg* (MEDSEG, 2020), sendo que, deste último, utilizamos dois conjuntos com imagens de COVID-19. Na etapa de extração de características, foram aplicados os índices de diversidade filogenética para mensurar o comportamento da textura das imagens. Por fim, na etapa de classificação, as características extraídas serviram de entrada para dois classificadores, que foram testados e avaliados. Os algoritmos de extração e classificação desenvolvidos neste estudo estão disponíveis publicamente no repositório do [GitHub](#). A Figura 89 descreve o fluxo de trabalho da metodologia.

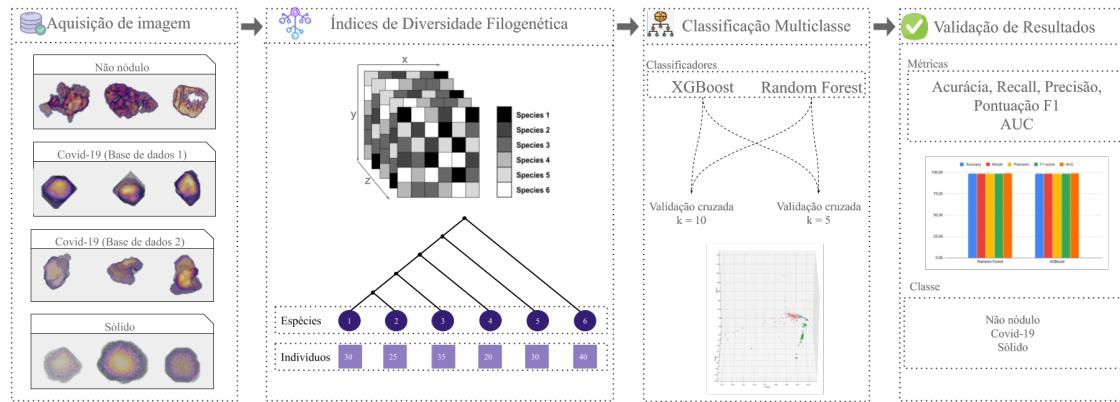


Figura 89 – Método proposto. Autoria própria.

12.5.1 Aquisição de imagens

Para avaliar o método, utilizou-se três conjuntos de VOIs extraídos da base *LIDC-IDRI* e do repositório *MedSeg*. i) no primeiro conjunto são extraídas as VOIs que apresentam lesões sólidas e regiões de não nodulares (regiões saudáveis). No caso das lesões sólidas, é utilizado a biblioteca *pyLIDC* (HANCOCK, 2016) para gerar as VOIs dos exames com base nas anotações dos especialistas; ii) no segundo conjunto de imagens são extraídas as VOIs de não nódulos, isso é, imagens saudáveis, as regiões foram geradas aplicando o método proposto por Filho et al. (2014b), que garante que as VOIs de não nódulos não possuem intersecção com as VOIs nodulares; iii) o terceiro conjunto de imagens foram adquiridos através do repositório *MedSeg* (MEDSEG, 2020), que fornece alguns conjuntos de dados externos de vários tipos de exames de CT, inclusive exames diagnosticados com a COVID-19. Assim, utiliza-se dois conjuntos de dados distintos com lesões provocadas pela COVID-19, isso é, podem haver regiões com lesões GGO, PC e PE.

Indicamos na Tabela 83 a quantidade de imagens em cada conjunto utilizado.

Tabela 83 – Distribuição de imagens entre conjuntos de dados.

Conjunto de dados	Diagnóstico	Sample
LIDC	Sólidas	1679
	Não nódulos	17742
COVID-19 (Dataset 1)	GGO, PC e PE	215
COVID-19 (Dataset 2)	GGO, PC e PE	274

12.5.2 Extração de características

Como mencionado na Seção 12.2, foi aplicada uma análise por textura para extrair as características das imagens por meio dos índices PD, SPD, MNND, PSV, PSR, MPD, (Δ) e (Δ^*). Os índices foram implementados na linguagem de programação *python*. A extração de características ocorreu para todos os exemplos, gerando uma representação da árvore filogenética das espécies, que pode ser abstruída para o histograma da imagem e por fim, a extração dos índices ocorre, embora a complexidade das imagens seja de forma 3D a construção do histograma mantém-se independente das dimensões da imagem. É importante ressaltar que a representação de bits que a imagem é construída influencia diretamente no tamanho do histograma, como as imagens analisadas pelo presente trabalho pertencem a ordem de 16 bits, o processamento e extração percorrem por todos os valores, sem perda de informações. Um outro ponto relevante, está na transladação dos valores negativos presentes na imagem, onde foi realizado uma técnica que move os valores negativos para positivos, acarretando com que o menor valor negativo assuma o valor zero e assim sucessivamente.

12.5.3 Classificação

Como mencionado na Seção 12.2, foram aplicados os classificadores RF e XGBoost. A escolha desses classificadores, está na sua larga utilização pela literatura, sendo ferramentas robustas já validadas pela academia. Outro ponto pelo qual foi escolhido esses classificadores, está no fato de que o RF é um algoritmo baseado em árvores de decisão, e o XGBoost em *machine learning*, assim tendo uma análise e testificação mais abrangente das características extraídas por intermédio dos índices filogenéticos.

Os classificadores, foram utilizados na linguagem de programação *python*, com auxílio da biblioteca *sklearn*, que por sua vez facilita a utilização dessas ferramentas, envolvendo-os e abstraindo sua complexidade.

A Tabela 84, apresenta os principais parâmetros utilizados nos modelos.

Tabela 84 – Parâmetros principais usados nos classificadores.

Classificador	Parâmetros Usados
RF	$number\ of\ estimators\ (number\ of\ trees) = 100,$ $min\ samples\ split = 2,$ $min\ samples\ leaf = 1,$ $max\ number\ of\ features = "auto" (\sqrt{number\ features})$ $bootstrap = True,$ $max\ depth = None\ (unlimited)$ $max\ depth = 6,$ $learning\ rate = 0.1,$
XGBoost	$number\ of\ estimators\ (number\ of\ trees) = 100,$ $booster = "gbtree",$ $objective = "binary:logistic",$ $gamma = 0,$ $max\ delta\ step = 0$

12.5.4 Validação

Como mencionado na Seção 12.2, foram utilizadas as métricas Acc, Rec, Prec, F1 e AUC para avaliar os resultados da classificação. As métricas escolhidas para realizar a validação dos resultados obtidos pelo

presente trabalho, são comumente utilizadas pela literatura, por conseguirem demonstrar de forma clara os erros e acertos obtidos pelos classificadores.

Por se tratar de uma classificação multiclasse, aplica-se as métricas descritas na Seção 12.2 usando a estratégia *one-vs.-rest*, ou seja, considerando uma classe em relação às demais. Considere X como uma das classes no conjunto de imagens. Em outras palavras, para a validação de X , consideramos TP como amostras positivas de X e TN como amostras positivas de outras classes. Além disso, FP são todas as amostras classificadas como X , mas não são X , e FN são todas as amostras de X que foram previstas como outras classes. Esse processo é realizado na validação de cada amostra. Por fim, para solucionar o desequilíbrio do conjunto de imagens, é calculada uma média ponderada das métricas, sendo os pesos o número de amostras em cada classe.

12.6 Resultados

Esta seção, apresenta os resultados dos testes realizados nos conjuntos de dados descritas na Seção 12.5.1. As características das VOIs foram extraídas a partir de descritores de textura, com base na diversidade filogenética, conforme especificado na Seção 12.5.2. Para a classificação das regiões em saudáveis, sólidas ou COVID-19, foram utilizados os classificadores apresentados na Seção 12.5.3. Para garantir que todas as amostras sejam treinadas e testadas pelo menos uma vez, foi utilizado o método *k-fold*, onde aplica-se $k = 5$ e $k = 10$, optou por variar o valor de k para garantir uma quantidade maior de amostras nos conjuntos de testes.

Esta seção divide-se em três partes: i) apresenta-se os resultados alcançados pelo método; ii) realiza-se uma exploração dos resultados obtidos através da análise de alguns casos de acerto e erro do método; e por fim, iii) compara-se os resultados do método proposto com os resultados dos estudos relacionados.

12.6.1 Resultados obtidos pelo método proposto

Foi dividido os testes em três experimentos, apresentados na Tabela 85. Cada experimento contempla combinações de classificação com as bases utilizadas, sendo que COVID-19 (1) e COVID-19 (2) referem-se às duas bases de VOIs de lesões provocadas pela COVID-19, fornecidas pela *MedSeg*. O intuito de executar o método em diferentes cenários de teste é avaliar a potencialidade das características diante de imagens provindas de bases distintas. Os três cenários fornecem desafios importantes para avaliação do método proposto, no entanto, considera-se como melhor resultado os dispostos no cenário 1, pois, acredita-se que este se aproxima mais de um ambiente clínico real, uma vez que as lesões diagnosticadas pelos especialistas possuem origens distintas.

Outro ponto importante é que, como visto na Tabela 83, os conjuntos de dados utilizados neste trabalho são desbalanceados, o que pode induzir à interpretações errôneas dos resultados em virtude da discrepância entre a proporção de dados em cada classe. Por este motivo, cada experimento foi realizado duas vezes: uma com os dados desbalanceados (maior aproximação com cenário clínico real) e outra com os dados balanceados. Em relação ao balanceamento, é aplicado apenas um *downsample* nos dados, baseando-se na quantidade de dados da base minoritária. Como o foco do nosso trabalho não está em técnicas de balanceamento, não implementamos abordagens robustas para balancear os dados, apenas uma técnica simples para que fosse possível analisar o desempenho do método nas duas situações.

A Tabela 86 aponta os resultados alcançados através dos experimentos realizados com 10 *folds* no *cross validation*, enquanto a Tabela 87 mostra os resultados com 5 *folds*. A referência de cada experimento está descrito na Tabela 85.

Tabela 85 – Experimentos realizados no trabalho.

Experimentos	Descrição/Classes	Amostra
1	não nódulos × sólidas × COVID-19 (1) e COVID-19 (2)	19,868
2	não nódulos × sólidas × COVID-19 (1)	19,624
3	não nódulos × sólidas × COVID-19 (2)	19,653

Tabela 86 – Resultados usando k-fold, com k = 10.

Cenário de teste	Classificador	Experimento	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC
Sem balanceamento	RF	1	99,09 ± 0,14	99,09 ± 0,14	99,09 ± 0,14	99,06 ± 0,16	0,95 ± 0,01
		2	99,32 ± 0,12	99,32 ± 0,12	99,29 ± 0,14	99,27 ± 0,15	0,92 ± 0,01
		3	99,68 ± 0,08	99,68 ± 0,08	99,68 ± 0,08	99,66 ± 0,09	0,96 ± 0,01
	XGBoost	1	99,25 ± 0,15	99,25 ± 0,15	99,24 ± 0,16	99,23 ± 0,16	0,96 ± 0,00
		2	99,33 ± 0,17	99,33 ± 0,17	99,30 ± 0,20	99,30 ± 0,19	0,93 ± 0,01
		3	99,81 ± 0,06	99,81 ± 0,06	99,81 ± 0,06	99,80 ± 0,06	0,98 ± 0,00
Com balanceamento	RF	1	92,23 ± 1,89	92,23 ± 1,89	92,30 ± 1,92	92,22 ± 1,88	0,94 ± 0,01
		2	91,31 ± 3,42	91,31 ± 3,42	91,66 ± 3,40	91,29 ± 3,45	0,93 ± 0,02
		3	94,94 ± 2,53	94,94 ± 2,53	95,05 ± 2,50	94,95 ± 2,52	0,96 ± 0,01
	XGBoost	1	92,81 ± 1,74	92,81 ± 1,74	92,87 ± 1,72	92,80 ± 1,74	0,95 ± 0,01
		2	92,25 ± 3,99	92,25 ± 3,99	92,59 ± 3,89	92,24 ± 4,00	0,94 ± 0,02
		3	96,31 ± 2,12	96,31 ± 2,12	96,42 ± 2,11	96,32 ± 2,11	0,97 ± 0,01

-Negrito é o melhor resultado considerando o cenário 1.

Tabela 87 – Resultados usando k-fold, com k = 5.

Cenário de teste	Classificador	Experimento	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC
Sem balanceamento	RF	1	99,09 ± 0,11	99,09 ± 0,11	99,08 ± 0,11	99,05 ± 0,11	0,95 ± 0,00
		2	99,32 ± 0,07	99,32 ± 0,07	99,29 ± 0,09	99,26 ± 0,08	0,92 ± 0,00
		3	99,69 ± 0,04	99,69 ± 0,04	99,68 ± 0,04	99,68 ± 0,04	0,96 ± 0,00
	XGBoost	1	99,25 ± 0,15	99,25 ± 0,15	99,24 ± 0,16	99,23 ± 0,16	0,96 ± 0,00
		2	99,33 ± 0,17	99,33 ± 0,17	99,30 ± 0,20	99,30 ± 0,19	0,93 ± 0,01
		3	99,81 ± 0,06	99,81 ± 0,06	99,81 ± 0,06	99,80 ± 0,06	0,98 ± 0,00
Com balanceamento	RF	1	92,30 ± 0,73	92,30 ± 0,73	92,36 ± 0,68	92,31 ± 0,72	0,94 ± 0,00
		2	90,85 ± 1,33	90,85 ± 1,33	91,14 ± 1,32	90,84 ± 1,32	0,93 ± 0,00
		3	95,50 ± 1,25	95,50 ± 1,25	95,63 ± 1,20	95,50 ± 1,25	0,97 ± 0,00
	XGBoost	1	92,09 ± 1,77	92,09 ± 1,77	92,13 ± 1,73	92,08 ± 1,78	0,94 ± 0,01
		2	91,01 ± 2,62	91,01 ± 2,62	91,25 ± 2,74	90,97 ± 2,62	0,93 ± 0,01
		3	95,63 ± 1,32	95,63 ± 1,32	95,67 ± 1,32	95,63 ± 1,32	0,97 ± 0,00

-Negrito é o melhor resultado considerando o cenário 1.

Analizando os resultados das Tabelas 86 e 87, percebe-se que o XGBoost apresentou os melhores resultados em praticamente todos os experimentos realizados, principalmente nos cenários de teste onde a execução do método encontra-se sem o balanceamento. Possivelmente, os resultados de classificação da classe majoritária influenciaram nos resultados gerais, por este motivo optou-se por avaliar a potencialidade do método diante de dados balanceados. Nesse caso, é possível perceber que mesmo com o balanceamento os resultados continuaram promissores, atingindo acurácia de 96,31%, recall de 96,31%, precision de 96,42%, F1 de 96,32% e AUC de 0,97 no experimento 3 usando k=10. No caso do cenário ideal, que seria o experimento 1, o método apresentou os melhores resultados através do classificador XGBoost. Sem balanceamento, os testes desse cenário resultaram em acurácia de 99,25%, recall de 99,25%, precision de 99,24%, F1 de 99,24% e AUC de 0,96. Com balanceamento, o método alcançou acurácia de 92,81%, recall de 92,81%, precision de 92,87%, F1 de 92,80% e AUC de 0,95.

Em relação à variação nos experimentos realizados, nosso intuito foi avaliar a robustez do método diante de diferentes tipos de bases. Os testes mostraram que mesmo com heterogeneidade das bases, os resultados foram consistentes e promissores em ambos os classificadores. Um outro ponto importante a ser discutido é que o experimento 3 apresentou os melhores resultados em ambos os cenários de teste e classificadores. Acredita-se que isso deva-se ao fato de que a base 2 de COVID-19 apresenta uma similaridade maior entre as lesões, o que pode ter contribuído para uma melhor distinção das características de cada classe por parte dos classificadores. Pelos resultados, é possível perceber que a base 1 de COVID-19 não contempla este mesmo padrão de similaridade, o que pode confundir o classificador em algumas amostras.

12.6.1.1 Estudo de caso

Nesta Subseção, foi realizada uma exploração dos resultados obtidos através da análise de alguns casos de acerto e erro do método. Para essa análise, foi escolhida o melhor resultado do cenário 1 com balanceamento, onde alcançou-se acurácia de 92,81%, recall de 92,81%, precision de 92,87%, F1 de 92,80% e AUC de 0,95.

Optou-se por mostrar apenas o resultado com o balanceamento para proporcionar uma visualização mais clara dos resultados, sem tantas amostras.

A primeira demonstração é o *plot* das características extraídas com os índices propostos, onde foi aplicado o algoritmo *Principal Component Analysis* (PCA) para reduzir a dimensionalidade dos dados preservando as informações mais importantes. O PCA resultante, representado pela Figura 90, possui três componentes. É possível observar que a classe não nodular, que teve maior assertividade em relação aos resultados obtidos, ficou espacialmente mais distante das demais classes, o que facilitou o trabalho dos classificadores. Em contrapartida, as classes solida e COVID-19 tiveram a maior parte das amostras espacialmente próximas, mas é notável a possibilidade de traçar uma fronteira de decisão entre elas.

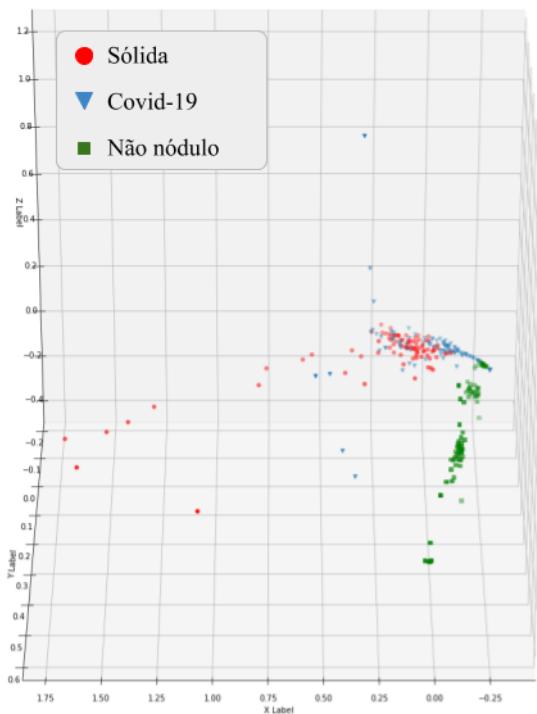


Figura 90 – Representação do PCA com três componentes para as características extraídas no caso de teste desta seção. Autoria própria.

Na próxima análise, foi selecionada aleatoriamente uma amostra de cada item da matriz de confusão (Figura 91) para o caso de teste abordado nesta seção. A Figura 92 apresenta as imagens selecionadas em forma de matriz de confusão. Os tracejados vermelhos representam as amostras onde o classificador errou, enquanto os verdes representam os acertos. É possível notar que, visivelmente, a lesão da classe não nodular apresenta a textura mais distinta entre as demais, comprovando assim o PCA mostrado na análise anterior. Da mesma forma, percebe-se que, visualmente, a Figura 92(b) possui uma similaridade de textura com a Figura 92(d), justificando assim a classificação errônea da classe solida com a de COVID-19. Já em relação à lesão da Figura 92(c), nota-se que ela apresenta uma textura mais próxima da classe solida do que as demais. Por fim, em relação à lesão Figura 92(e), acredita-se que a textura presente se distanciou do padrão encontrado nas lesões de COVID-19 e solida, por esse motivo o classificador identificou-a como uma classe não nodular.

Também observa-se o histograma das imagens selecionadas na análise anterior para fornecer uma visualização da distribuição dos voxels das imagens, ilustrada na Figura 93. Além disso, a fim de apresentar uma visualização das características dessas imagens, foi criada a Figura 94. Sobre a Figura 94, é possível notar que o gráfico da Figura 94(e) foge do padrão visto nos outros gráficos da Figura 94, sendo ela a única instância classificada erroneamente como não nodular. Em relação aos gráficos das Figuras 94(d) e 94(b), é possível notar que eles possuem similaridade em relação aos valores de índice, o que pode ter motivado a

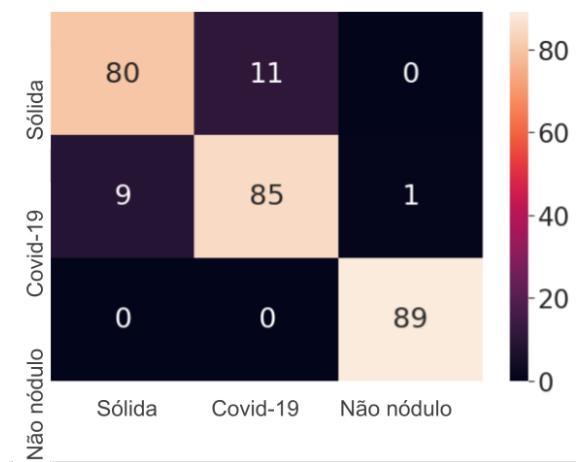


Figura 91 – Resultado da matriz de confusão. Autoria própria.

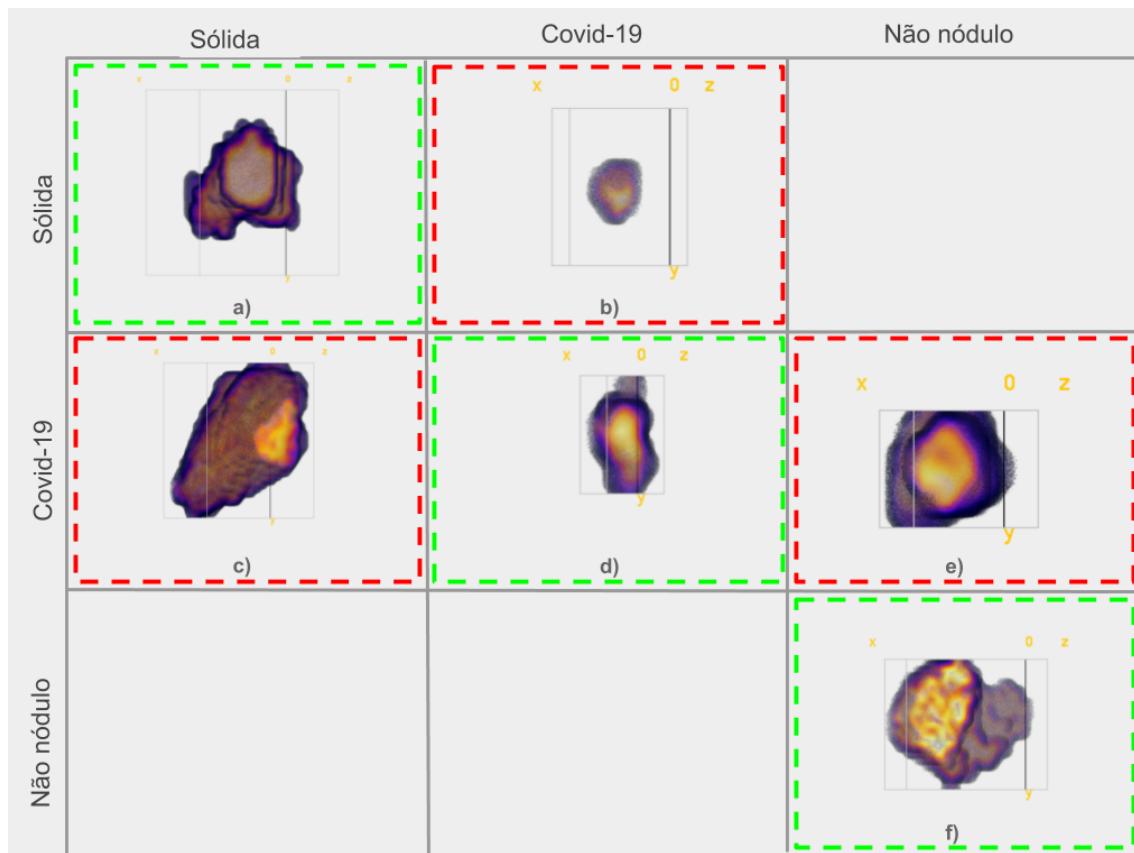


Figura 92 – Representação de uma matriz de confusão com casos de acerto e erro do classificador. Autoria própria.

classificação errônea como COVID-19, sendo ela uma lesão solida. O mesmo acontece no caso dos gráficos das Figuras 94(a) e 94(c), onde há uma similaridade entre os valores, consequentemente, dificultando o processo de classificação

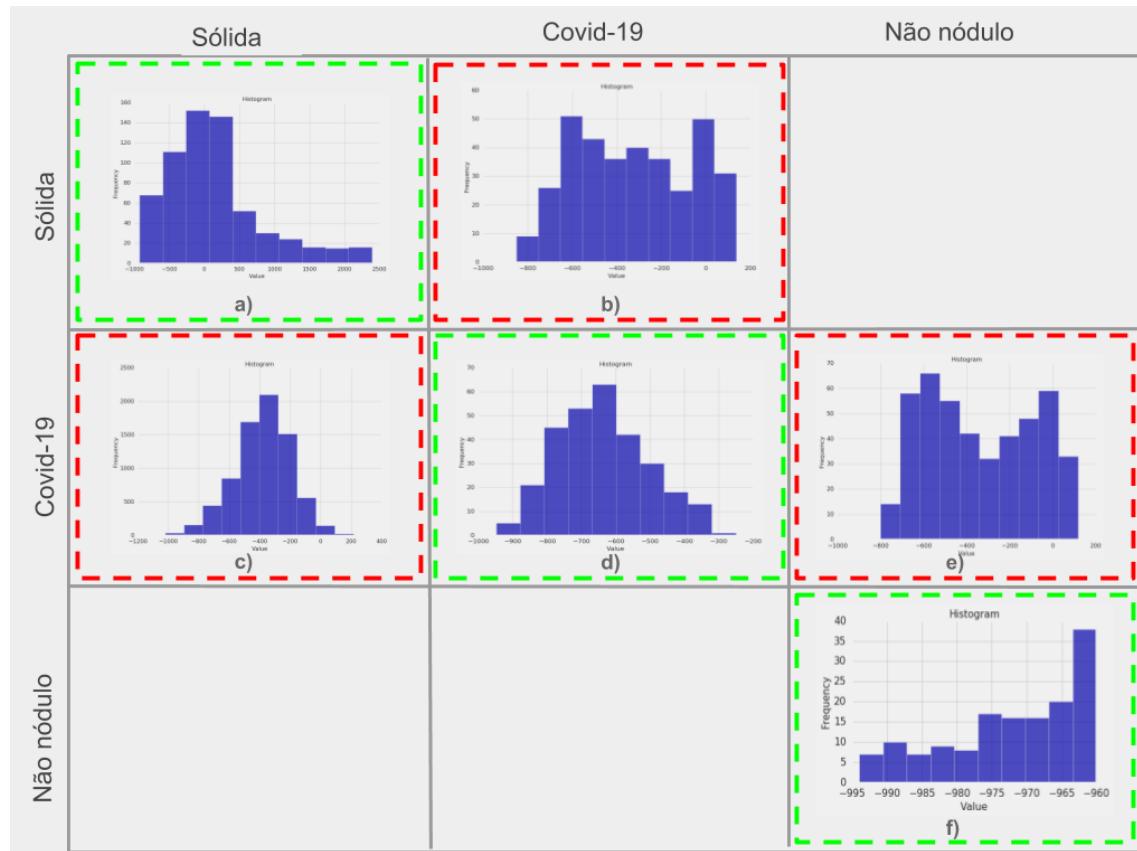


Figura 93 – Histograma das imagens selecionadas da matriz de confusão. Autoria própria.

12.6.1.2 Comparação com trabalhos relacionados

Nesta Subseção, foi realizado uma comparação dos resultados obtidos neste trabalho com os estudos apresentados na Seção 12.4. O objetivo dessa comparação é apenas relacionar os resultados alcançados pelo método proposto com o estado da arte, sem prejudicar ou diminuir nenhum dos métodos mencionados. A Tabela 88 aponta esta comparação. Em relação aos resultados do método proposto, destaca-se na tabela apenas os resultados do classificador XGBoost no cenário 1, pelo fato de que esse é o cenário mais próximo de um ambiente real, já que contempla imagens de diferentes fontes, aumentando a confiabilidade da solução.

Observa-se, através da Tabela 88, que, de modo geral, o trabalho proposto apresenta resultados promissores tão quanto os demais estudos abordados. Inclusive, alcançamos acurácia, recall e F1 superiores aos demais. Vale ressaltar que apenas a nossa abordagem aplica uma classificação multiclasse, o que por si só, já torna a solução mais complexa. Além disso, foi utilizado o maior conjunto de imagens entre todos os estudos relacionados.

O ideal para um sistema CAD é atingir um equilíbrio entre as métricas de validação, pois deve diagnosticar corretamente indivíduos saudáveis e doentes. Diante dos resultados apresentados, observa-se que o x método se mostrou promissor na classificação das VOIs em todos os cenários de teste, enfatizando a relevância dos índices filogenéticos para a descrição das imagens de CT.

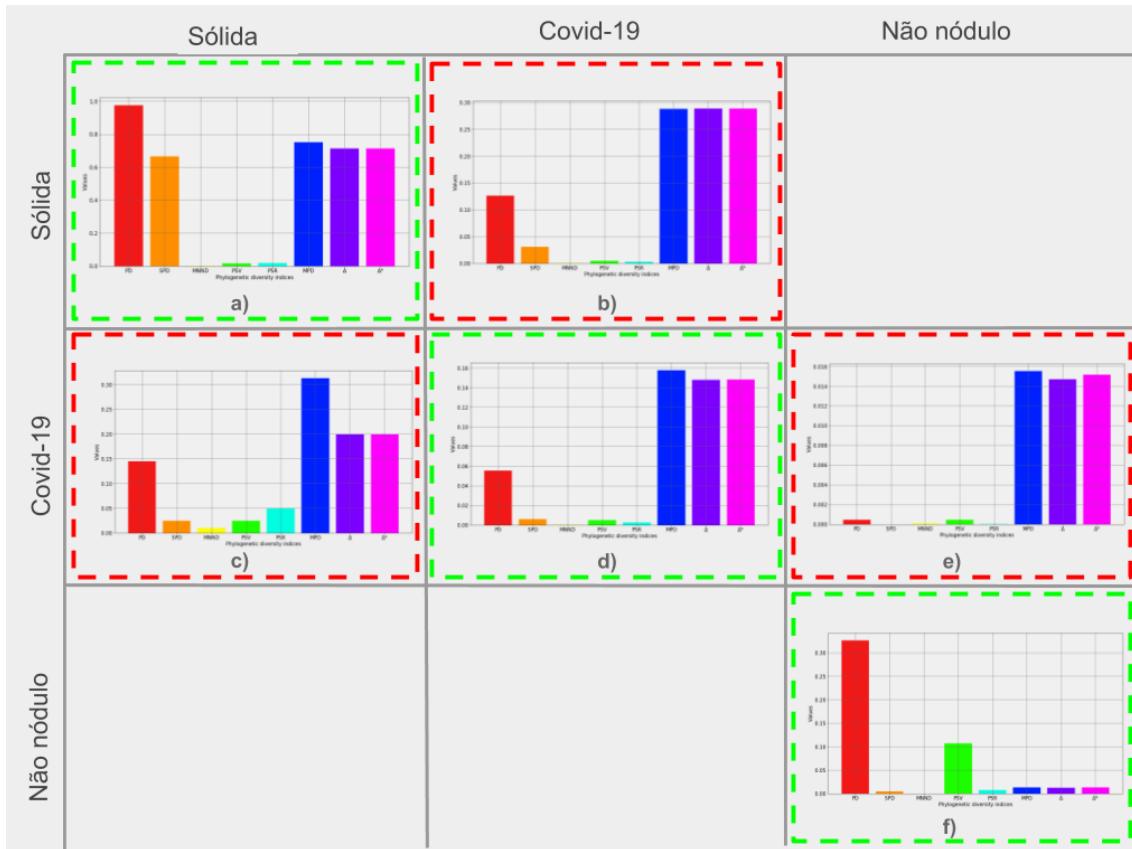


Figura 94 – Representação dos índices extraídos das imagens selecionadas da matriz de confusão. Autoria própria.

12.7 Discussões

Conforme apresentado no decorrer deste trabalho, a classificação de COVID-19 não é uma tarefa trivial, principalmente por se tratar de um problema recente, onde ainda não foi amplamente abordado pela literatura. Uma das principais dificuldades na exploração de abordagens voltadas à esse problema é a escassez de dados de exames.

Nos estudos relacionados, é possível notar os esforços na classificação da COVID-19 usando diferentes abordagens, sendo a maioria baseadas em *deep learning*. No entanto, embora os modelos de CNN tenham se mostrado abordagens cada vez mais capazes de representar imagens de forma eficiente, por meio de mapas de características e camadas profundas, essa eficácia depende de vários fatores que muitas vezes tornam o uso das CNNs mais complexo do que outros métodos aplicados ao mesmo problema, por exemplo , i) o problema do balanceamento de classes para obter um bom aprendizado nos modelos; ii) parametrização adequada ao problema; iii) exigir bom hardware dependendo do problema; iv) ampla quantidade de dados para identificar os padrões corretos; e, finalmente, v) cada arquitetura requer uma dimensão diferente das imagens de entrada.

Em relação aos trabalhos apresentados na Seção 12.4, apenas (ZHENG et al., 2020) aplicou volumes de CT em seu método. Já a abordagem proposta neste trabalho apresenta um método simples e eficiente para a detecção de COVID-19 em exames de CT, utilizando volumes 3D das lesões. Os índices de diversidade filogenética não necessitam de uma parametrização complexa, também não se limitam à quantidade de bits das imagens, não requerem padronização no tamanho das entradas e são escaláveis, sendo possível aplicá-los a imagens 2D e 3D. Além disso, em união aos classificadores utilizados nos experimentos, os índices mostraram-se promissores mesmo sem a aplicação de técnicas de aumento ou pré-processamento

Tabela 88 – Comparação com trabalhos relacionados.

Trabalho	Tipo de exame	Objetivo	Acc (%)	Rec (%)	Prec (%)	F1 (%)	AUC
(ABBAS; ABDELSAMEA; GABER, 2020)	X-ray	COVID-19 e Não COVID-19	95,12	97,91	-	-	-
(NARIN; KAYA; PAMUK, 2020)	X-ray	COVID-19 e Não COVID-19	98,00	96,00	100	98,00	-
(ZHAO et al., 2020b)	CT	COVID-19 e Não COVID-19	89,00	-	-	90,00	0,98,00
(OZKAYA; OZTURK; BARSTUGAN, 2020b)	CT	COVID-19 e Não COVID-19	98,27	98,93	97,63	98,28	-
(WANG et al., 2020b)	CT	COVID-19 e Não COVID-19	82,9	84,00	-	-	-
(HE et al., 2020b)	CT	COVID-19 e Não COVID-19	86,00	-	-	85,00	0,94
(BARSTUGAN; OZKAYA; OZTURK, 2020)	CT	COVID-19 e Não COVID-19	98,71	97,56	99,62	98,58	-
(JAISWAL et al., 2020)	CT	COVID-19 e Não COVID-19	96,25	96,29	96,29	96,29	-
(ZHENG et al., 2020)	CT	COVID-19 e Não COVID-19	-	90,7	-	-	0,95
Método proposto	Sem Balanceamento	COVID-19, Solido e Não nódulo	99,25	99,25	99,24	99,24	0,96
	Com Balanceamento	COVID-19, Solido e Não nódulo	92,81	92,81	92,87	92,80	0,95

dos dados.

A seguir, alguns pontos relevantes sobre o trabalho:

- O método proposto foi testado em dois cenários de classificação: um deles usando o *cross validation* com $k=10$ e outro com $k=5$. O intuito em variar o parâmetro k era avaliar como o método se comporta quando é treinado com quantidades de dados diferentes;
- Foram realizados três experimentos combinando as bases utilizadas neste trabalho, visando avaliar o método com amostras de imagens provindas de diferentes fontes. Cada um dos experimentos foi executado com as amostras de dados balanceadas e não balanceadas, proporcionando uma visão mais crítica sobre como o método se comporta diante de conjuntos desbalanceados ou não;
- Os resultados alcançados mostraram-se promissores e consistentes em todos os cenários de teste e classificadores utilizados, motivando a aplicação dos descritores em ambientes reais;
- Os índices caracterizam as imagens em suas dimensões originais, sem a necessidade de um redimensionamento, preservando a integridade das características das regiões;
- Foram utilizados os oito índices para a análise de textura pelo fato de que essa combinação pode caracterizar melhor a lesão, uma vez que cada índice mede separadamente alguma propriedade que o outro não é capaz de medir. Dessa forma, os índices combinados podem fornecer aos classificadores uma descrição mais robusta das imagens.

12.8 Conclusão

Este trabalho propôs um método capaz de descrever e classificar regiões de CT em três classes: COVID-19, sólidas e não nódulos. Para tanto, foram utilizadas características de textura baseadas nos índices de diversidade filogenética. A abordagem alcançou acurácia de 99.25% e AUC de 0.96 nos dados desbalanceados, enquanto que nos dados balanceados obteve acurácia de 92.81% e AUC de 0.95. Acredita-se que os resultados promissores se devam ao fato de os índices filogenéticos podem potencializar as características de textura das VOIs, uma vez que cada um deles contribui para que um atributo específico da região seja adicionado aos demais para se chegar a uma distinção satisfatória das classes.

Assim, acredita-se que o método apresentado neste trabalho seja capaz de integrar uma ferramenta CAD, podendo ser aplicada em casos reais no auxílio ao diagnóstico de COVID-19. Como abordado durante o trabalho, a detecção de COVID-19 vem sendo um desafio para os profissionais da saúde, visto que é um problema recente que, apesar de não apresentar uma alta taxa de letalidade, precisa ser diagnosticado em seus estágios iniciais para evitar a progressão da doença e o contágio para outras pessoas. Sendo assim, a abordagem traz benefícios tanto para o especialista, que pode contar com uma segunda opinião durante o diagnóstico, quanto para o paciente, visto que a detecção precoce é fundamental para que o paciente alcance o tratamento em tempo hábil, aumentando as chances de cura.

Diagnóstico Automático da Presença de Exsudatos em Imagens de Retina Utilizando Aprendizado Profundo

Autores: Antônio Oseas de Carvalho Filho; Deusimar Damião de Sousa.

Resumo: Atualmente, o Diabetes é uma das doenças crônicas que mais cresce no mundo e é capaz de desencadear uma série de complicações na saúde dos portadores. A Retinopatia Diabética, é uma complicação do Diabetes que afeta a visão e, se não for tratada imediatamente, pode levar à cegueira total do paciente. Essa anormalidade não tem cura, mas se descoberta nos seus estágios iniciais, o paciente tem grandes chances de não atingir a cegueira total. Nos estágios iniciais da doença, há o desenvolvimento dos exsudatos na retina. A detecção dos exsudatos na retina é essencial para o diagnóstico precoce da Retinopatia Diabética. Neste trabalho, apresentamos um modelo de aprendizado profundo com uma Rede Neural Convolucional para diagnosticar a presença ou ausência desses exsudatos, e assim poder auxiliar especialistas na triagem da Retinopatia Diabética. Os melhores resultados obtidos foram uma Sensibilidade de 99,52%, Especificidade de 100% e Precisão de 99,76% para todo o conjunto de teste. Assim, acreditamos que o método proposto em nosso trabalho seja capaz de integrar um sistema clínico real.

Keywords: Diagnóstico Automático da Presença de Exsudatos em Imagens de Retina Utilizando Aprendizado Profundo.

13.1 Introdução

O Diabetes é uma doença metabólica crônica caracterizada por níveis elevados de glicose no sangue, essa anormalidade pode causar sérios danos ao coração, vasos sanguíneos, olhos, rins e nervos. Segundo a Organização Mundial da Saúde, cerca de 422 milhões de pessoas têm Diabetes em todo o mundo e 80% dessas pessoas residem em países de baixo ou médio desenvolvimento (OMS, 2020). No Brasil, os dados sobre a doença são alarmantes, em 2017 o país ocupava a quarta posição entre os países com maior número de pessoas diabéticas (SBD, 2019-2020).

A duração do Diabetes é o preditor mais forte para o desenvolvimento e progressão da Retinopatia Diabética (RD), que é uma complicaçāo do Diabetes. A RD afeta diretamente a visão do paciente, podendo, em alguns casos, se não descoberta precocemente, chegar a perca total da visão (FONG et al., 2004).

A RD possui dois estágios clínicos, que são o estágio proliferativo e o não proliferativo (JOSHI; KARULE, 2018). O estágio não proliferativo indica os primeiros sinais da RD e caracteriza-se principalmente pela presença de aneurismas, pequenas hemorragias e exsudatos na retina (HENDRICK; GIBSON; KULSH-RESHTHA, 2015). O estágio proliferativo se dá em um momento mais avançado da RD, e se apresenta com a perda visual grave. Obtendo-se o diagnóstico da RD ainda no seu estágio não proliferativo, há grandes chances do paciente não chegar a cegueira total (SBD, 2014-2015).

O diagnóstico manual da presença de exsudatos em imagens da retina é um processo demorado, uma vez que demanda tempo e esforço por parte do profissional. Exigir um conjunto de esforços repetitivos pode provocar cansaço ao especialista, deixando-o propício a equívocos durante a análise das imagens (FILHO et al., 2014c). Assim, automatizar esse processo pode economizar tempo, fornecer resultados rápidos e precisos, além de auxiliar os oftalmologistas a tomarem decisões de tratamento em tempo hábil ou atuar como forma de triagem (Win; Choomchuay, 2016).

Nos últimos anos, houve um crescente interesse no desenvolvimento e uso de técnicas de processamento de imagens, onde os principais objetivos são aumentar a precisão do diagnóstico e fornecer uma segunda opinião aos médicos. Como alternativa para auxiliar no diagnóstico por imagem, entram os sistemas de Diagnóstico Auxiliado por Computador (CADx) (OLIVEIRA et al., 2015b).

Atualmente, os estudos no âmbito do aprendizado profundo estão fazendo grandes avanços na solução de problemas que resistiram às melhores tentativas da comunidade de inteligência artificial por muitos anos. Ele se mostrou muito bom em descobrir estruturas complexas em dados de alta complexidade, portanto, pode ser aplicado em vários contextos distintos. Os métodos de aprendizagem profunda são capazes transformar representações brutas em representações mais abstratas (??).

Este trabalho propõe um modelo de aprendizagem profunda treinado com imagens de retina. O resultado é a capacidade de classificar precisamente a presença de exsudatos nas imagens de retina, o que proporciona uma segunda opinião aos especialista e consequentemente diminui seus esforços no momento do diagnóstico da RD.

As principais contribuições deste trabalho incluem a concepção e implementação de um modelo de aprendizagem profunda baseado em Redes Neurais Convolucionais. Contribuímos também para área médica, fornecendo um sistema totalmente automático, robusto e de baixo custo, capaz de atuar como uma segunda opinião ao médico.

13.1.1 Objetivo

O objetivo principal desse trabalho é a implementação de um método capaz de auxiliar do diagnóstico da RD.

13.1.2 Objetivos específicos

1. Realizar um levantamento bibliográfico a cerca do tema proposto, e com isso, nortear e fundamentar o método proposto;
2. Aplicar Redes Neurais Convolucionais para desenvolver um modelo treinável;
3. Desenvolver um método que seja preciso e fácil de ser implantando em um ambiente real para dar uma segunda opinião ao especialista.

13.1.3 Organização do capítulo

O restante deste trabalho está organizado da seguinte forma. O referencial teórico necessário é apresentado na Seção 13.3. Os trabalhos relacionados com este são discutidos na Seção 13.2. A metodologia e detalhes do modelo proposto estão descritos na Seção 13.4. Os resultados obtidos e a conclusão são apresentados nos Capítulos 13.5 e 13.6, respectivamente.

13.2 Trabalhos Relacionados

As técnicas de aprendizado profundo são amplamente difundidas em diversas áreas, no âmbito médico não é diferente. Neste Capítulo, abordaremos trabalhos correlatos ao método proposto.

[Osareh et al. \(2003\)](#) propõe um método para classificação de exsudatos na retina seguindo de etapas de segmentação, usando o agrupamento de regiões semelhantes, seguindo algumas etapas importantes do pré-processamento. Para classificar as regiões segmentadas em exsudatos e não exsudatos, foi utilizado um classificador de rede neural artificial. O método obteve uma sensibilidade de 96,70% e especificidade de 100%.

[Reza, Eswaran e Hati \(2008\)](#) apresenta um algoritmo para detectar exsudatos. Os autores utilizam o canal verde das imagens e algumas etapas de pré-processamento, como filtragem média, ajuste de contraste e limiar. As outras técnicas de processamento utilizadas são abertura morfológica, operador de máximos estendidos, imposição de mínimos e transformação de bacias hidrográficas. Seus resultados foram uma sensibilidade de 95% e especificidade de 88,90%.

[Sopharak et al. \(2010\)](#), apresenta experimentos sobre seleção de características e classificação de exsudatos usando os classificadores *Naive Bayes* e Máquina de Vetores de Suporte. Primeiro, ajustam-se os classificadores com base em 15 características extraídas a partir de 115.867 imagens com exsudatos e em um número igual para imagens sem exsudatos. O melhor resultado encontrado foi uma sensibilidade geral de 92,28%, especificidade de 98,52% e acurácia de 98,41%.

[Harangi, Antal e Hajdu \(2012\)](#) propõe a identificação de possíveis exsudatos usando a morfologia em escala de cinza. Em seguida, são extraídos mais de 50 descritores de características para cada pixel candidato, onde os mais relevantes são selecionados e removidos. Após essa etapa, as imagens são utilizadas para treinar o classificador *Naive Bayes*. O método obteve uma sensibilidade de 63%.

[Prentašić e Lončarić \(2015\)](#) e [Yu, Xiao e Kanagasingam \(2017\)](#) propuseram métodos para detecção automática de exsudatos. Ambos consistem na aplicação de algoritmos de pré-processamento nas imagens, a fim de retirar as regiões que possam se confundir com os exsudatos. Após retirar os possíveis exsudatos, as imagens obtidas ao fim desse processo são colocadas na rede para treino. Os resultados obtidos foram, respectivamente, uma sensibilidade de 77% e 88,85%.

[Prentašić e Lončarić \(2016\)](#), utilizou de uma rede neural convolucional profunda para detecção de exsudatos. Com o intuito de incorporar um conhecimento anatômico de alto nível sobre possíveis localizações de exsudato, é feita uma combinação entre a saída dos procedimentos de detecção de disco óptico e detecção de vaso e os dados de saída da rede. Esse procedimento diminui as chances da rede encontrar um não exsudato. O método obteve uma sensibilidade de 78%.

Os trabalhos apresentados neste Capítulo utilizam de técnicas para extração de características, além de apresentar a necessidade da remoção dos possíveis exsudatos na imagem, para então treinar um modelo e realizar testes. Esse processo envolve mais etapas, o que pode dificultar a sua implantação em um sistema clínico real. Outro fato importante é a remoção dos possíveis exsudatos, pois esse procedimento pode resultar na remoção desnecessária de algum exsudato na imagem, diminuindo as características imprescindíveis para resolução do problema.

O nosso trabalho necessita apenas da aplicação da normalização das imagens. Não aplicamos algoritmos para extraír as regiões que podem se confundir com os exsudatos, pois, após estudo do problema, verificamos que a região que mais se assemelha com os exsudatos é o disco óptico. As duas classes de imagens que utilizamos (com exsudatos e sem exsudatos) possuem o disco óptico. Portanto, não foi necessário removê-lo, assim conseguimos diminuir as chances de ter algum problema com a perda de característica.

13.3 Referencial Teórico

É necessário abordar conceitos, conhecimentos e informações pertinentes a este trabalho. Contudo, este Capítulo irá tratar de facilitar o seu entendimento.

13.3.1 Processamento Digital de Imagens

Uma imagem é representada por uma função bi-dimensional $f(x, y)$, sendo x e y coordenadas espaciais. O valor de f , em um ponto qualquer (x, y) , equivale ao brilho da imagem naquele ponto. Caracterizando-a matematicamente, uma imagem digital $f(x, y)$ pode ser considerada uma matriz, cujos elementos são os pixels (NETO, 1999).

O Processamento Digital de Imagens (PDI) expressa-se pela manipulação de imagens por meio do uso de computadores. Nas últimas décadas, seu uso vem crescendo exponencialmente, por exemplo, as aplicações na área da medicina, entretenimento e sensoriamento remoto. Sistemas multimídia, que são um ponto-chave na sociedade moderna, estão cada vez mais dependentes do PDI (SILVA; MENDONCA, 2005).

Resumidamente, as técnicas de PDI visam alterar uma imagem de modo que seja possível um humano encontrar recursos mais facilmente. O reconhecimento humano é concretizado pelo cérebro, mas os programas que possuem inteligência artificial diferenciam-se por substituírem o cérebro para fazerem essa tarefa, ou seja, a interpretação é realizada por meio da visão computacional (YOUN et al., 2018).

Na medicina, houveram grandes avanços tecnológicos sobre o uso de imagens no diagnóstico médico. Paralelamente, esse fato possibilitou que, com novos equipamentos e melhorias na qualidade das imagens, fosse possível aplicar técnicas de PDI, a fim de facilitar a interpretação das mesmas (FILHO; NETO, 1999).

13.3.2 Aprendizado profundo

As tecnologias que envolvem o aprendizado de máquina estão presentes em vários aspectos da sociedade moderna e potencializam muitos segmentos que vão desde pesquisas na Web até a câmera dos nossos *smartphones*. Aplicativos que identificam objetos nas imagens, transcritores de texto e aplicativos de comandos de voz fazem uso de um ramo do aprendizado de máquina chamado aprendizado profundo (??).

A forma mais comum de aprendizado de máquina, seja profunda ou não, é o aprendizado supervisionado. Quando se quer construir um sistema capaz de classificar imagens pelo seu conteúdo, como, por exemplo, gato, cachorro ou uma ave, alguns passos devem ser seguidos. Primeiro deve-se coletar um grande conjunto de imagens de gatos, cachorro e aves, cada categoria deve ser identificada por um rótulo. No momento do treinamento, a máquina transforma cada imagem em vetores de representações numéricas, um para cada categoria (??).

13.3.3 Retinopatia Diabética

A RD progride de anormalidades não proliferativas leves, que se caracterizam pelo aumento da fragilidade vascular, até o estágio proliferativo, caracterizado pelo aumento de vasos sanguíneos na retina. Existem dois métodos aceitos para examinar a RD, a oftalmoscopia junto a biomicroscopia e a fotografia de fundo da retina, sendo o último o mais sensível no diagnóstico da RD (DAVIDSON et al., 2007).

Sendo uma importante causa de cegueira em todo o mundo, a RD apresenta poucos sintomas visuais ou oftalmológicos até chegar a perda visual. Atualmente existem métodos eficazes para retardar a perda visual, porém não há como restaurar a visão perdida. Esses tratamentos visam prevenir a perda total da visão, portanto é muito importante identificar e tratar os pacientes no início da doença. Para isso, os especialistas recomendam que pacientes com Diabetes devem ser rotineiramente submetidos à avaliação para detectar doenças tratáveis como a RD (KINGMAN, 2004a).

13.4 Metodologia

A metodologia empregada neste trabalho, consiste em: I), aquisição da base de imagens, com os respectivos diagnósticos. II), aplicar um pré-processamento nas imagens, seguido do aumento de dados. III), modelar e implementar a arquitetura da Rede Neural Convolucional, que passou por alguns ajustes até sua versão final onde obteve-se o melhor resultado. Na terceira etapa, o modelo foi treinado. E IV), classificação das imagens com um conjunto de imagens de teste. A Figura 95 descreve um resumo das etapas da metodologia utilizada.

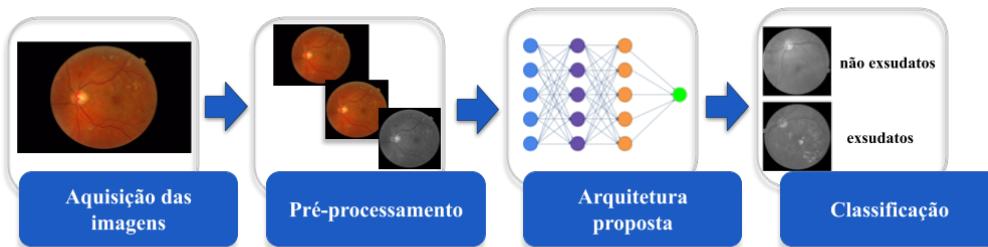


Figura 95 – Resumo da metodologia proposta.

13.4.1 Base de Imagens

A base de imagens MESSIDOR ([DECENCIÈRE et al., 2014](#)) é uma base pública que foi criada para facilitar estudos sobre diagnósticos de RD assistidos por computador. O banco de dados contém um diagnóstico médico para cada imagem, mas nenhuma anotação manual nas imagens, como contornos ou posição das lesões. As imagens foram adquiridas por 3 departamentos oftalmológicos.

Junto as imagens é fornecido um arquivo contendo o respectivo diagnóstico sobre o grau de RD para a imagem. As anormalidades são diagnosticadas pelo grau de RD. Esse grau varia de 0 a 3, onde 0 significa não haver sinais de RD e os graus 2 e 3 apresentam de sinais exsudatos e hemorragias, respectivamente. A Figura 96(a) apresenta a imagem da retina com a presença de exsudatos, e a Figura 96(b) sem a presença de exsudatos.

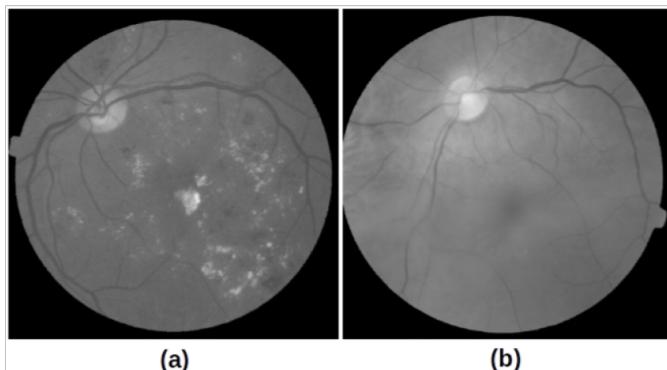


Figura 96 – (a) Imagem da retina com exsudatos. (b) Imagem da retina sem exsudatos.

A base contém um total de 1.200 imagens. Para o desenvolvimento deste trabalho, foram selecionadas desse conjunto, um total de 1.046 imagens. Sendo 500 imagens com a presença de exsudatos e 546 imagens sem sinais de RD. As imagens do nível 1 não foram utilizadas, pois seu diagnóstico refere-se a microaneurismas, não se aplicando a este trabalho.

13.4.2 Pré-processamento

As imagens do conjunto de dados são disponibilizadas, originalmente, nas dimensões de 1440x960, 2240x1488 e 2304x1536 pixels. Para padronizar a dimensão de todas as imagens, aplicamos um algoritmo para cortar e redimensioná-las para quadrados de 460 pixels, de modo que não fossem cortadas regiões referentes à retina. Paralelamente a esta etapa, todas as imagens são convertidas de RGB para escala de cinza. É interessante expôr que, diferentemente dos trabalhos relacionados aqui apresentados, este foi o único

pré-processamento aplicado nas imagens, a intenção é normalizar os dados que serão passados para treinar a rede.

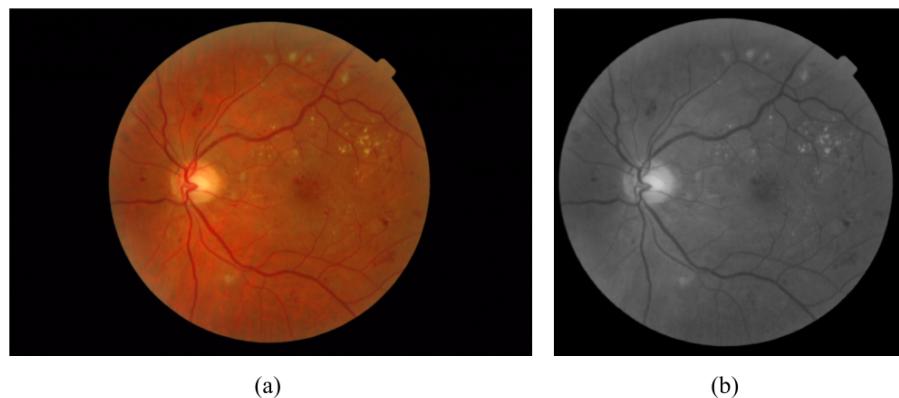


Figura 97 – (a) Imagem original. (b) Imagem resultante desta etapa.

13.4.3 Rede Neural Convolucional

As Redes Neurais Convolucionais herdam características das Redes Neurais Artificiais baseadas em arquiteturas hierárquicas de *feed-forward* que concatenam, principalmente, diferentes níveis de camadas convolucionais e de agrupamento máximo. Cada camada opera em suas entradas para produzir uma mudança de representações para as camadas seguintes (RAFEGAS; VANRELL, 2018).

Existem dois fatores, que consideramos principais, e que motivam o uso das Redes Convolucionais em diversos ramos da computação. O primeiro é que diferentes arquiteturas podem ser facilmente modeladas e implementadas, necessitando apenas conhecer o campo de visão onde serão aplicados, no nosso caso, o aprendizado profundo para reconhecimento de padrões em imagens.

O outro fator é o de serem treinadas automaticamente, isso possibilita que, mostrando o conjunto de treino em duas classes distintas, ela consiga encontrar os recursos importantes e obter o desempenho ideal para resolver sua tarefa.

Tomando como princípio que todas as imagens, saudáveis ou não, possuem a presença do disco óptico, a rede foi projetada para ser capaz de entender isso. Logo, sua tarefa é encontrar o padrão presente em cada classe de imagens passadas para treino, que nesse caso, é a presença ou ausência dos exsudatos.

O modelo final, no qual obtivemos o melhor resultado, contém um total de oito camadas de convolução, oito camadas de normalização, quatro camadas de agrupamento máximo, seis camadas de dispersão de dados, uma camada totalmente conectada e três camadas densas. A Figura 98 apresenta um resumo do nosso modelo.

13.4.3.1 Camadas de Convolução

Uma das partes mais importantes são as camadas convolucionais. Na Figura 98, para a parte 1, 2, 3 e 4 as camadas de convolução têm 16, 32, 64 e 128 filtros, respectivamente, e possuem ativação com a função *Rectified Linear Unit* (ReLU). A quantidade de pesos é definida pelo *kernel size* (definido em 3x3), logo, cada filtro usará 9 pesos para transformar um pixel em uma média ponderada dele mesmo e seus oito vizinhos.

13.4.3.2 Camadas de Agrupamento Máximo

A parte negativa das camadas convolutivas é o aumento excessivo de dados na Rede. Para diminuir isso, aplicamos uma camada de agrupamento máximo de tamanho 2x2 sempre após duas camadas convolutivas. Ela agrupa 4 dos 9 pesos do filtro e seleciona o maior valor, assim, essa camada consegue reduzir a quantidade de dados na rede. A Figura 99 ilustra como ocorre o processo de agrupamento máximo.

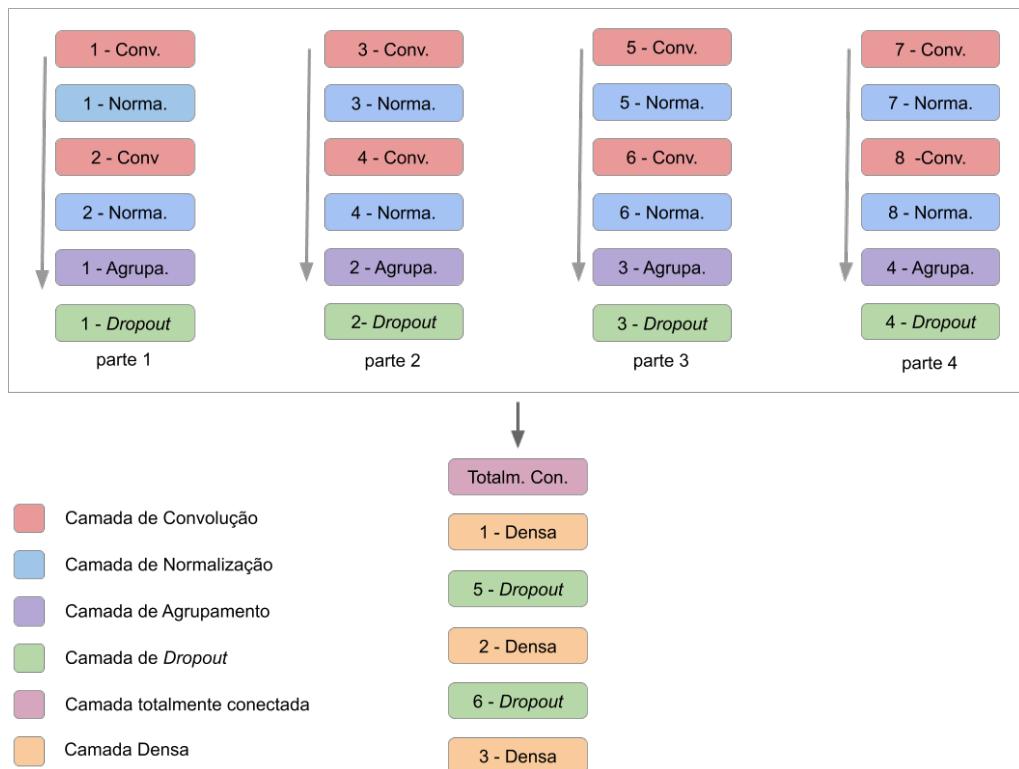


Figura 98 – Arquitetura do modelo proposto.

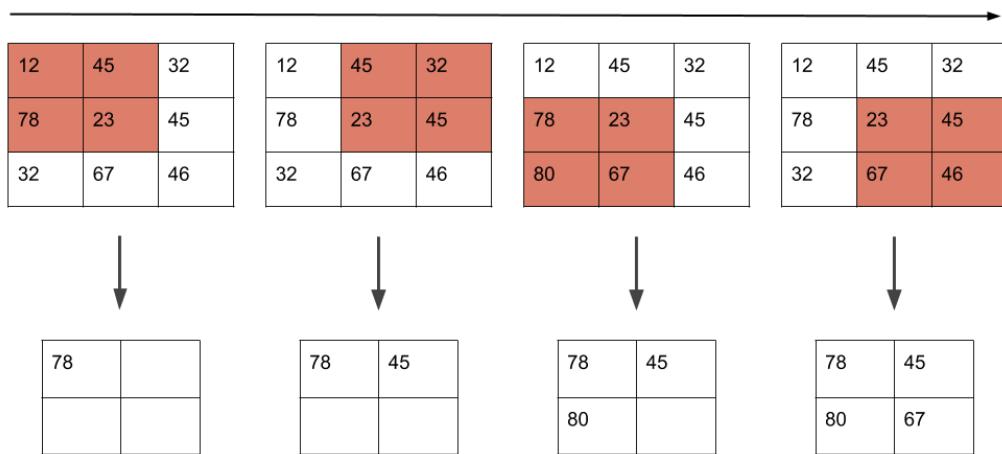


Figura 99 – Funcionamento do agrupamento máximo utilizado.

13.4.3.3 Camadas de Normalização

A distribuição das entradas de cada camada muda durante o treinamento, conforme os parâmetros das camadas anteriores mudam, tornando o treinamento mais lento. (IOFFE; SZEGEDY, 2015) propõe um método que incorpora a normalização desses dados no próprio modelo e realiza a normalização para cada etapa de treinamento. A normalização em lotes nos permite usar taxas de aprendizado muito mais altas e ter menos cuidado com a inicialização, e previne que aconteçam problemas de atraso no treinamento.

13.4.3.4 Camadas de Dispersão

As camadas de dispersão de dados, fornecem uma técnica que ajuda impedir o superajuste da rede. Para fazer isso a dispersão ocorre pelo descarte de unidades de treino, ou seja, algumas unidades são removidas da rede, juntamente com todas as suas conexões de entrada e saída. No modelo proposto foram utilizadas 4 camadas com dispersão de 0,25 e as duas últimas com dispersão de 0,5.

13.4.3.5 Camadas Densas

As camadas densas que utilizamos na rede, possuem ativação ReLu, com exceção da última camada. Ela, possui dois neurônios que recebem o mapeamento do restante da rede e produz a saída necessária para classificar as imagens, sua ativação é feita com a função Softmax. Utilizamos essa função, porque seu objetivo é converter qualquer vetor de números reais em um vetor de probabilidades. No nosso contexto, as probabilidades correspondem a possibilidade de uma imagem de entrada pertencer a uma determinada classe.

13.4.4 Treinamento

Existem modelos pré-treinados que auxiliam na inicialização dos valores da rede. Porém, neste trabalho, o treinamento foi realizado a partir do zero, ou seja, nenhum modelo pré-treinado foi usado e os pesos dos filtros convolucionais foram iniciados aleatoriamente no início de cada treino. Obtivemos o melhor resultado com 100 épocas de treino.

Uma etapa muito importante e que merece muita atenção, é a seleção dos dados de entrada para o treinamento da Rede. Tínhamos uma quantidade limitada de imagens para treinar um modelo de aprendizagem profunda. Redes Neurais Convolucionais são eficientes no aprendizado com uma grande quantidade de dados de entrada. Portanto, observando o pequeno número de imagens disponíveis para treinamento, foi adotado como alternativa a rotação das imagens.

13.4.4.1 Aumento de dados

As 500 imagens com presença de exsudatos foram rotacionadas 10 vezes, iniciando com grau 0, que representa a imagem original, e acrescentando 30 graus a cada iteração, totalizando 5.000 imagens. Já as 546 imagens sem a presença de exsudatos foram rotacionadas 9 vezes, iniciando com grau 0, e acrescentando 30 graus a cada iteração, totalizando 4.914 imagens.

A Figura 100 mostra como ficou uma imagem depois de aplicarmos as rotações. A Figura 100(a) é a imagem sem rotação. As imagens da Figura 100(b) e da Figura 100(c) apresentam o resultado após aplicarmos rotações de 30 e 60 graus, respectivamente. Ao final desta etapa, o novo conjunto de dados passou a ter 9.914 imagens. Verificamos que haviam mais imagens com presença de exsudatos, portanto, foram retiradas, aleatoriamente, 86 imagens dessa classe, ficando 4.914 imagens para ambas as classes, e um total de 9.828 imagens.

De todo o conjunto de dados disponível, foram utilizadas 70% das imagens para treinar o modelo proposto, 10% para validação durante o treino e 20% para teste, divididos aleatoriamente com a mesma quantidade de imagens em ambas as classes (com exsudatos e não exsudatos).

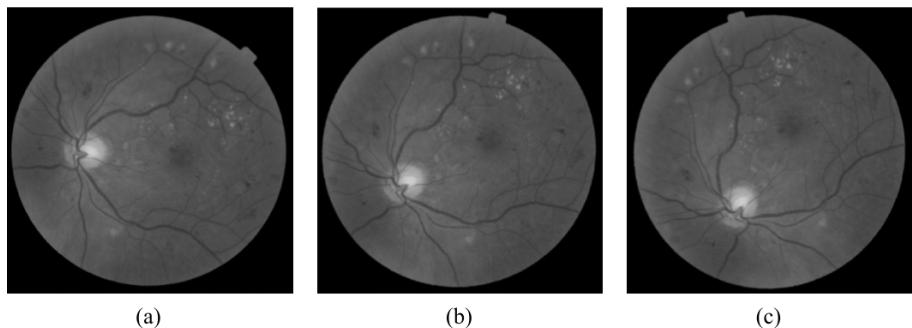


Figura 100 – (a) Imagem original. (b) Imagem após o final desta etapa.

13.4.5 Validação dos resultados

Foram selecionadas algumas métricas estatística comumente utilizadas em sistemas CADx, como Sensibilidade, Especificidade e Acurácia, a fim validar os resultados. Essas métricas foram encontradas usando uma matriz de confusão. A matriz é formada por quatro valores, onde a sua diagonal principal mostra os valores classificados acertadamente, como mostra a Figura 101.

		RÓTULO CLASSIFICADO	
		Presença	Ausência
RÓTULO VERDADEIRO	Presença	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Ausência	Falso Positivo (FP)	Verdadeiro Negativo (VN)
		Presença	Ausência

Figura 101 – Matriz de confusão.

O valor de VP representa o número imagens classificadas corretamente como exsudatos; FP, representa o número de imagens classificadas como exsudatos, porém não possui exsudatos; FN, marca o número de imagens classificadas como saudáveis, mas que têm exsudatos e VN que representa o número de imagens classificadas acertadamente sem sinais de RD, ou seja, saudáveis.

Sensibilidade (S), avalia a capacidade do teste detectar a doença quando ela está presente. Como mostra a Equação 13.1.

$$S = \frac{VP}{VP + FN} \quad (13.1)$$

Especificidade (E), avalia a capacidade do teste não detectar a doença quando ela não está presente. Como mostra a Equação 13.2.

$$E = \frac{VN}{VN + FP} \quad (13.2)$$

Acurácia (A) que é a proporção de acertos, ou seja, VN e VP, pela quantidade total da amostra de teste. Como é mostrado na Equação 13.3.

$$A = \frac{VP + VN}{VP + FP + VN + FN} \quad (13.3)$$

13.5 Resultados

Para que um sistema possa ser realmente usado para auxiliar no diagnóstico da RD, é necessário encontrar um bom equilíbrio entre as métricas de avaliação. Todos os experimentos foram feitos utilizando sempre a mesma quantidade de imagens para treino e teste. A Tabela 89 mostra os resultados dos experimentos até chegar ao melhor resultado.

Tabela 89 – Melhores resultados variando o número de épocas.

Experimento	Épocas	S (%)	E (%)	A (%)
1º	20	92,59	89,56	91,02
2º	20	64,21	99,88	82,05
3º	30	84,54	100	92,27
4º	100	94,53	100	97,27
5º	100	99,52	100	99,76

O primeiro experimento foi realizado utilizando a parte 1 da Rede demonstrada na Figura 98 com 20 épocas, esse número de épocas foi aumentado, porém sem sucesso no aumento das métricas. O segundo experimento foi realizado adicionando a parte 2 da Rede, porém, sem nenhuma camada de normalização dentro da Rede. O terceiro experimento foi feito utilizando as partes 1, 2 e 3 da Rede e com três camadas de normalização. Nesse experimento, o número de épocas também foi aumentado, mas obteve-se o melhor resultado com 30 épocas.

O quarto experimento foi feito utilizando as 4 partes da arquitetura proposta, porém com apenas 4 camadas de normalização dos dados. O resultado foi uma acurácia de 97,27%. No quinto experimento, após cada camada de convolução, adicionamos uma camada de normalização, o resultado foi uma acurácia de 99,76% para o conjunto de teste. Todos os testes realizados utilizaram da mesma quantidade de imagens.

Para mostrar que o aumento de dados apresentado na Seção 13.4.4.1, proporcionou melhores resultados para a rede e considerando a amostra de treino, (80%) e teste (20%), criamos a Tabela 90 que contém os experimentos mostrados anteriormente na (Tabela 89), sem aplicar o aumento de dados.

Tabela 90 – Melhores resultados sem aplicar o aumento de dados.

Experimento	Épocas	S (%)	E (%)	A (%)
1º	20	67,43	57,57	60,5
2º	50	75,21	52,4	63,8
3º	100	77,54	53,2	65,3
4º	150	77,92	56,67	67,29

Na Tabela 91, apresentamos os resultados obtidos com experimentos feitos com modelos de aprendizagem profunda bem renomados na Literatura, como VGG16 (REN et al., 2015), VGG19 (REN et al., 2015) e Xception (CHOLLET, 2016). Para fins de comparação, foi utilizado o mesmo conjunto de imagens para todos os experimentos, tanto para treino quanto nos testes.

Tabela 91 – Comparação com outros modelos da Literatura.

Modelo	S (%)	E (%)	A (%)
VGG16	97,06	97,90	97,90
VGG19	97,38	98,97	98,16
Xception	96,95	98,64	97,78
Método	99,52	100	99,76

Conforme mostrado na Tabela 91, nosso método pôde alcançar resultados mais promissores do que aqueles comumente usados na literatura disponível. Acreditamos que isso seja porque o modelo é capaz de extrair milhares de características através das 8 camadas de convolução, isso também somado às camadas

de dispersão e normalização de dados que tornam a rede menos propícia ao *overfitting*, melhorando sua capacidade de adaptação a novos dados que serão parâmetros para as camadas posteriores.

A comparação com outros trabalhos é uma tarefa delicada, pois, os trabalhos diferenciam em vários aspectos, como: técnicas usadas, base de imagens usadas, quando usam a mesma base, os casos de teste e treino podem ser distintos. Desse modo, a comparação feita aqui, leva em conta apenas os valores quantitativos com intuito de mostrar que o método proposto, compara-se com os demais trabalhos disponíveis na literatura correlata.

Tabela 92 – Comparação com os trabalhos relacionados.

Trabalho	S (%)	E (%)	A (%)
Osareh et al. (2003)	95	88,90	–
Reza, Eswaran e Hati (2008)	96,7	100	–
Sopharak et al. (2010)	92,28	98,52	98,41
Harangi, Antal e Hajdu (2012)	63	–	–
Prentašić e Lončarić (2015)	77	–	–
Prentašić e Lončarić (2016)	–	–	78
Yu, Xiao e Kanagasingam (2017)	88,85	96	91,92
Proposto	99,52	100	99,76

Na Tabela 92, apresentamos, resumidamente, uma comparação entre os resultados obtidos no nosso trabalho, com outros trabalhos relacionados. Como pode-se verificar, obtivemos um resultado melhor ou igual em relação aos demais. É importante ressaltar que precisão geral do nosso método foi melhor que todos os outros.

Como apresentado nas Tabelas 89 e 92, apresentamos um método com bom resultado. Um fator importante que ainda podemos ressaltar, são as máquinas que utilizamos para desenvolver todo esse trabalho. Ela possui processador Intel Core I5 inside de 64 bits, 4GB de memória RAM DDR3. Não dispomos de máquinas com Unidade de Processamento Gráfico de ponta.

Se tratando de sistemas de aprendizagem profunda que, na maioria das vezes, necessitam de máquinas potentes para serem implementados, conseguimos superar essa dificuldade. Portanto, ele pode ser usado em regiões mais pobres e de difícil acesso, inclusive sistemas públicos de saúde, como o Sistema Único de Saúde no Brasil.

13.6 Conclusão

O aprendizado profundo está crescendo continuamente; isso se deve às inovações e ao surgimento de novas aplicações e técnicas, que impulsionam uma nova era de automação inteligente. Este trabalho propôs uma metodologia para diagnosticar a presença ou ausência de exsudatos na retina, por meio de uma técnica de aprendizado profundo. Além de mensurar quantitativamente o problema exposto, o método mostrou que é possível obter bons resultados mesmo com recursos e hardware limitados.

Por exigir tempo e esforço por parte dos especialistas, o diagnóstico manual da presença de exsudatos é um processo que causa desconforto e pode levar para decisões erradas de sua parte. Os trabalhos relacionados apresentaram diferentes maneiras de automatizar esse diagnóstico, mas algumas etapas são necessárias para extrair dados essenciais das imagens. Usamos uma metodologia que usa a imagem inteira. Tivemos dificuldade em termos algumas amostras no conjunto de dados de treinamento, o que foi superado com a criação de versões rotacionadas das imagens originais.

No entanto, nosso trabalho tem relevância no campo computacional e médico, uma vez que essas novas tecnologias fornecem ao profissional médico recursos para auxiliar no diagnóstico de RD, tornando o método proposto uma ferramenta benéfica para a sociedade, podendo ser aplicado em ambientes clínicos reais, atuando como uma segunda opinião ao médico ou na triagem de pacientes.

Classificação automática de texturas de imagens de pele em melanoma e não melanoma utilizando índices de diversidade

Autores: Antônio Oseas de Carvalho Filho; Alex William Leal.

Resumo: Skin cancer is defined by the abnormal and uncontrolled growth of the cells that make up the skin. Among skin cancers, the least frequent is melanoma, but it has the worst prognosis, being the most lethal and corresponding to 3% of the 30% of skin cancers in Brazil. Despite being the most harmful, if there is an early detection the chances of cure are 95%. The method applied for the early diagnosis of melanoma is the dermatological examination. The present study presents a method of classifying skin lesions in melanoma and non-melanoma by means of images. In the first stage, the acquisition of images from the PH2 base is made. In the second, pre-processing, using image processing techniques. In the third stage, the extraction of characteristics, through the diversity indexes. And in the fourth step, it is the classification in melanoma or not, where the classifiers *Random Forest*, *Sequential Minimal Optimization (SMO)*, *Multilayer Perceptron (MLP)* and *Instance Based Learning (IBK)*. Therefore, the process is completed with the results validation. In this method, texture descriptors based on diversity indexes were used to extract characteristics. For all tests performed, the best result reached an accuracy of 90,5%, sensitivity of 97,5%, specificity of 62,5%, ROC curve of 0,800 and a Kappa of 0,699. The use of diversity indexes to describe patterns in regions of mammography images proved to be moderate in the categorization of melanoma and non-melanoma. It contributed positively in two different areas. Since in the health area, it provided an automatic methodology to aid in the diagnosis of skin lesions in melanoma and non-melanoma, which, despite not providing excellent results, is promising. And in computing, with the adaptation of techniques from other areas of knowledge that served as texture descriptors, thus allowing them to be used for characterizations in other types of images.

Keywords: Skin Cancer, Melanoma, Texture Descriptors, Diversity Indices.

14.1 Introdução

O câncer de pele é uma doença que é provocada pelo crescimento anormal e descontrolado das células que compõem a pele. Essas células se dispõem formando camadas e, de acordo com as que forem afetadas, são definidos os diferentes tipos de câncer. Os mais comuns são os carcinomas basocelulares e os espinocelulares. O câncer de pele mais agressivo é conhecido como melanoma, sendo o mais raro e nocivo entre os outros tipos de cânceres de pele (INCA, 2020). Quanto maior for a exposição solar, mais elevado é o risco de tumores pré-cancerosos e cânceres de pele, incluindo o melanoma.

O câncer de pele responde por 30% de todos os diagnósticos desta doença no Brasil, sendo que o Instituto Nacional do Câncer (INCA) registra, a cada ano, cerca de 180 mil novos casos (INCA, 2020). Entre os cânceres da pele, o melanoma é o menos frequente, porém o mais letal. Segundo o INCA (2020), no Brasil, embora o câncer de pele seja o mais frequente tendo 30% de todos os tumores malignos registrados, apenas 3% são do tipo melanoma, com 8.450 novos casos diagnosticados com essa doença no ano de 2020. As mulheres são maioria nos números de casos, com um total de 4.250 casos e os homens com 4.200. O número de mortes são de 1.978, sendo 1.159 homens e 819 mulheres.

Com o uso de exames dermatológicos, a incidência de câncer de pele pode ser tratada de forma precoce, o que aumenta a chance de cura do paciente, porém, alguns fatores podem comprometer os prognósticos, como a inexperiência do especialista, a qualidade do exame, diagnósticos de doenças parecidas na pele, acarretando em possíveis erros na detecção da doença (INCA, 2020).

No diagnóstico dermatológico, as lesões são examinadas clinicamente, utilizando primeiramente a técnica de análise das características e textura, para então diagnosticá-las e tratá-las. Para facilitar este processo, os dermatologistas podem dispor de sistemas computacionais, que analisam as características das lesões de forma mais precisa, utilizando imagens digitais, para auxiliar no seu diagnóstico (OLIVEIRA, 2012b).

A área de processamento e análise digital de imagens é um dos campos mais importantes da ciência médica devido ao rápido e contínuo progresso na visualização de imagens médicas e avanços nos métodos de diagnóstico assistido por computador e terapias guiadas por imagens. Essa área vem sendo essencial para a detecção precoce, o diagnóstico e a avaliação da resposta ao tratamento do câncer (SILVA; PATROCÍNIO; SCHIABEL, 2019).

De acordo com Guaragnella e Rizzi (2020), os sistemas de diagnóstico auxiliados por computador (*computer-aided diagnosis CADx*) tem como objetivo aumentar o desempenho clínico, auxiliando na identificação e localização precoce de potenciais anormalidades. Portanto, o sistema CADx opera como uma segunda opinião automatizada ou como um sistema de dupla leitura que auxilia os dermatologistas na avaliação precoce do câncer de pele e no acompanhamento de lesões cutâneas pigmentadas.

O trabalho desenvolveu um método capaz de integrar um sistema CADx atuando no diagnóstico automático do melanoma, categorizando em melanoma ou não. Para isso, são extraídas as características das imagens baseadas em textura utilizando índices de diversidade. Por fim, é feito o reconhecimento de padrões com múltiplos classificadores para avaliar a relevância da pesquisa proposta, com base nos resultados obtidos em cada classificador e determinar se o trabalho proposto é relevante para que os especialistas tenham uma segunda opinião no diagnóstico da doença.

14.1.1 Objetivos

Desenvolver uma abordagem para a classificação de regiões em melanoma e não melanoma, utilizando índices de diversidade.

Os objetivos específicos são:

1. Desenvolver e adaptar técnicas para caracterizar propriedades de textura;
2. Classificar as regiões de interesse em melanoma e não melanoma.
3. Analisar a usabilidade do uso de índices de diversidade como forma de extração de características; e

4. Construir uma metodologia para que os especialistas tenham uma segunda opinião no diagnóstico do melanoma.

14.1.2 Organização do capítulo

Além da Introdução, este trabalho está organizado na seguinte ordem: A Seção 14.2 descreve todo o embasamento teórico para o entendimento da metodologia utilizada no trabalho; Na Seção 14.3, são mostrados os principais trabalhos relacionados; Na Seção 14.4, é abordado todo o desenvolvimento da pesquisa, tendo como início a aquisição das imagens, extração de características, classificação e por último a validação da metodologia proposta; Na Seção 14.5, detalha os resultados obtidos com a metodologia proposta; e Por fim, é apresentado a conclusão do trabalho proposto na Seção 14.6.

14.2 Referencial Teórico

Para melhor entendimento deste trabalho, faz-se necessário a descrição de conceitos relacionados ao tema. Este capítulo aborda todo o embasamento teórico utilizado na metodologia proposta para facilitar sua compreensão. Diante disto, serão abordados os seguintes tópicos: Melanoma, Sistemas CADx, Processamento digital de imagens, Pré-processamento, Índices de diversidade, Reconhecimento de padrões e Métricas de validação.

14.2.1 Melanoma

O melanoma é o câncer originado a partir do crescimento desordenado dos melanócitos. Os melanócitos são as células que produzem o pigmento principal na produção de cor da pele, a melanina. Na maioria dos casos, a exposição excessiva à radiação ultravioleta é a causadora dos danos ao DNA dos melanócitos. Um dos tumores mais perigosos, o melanoma tem a capacidade de invadir qualquer órgão, criando metástases, inclusive cérebro e coração. Portanto, é um câncer com grande letalidade (UTIYAMA et al., 2019).

O diagnóstico precoce do melanoma é fundamental. Embora apresente pior prognóstico, avanços na medicina e o recente entendimento das mutações genéticas, que levam ao desenvolvimento dos melanomas, possibilitaram que pessoas com melanoma avançado hoje tenham aumento na sobrevida e na qualidade de vida.

Quando o médico percebe sinais de melanoma, o mesmo analisa o histórico clínico completo do paciente, observando os sintomas e fatores de risco. Para diagnosticar o melanoma e determinar o grau de comprometimento da doença, sendo solicitado a realização de alguns exames.

Durante o exame físico, o médico observará o tamanho, forma, cor e textura das lesões em questão, e se há sangramento ou descamação. Junto com o exame físico, alguns especialistas usam dermatoscopia para avaliar as manchas na pele com mais precisão. O médico utiliza um cromatoscópio, uma lente de aumento especial com fonte de luz própria para observar a pele. Às vezes uma fina camada de álcool ou óleo é usada com esse instrumento. Uma imagem digital ou fotográfica pode ser feita do local (AMERICAN CANCER SOCIETY, 2019). A Figura 102 mostra uma imagem do tipo melanoma.

14.2.2 Sistemas Computacionais de Auxílio à Detecção e Diagnóstico

A computação na área da saúde estuda e desenvolve sistemas computacionais para apoio às atividades médicas, trabalhando com análise e processamento digital de sinal bioelétricos, processamento digital de imagens médicas, desenvolvimento de sistemas de monitoração, apoio ao diagnóstico e à decisão e sistemas de instrução inteligente auxiliada por computador.

Diagnóstico auxiliado por computador (CADx) pode ser definido como um diagnóstico utilizando os resultados de análises quantitativas automatizadas de imagens digitais, auxiliando os profissionais na tomada de decisões para o diagnóstico. (AZEVEDO-MARQUES, 2001a).



Figura 102 – Imagem de mancha na pele do tipo melanoma.

A detecção de anomalias em imagens médicas, em geral é um procedimento demorado, propício a erros e a algum grau de subjetividade devido a várias razões.

CADx alerta o profissional a examinar detalhadamente padrões suspeitos detectados pelo sistema. Isto melhora a acurácia do diagnóstico e a consistência da interpretação da imagem, servindo como uma “segunda opinião” quanto à tomada de decisões diagnósticas (PIRES et al., 2006).

14.2.3 Processamento Digital de Imagens

O processamento Digital de Imagens (PDI) Trata-se de um conjunto de técnicas para a análise de dados multidimensionais, que permitem manipular e tratar imagens com objetivo de obter informações e melhorar as características visuais da imagem. Porém, não é uma tarefa simples de ser realizada, pois ela envolve um conjunto de etapas interligadas (PEREIRA, 2018). A seguir, apresenta-se as etapas do processamento digital de imagens.

Uma imagem pode ser definida como uma função bidimensional, $f(x,y)$, em que x e y são coordenadas espaciais (plano), e a amplitude de f em qualquer par de coordenadas (x,y) é chamada de intensidade ou nível de cinza da imagem nesse ponto. Quando x , y e os valores de intensidade de f são quantidades finitas e discretas, denomina-se de imagem digital (JÚNIOR; SOARES, 2019).

Para ROCHA et al. (2014) os passos fundamentais de processamento digital de imagens podem ser divididos em cinco etapas conforme mostrado na Figura 103.

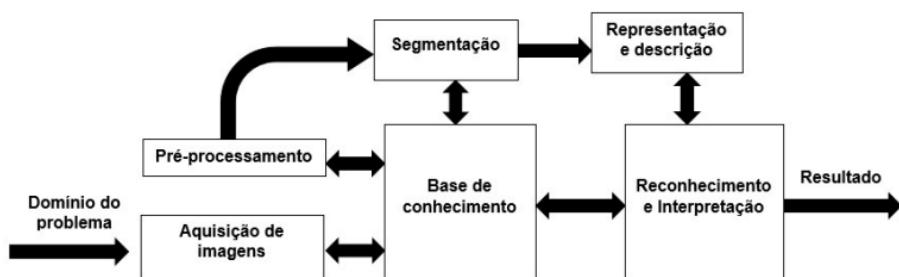


Figura 103 – Etapas de um sistema PDI (ROCHA et al., 2014).

A primeira etapa é a Aquisição de Imagens, que consiste em capturar as imagens necessárias, por meio de dispositivos móveis ou digitalização via *pixels*, para que sejam interpretadas pelas próximas etapas.

A segunda etapa é o Pré-processamento. Nesta etapa a imagem passa por um processo de filtragem, no qual são eliminados os ruídos, que podem ser obtidos durante a captura da imagem, melhorando assim a qualidade e permitindo uma melhor discriminação dos objetos presentes na mesma.

A terceira etapa é a Segmentação. Esta etapa tem por objetivo separar a imagem como um todo nas partes que a constituem e que se diferenciam entre si. É usual denominar “objetos” da imagem os grupos de *pixels* de interesse, ou que fornecem alguma informação para o PDI. Da mesma forma, a denominação

“fundo” da imagem é utilizada para o grupo de *pixels* que podem ser desprezados ou que não têm utilidade no PDI. É nesta etapa que são definidas as regiões de interesse para processamento e análise posteriores.

A quarta etapa é Representação e Descrição. Nesse momento se extraí as informações úteis da imagem processada. Quando o objetivo do processamento é obter informações numéricas, realiza-se a extração de atributos da imagem.

Por fim, a quinta e última etapa é a Classificação e Reconhecimento, tem por objetivo, de forma automática, “rotular” os objetos segmentados da imagem com base nos seus descritores através de uma base de conhecimento que foi construída na etapa anterior.

14.2.4 Pré-processamento

O pré-processamento consiste na utilização de mecanismos de processamento de imagens para o realce das mesmas. Dentre os mecanismos, para que haja maior nitidez nas características da imagem, podemos citar o aumento de contraste e a retirada de possíveis ruídos. Na metodologia proposta utilizou-se de técnicas visando o melhoramento do contraste das imagens, sendo elas, Equalização do histograma, *Contrast-limited Adaptive Histogram Equalization* (CLAHE) e Quantização.

14.2.4.1 Equalização do histograma

Este realce consiste numa maneira de manipulação de histograma reduzindo automaticamente o contraste em áreas muito claras ou muito escuras, expandindo também os níveis de cinza ao longo de todo intervalo. É uma transformação não-linear que considera a distribuição acumulativa da imagem original, para gerar uma imagem resultante, cujo histograma será aproximadamente uniforme (LEITE; ROSA, 2012).

De acordo com Jantsch e Felipussi (2013) forma mais usual de se equalizar um histograma é pelo emprego da função de distribuição acumulada da distribuição de probabilidades original, expressa pela equação 14.1,

$$S_k = \sum_{j=0}^k p_r(r_j) \quad (14.1)$$

onde $0 \leq r_k \leq 1$ representando da intensidade do nível de cinza; $k = 0, 1, \dots, L - 1$ com L o valor máximo do nível de cinza.

14.2.5 Contrast-limited Adaptive Histogram Equalization (CLAHE)

A técnica de realce de contraste CLAHE foi desenvolvida por Pizer et al. (1990), consistindo em uma variação de equalização de histograma, com o diferencial de que, na CLAHE, a equalização é adaptativa com limitação de contraste. Tal método propõe um algoritmo que ajusta o histograma original da imagem em um outro histograma pré-definido, redistribuindo, posteriormente, os valores excedentes em outras intensidades, ou seja, aqueles que não se encaixaram na distribuição, antes de computar a função de distribuição cumulativa (CARNEIRO et al., 2019). A Equação 14.2 apresenta o cálculo efetuado para o processamento da CLAHE:

$$g = [g_{\min} - g_{\max}]p(f) + g_{\min} \quad (14.2)$$

em que g é o novo valor de pixel computado, g_{\max} e g_{\min} significam o máximo e o mínimo valor do pixel, respectivamente, e $p(f)$ é a função de probabilidade de distribuição acumulada.

14.2.5.1 Quantização

A quantização de um imagem digital consiste em agrupar a gama de cores da imagem, o que acarreta na quantização da informação de cor de cada pixel da imagem. O processo de quantização transforma um conjunto de cores com M elementos em um conjunto de cores com N elementos, onde $M > N$ (NASCIMENTO et al., 2012). Por exemplo, em uma imagem de tons de cinza, o conjunto de cores e entrada é subdividido em conjuntos menores denominados intervalos de quantização ou células de quantização. Todas as cores de

uma célula de quantização assumem um mesmo valor de saída (um determinado nível de quantização), que corresponde ao calor representante dos elementos da célula ([MIRANDA et al., 2015](#)).

De acordo com [Nascimento et al. \(2012\)](#), dentre as formas para determinar as células de quantização, a escolha mais natural e simples à primeira vista, consiste em dividir o espaço de cor em células congruentes e em cada célula tomar o seu “centro” como sendo o nível de quantização associado. E no caso de quantização escalar com L níveis, as células de quantização são intervalos (c_{i-1}, c_i) de igual comprimento e em cada célula o valor de quantização é dado pela média, como apresentado na Equação 14.3:

$$q_i = \frac{c_i + c_{i-1}}{2}, 1 \leq i \leq L \quad (14.3)$$

14.2.6 Índices de diversidade

O termo diversidade, em ecologia, pode ser definido como a variedade e a variabilidade entre os organismos vivos e os complexos ecológicos em que ocorrem. Uma medida de diversidade é um parâmetro extremamente reducionista que objetiva expressar toda a complexidade estrutural de uma comunidade ecológica através de um único número ([NASCIMENTO et al., 2012](#)). A Tabela 93 apresenta uma analogia entre os termos utilizados na biologia em correspondência com o presente método.

Tabela 93 – Índice de Diversidade

Metodologia Proposta	Ecologia
Região de Interesse da imagem	Comunidade
Níveis de Cinza da Imagem	Espécies
Píxels da Imagem	Indivíduos

14.2.6.1 Índice de Shannon

Originado da teoria da informação ([SHANNON, 2001](#)), o Índice de Shannon assume que os indivíduos são randomicamente amostrados a partir de uma comunidade infinitamente grande, e que todas as espécies estão representadas na amostra. O Índice de Shannon é calculado conforme a Equação 14.4,

$$H' = - \sum_{i=1}^s p_i \ln p_i \quad (14.4)$$

onde, S é o total de espécies, \ln é o logarítmico natural, p_i a proporção de indivíduos pertencentes a espécie i na amostra, calculado com a Equação 14.5,

$$p_i = \frac{n_i}{N} \quad (14.5)$$

e n_i é o número de indivíduos na espécie i na amostra e N é o número total de indivíduos na comunidade.

14.2.6.2 Índice de Simpson

Proposto em 1949, por E. H Simpson ([SIMPSON, 1949](#)), O Índice de Simpson (D) é a medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma comunidade infinitamente grande, pertencerem à mesma espécie. O Índice é calculado conforme Equação 14.6,

$$D = 1 - \sum_{i=1}^s p_i^2 \quad (14.6)$$

onde, p_i é a proporção de indivíduos pertencentes a espécie i na comunidade, calculado com a equação 14.5, n_i o número de indivíduos na espécie i e N é o número total de indivíduos na comunidade. S é o total de espécies predominante na comunidade.

14.2.6.3 Índice de Margalef

Índice de Margalef (D) ([MARGALEF, 1969](#)), é usado na ecologia para estimar a biodiversidade de uma comunidade com base na distribuição numérica dos indivíduos das diferentes espécies em função do número total de indivíduos existentes na amostra analisada. O Índice é calculado conforme Equação [14.7](#),

$$D = \frac{S - 1}{\ln(N)} \quad (14.7)$$

onde, S é o número de espécies e $\ln(N)$ é o logaritmo do número de indivíduos pertencente a comunidade

14.2.6.4 Índice de McIntosh

O Índice de McIntosh foi proposto por ([MCINTOSH, 1967](#)). A comunidade pode ser vista como um ponto em um hiper volume S -dimensional e a distância euclidiana da comunidade para a origem pode ser utilizada como uma medida de diversidade. É uma medida de probabilidade de dois indivíduos selecionados aleatoriamente de uma comunidade infinitamente grande pertencerem a mesma espécie ([ARAÚJO et al., 2017](#)). O índice de diversidade de McIntosh (D) é estimado através das seguintes Equações [14.8](#),

$$D = \frac{N - U}{N - \sqrt{N}} \quad (14.8)$$

onde, N é o número total de indivíduos na amostra e U é dado pela expressão [14.9](#),

$$U = \sqrt{\sum n_i^2} \quad (14.9)$$

onde, n_i é o número de indivíduos na espécie i e o somatório é realizado sobre todas as espécies. U é a distância euclidiana da comunidade desde a origem quando plotada em um hiper volume S -dimensional.

14.2.7 Reconhecimento de Padrões

Entende-se por padrão as propriedades que possibilitam o agrupamento de objetos semelhantes dentro de uma determinada classe ou categoria, mediante a interpretação de dados de entrada, que permitam a extração das características relevantes desses objetos ([CASTRO; PRADO, 2002b](#)).

O reconhecimento de padrões é uma tarefa comumente utilizada tanto para sistemas computadorizados como pelo próprio ser humano. Muitas técnicas ilusionistas se baseiam no reconhecimento de padrões, assim como há várias aplicações computacionais que conseguem reconhecer vozes, imagens, vídeos, etc. Para [Wizbicki e Battisti \(2014\)](#), a base dessa técnica é classificar as informações baseando-se em um conhecimento inicial ou informações estatísticas dos próprios padrões. Quando falamos de reconhecimento de padrões gráficos em sistemas computacionais, estamos falando de visão computacional.

O uso do reconhecimento de padrões é recorrente para discriminar e classificar possíveis alterações em imagens médicas, determinando a presença ou ausência da doença. Após a extração de características, geralmente utiliza-se o reconhecimento de padrões para detectar as possíveis alterações nas imagens e classificar as possíveis anomalias ou não da imagem que desencadeiam a doença. Foi utilizado a ferramenta WEKA para realizar a classificação da pesquisa proposta.

Weka é uma coleção de algoritmos de aprendizado de máquina para tarefas de mineração de dados. Ele contém ferramentas para preparação de dados, classificação, regressão, agrupamento, mineração de regras de associação e visualização ([WITTEN et al., 2016](#)).

Para o reconhecimento de padrões desta metodologia, baseado em estudos sobre classificadores levando em consideração seu desempenho, facilidade de adaptação nos experimentos, eficácia e junção com as ferramentas de extração de características em imagens. Embasado nestes critérios, foram escolhidos os seguintes classificadores: *Random Forest*, *Sequential Minimal Optimization (SMO)*, *Multi Layer Perceptron (MLP)* e *Instance Based Learning (IBK)*.

O *Random Forest* é um classificador composto por uma coleção de árvores, onde são aleatoriamente independentes e identicamente distribuídas e cada árvore vota na classe mais popular, sendo a escolhida para o objeto ([OSHIRO, 2013b](#)).

O SMO é um dos principais métodos atuais para resolver o problema de programação quadrática que surge durante o treinamento de uma máquina de vetores de suporte (SVM). Este método, em certa medida, pode diminuir o grau de dificuldade de um problema de programação quadrática através de estratégias de decomposição. Esses problemas são resolvidos analiticamente, permitindo que o SMO manipule um grande conjunto de treinamento e realize a distinção dos padrões entre os conjuntos de dados avaliados (ZENG et al., 2008).

O MLP é uma rede neural *feedforward*, que consiste em uma rede neural artificial em que as conexões entre os nós não formam um ciclo. É composta em camadas na qual a informação flui unidirecionalmente da camada de entrada para a camada de saída, passando pelas camadas ocultas. Cada conexão entre neurônios tem seu próprio peso. Perceptrons para a mesma camada têm a mesma função de ativação. Em geral, é uma conexão para as camadas ocultas. Dependendo da aplicação, a camada de saída também pode ser uma conexão ou uma função linear (TAUD; MAS, 2018).

O IBK usa uma abordagem bastante simples de armazenar os dados de treinamento e classificar novos dados não vistos calculando a distância do exemplo mais semelhante já armazenado. O algoritmo usa o método de votação para decidir a classificação do novo exemplo e o número de votos é denotado pelo valor k onde, o valor de k precisa ser ímpar para evitar situações de empate (KANWAL; BOSTANCI, 2016).

14.2.8 Métricas de Validação

Para avaliar o desempenho dos classificadores, geralmente são usados métricas para validar a relevância da metodologia, onde são comumente utilizados as seguintes métricas: Acurácia (A), Sensibilidade (S) e Especificidade (E). Essas métricas fazem o uso da matriz de confusão, que faz a classificação correta e incorreta das classes, agrupando os resultados em quatro classes, respectivamente: Verdadeiro Positivo (VP), Verdadeiro Negativo (VN), Falso Positivo (FP) e Falso Negativo (FN).

- VP - Casos onde a método define como melanoma e o especialista confirma que é melanoma.
- FP - Casos onde a método define como melanoma, mas a marcação do especialista diz ser não melanoma.
- VN - Casos onde, tanto a marcação do método quanto a do especialista, define como não melanoma.
- FN - Casos onde o método diz se tratar de um não melanoma, mas o especialista define como melanoma.

A Tabela 94 apresenta a matriz de confusão.

Tabela 94 – Matriz de confusão

Resultado	Melanoma	Ausente
Positivo	Verdadeiro Positivo - VP	Falso Negativo - FN
Negativo	Falso Positivo - FP	Verdadeiro Negativo - VN

Acurácia (A): É a proporção de previsões corretas, sem levar em consideração o que é positivo e o que é negativo. Esta medida é altamente suscetível a desbalanceamentos do conjunto de dados e pode facilmente induzir a uma conclusão errada sobre o desempenho do sistema. Representado pela Equação 14.10.

$$A = \frac{VP + VN}{VP + VN + FP + FN} \quad (14.10)$$

Sensibilidade (S): É a proporção de verdadeiros positivos, isto é, a capacidade do sistema em predizer corretamente a condição para casos que realmente a têm. Representado pela Equação 14.11.

$$S = \frac{VP}{VP + FN} \quad (14.11)$$

Especificidade (E): É a proporção de verdadeiros negativos, isto é, a capacidade do sistema em predizer corretamente a ausência da condição para casos que realmente não a têm. Representado pela Equação 14.12.

$$E = \frac{VN}{VN + FP} \quad (14.12)$$

A análise ROC (*Receiver Operating Characteristic*) é uma ferramenta amplamente difundida para avaliar o desempenho de testes de diagnósticos na medicina. A AUC (*area under the ROC curve*) é uma maneira de resumir a curva ROC em um único valor, agregando todos os limiares da ROC. Como visto em Krohling, Paula e Behlau (2016), a curva ROC é uma ferramenta binária cujo valor varia de 0,0 até 1,0 e o limiar entre a classe é 0,5. Ou seja, acima desse limite, o algoritmo classifica em uma classe e abaixo na outra classe. Além disso, o AUC possui cinco graus de classificação: Excelente (0,9 a 1), boa (0,8 a 0,9), razoável (0,7 a 0,8), ruim (0,6 a 0,7) e não discriminatória (0,5 a 0,6).

Para avaliar a concordância do experimento é usado o método estatístico de cálculo do Coeficiente Kappa (K). Sendo utilizado para avaliar a reprodutibilidade dos resultados na validação. Este resultado quanto mais próximo de 1, maior será a concordância do experimento. A Tabela 95 representa os níveis de concordância do K.

Tabela 95 – Tabela de Concordância do Coeficiente Kappa.

Valor do Coeficiente KAPPA	Nível de Concordância
<0	Não existe concordância
Entre 0 e 0,2	Concordância Mínima
Entre 0,21 e 0,4	Concordância Razoável
Entre 0,41 e 0,6	Concordância Moderada
Entre 0,61 e 0,8	Concordância Substancial
Entre 0,81 e 1	Concordância Perfeita

No cálculo do K, P_o é a taxa de aceitação relativa e P_e é a taxa hipotética da aceitação. Representado pela Equação 14.13.

$$K = \frac{P_o - P_e}{1 + P_e} \quad (14.13)$$

As métricas citadas acima, tem como finalidade estimar o desempenho da metodologia realizada neste trabalho, para determinar se os resultados obtidos foram relevantes ou não e também apontar os pontos que foram positivos e negativos para que seja aplicado melhorias em trabalhos futuros para essa pesquisa.

14.3 Trabalhos Relacionados

A literatura especializada oferece estudos relacionados a classificação do melanoma, assim como técnicas de extração de características. Nesta seção será abordado um resumo de alguns desses trabalhos.

Na metodologia de Li e Shen (2018) é proposto uma abordagem para segmentar e classificar as imagens de lesões de pele de forma simultânea através da técnica *Lesion Indexing Network* (LIN). Através de 2 CNNs profundas, FCRNs-88, treinadas com diferentes conjuntos de treinamento, para produzir o resultado da segmentação e classificação. Usa-se a técnica *Lesion Indexing Calculation Unit* (LICU) para medir a importância do pixel na decisão da classificação. Resultando em 85,7% de acurácia, 96,6% em especificidade e 49,0% na sensibilidade.

Maia et al. (2019) propõe uma metodologia para o diagnóstico do melanoma através das regras ABCD (Assimetria, borda, cor e diâmetro), com o uso da rede neural VGG16, Inception e Resnet para a extração de características. Realizando o reconhecimento de padrões com o classificador Random Forest e obtendo uma acurácia de 92%, sensibilidade em 92,5% e especificidade de 91,8%.

no trabalho de Saba et al. (2019) é proposto uma abordagem automatizada para detecção e reconhecimento de lesões cutâneas usando uma *deep convolutional neural network* (DCNN). A DCNN possui três etapas para segmentação e reconhecimento de lesão: (a) aumento de dados e contraste aprimorado, (b) identificação

de lesão com base em valor de cor e extração de limite de lesão, e (c) extração, fusão e seleção de recursos para reconhecimento. Foi utilizado o classificador *Multi Layer Perceptron* para classificação e atingiu a melhor precisão de 98,4% de acurácia.

No trabalho de [Ünver e Ayan \(2019\)](#) mostra uma metodologia para segmentação de lesões cutâneas em imagens dermatoscópicas combinando o algoritmo de segmentação *GrabCut* e uma rede neural convolucional profunda (DCNN) denominada *You Only Look Once* (YOLO) para classificação. Foram obtidos os resultados de 92,99% de acurácia, 83,63% de sensibilidade e 94.02% de especificidade.

Na abordagem de [Adegun e Viriri \(2019\)](#) é proposto um método baseado no aprendizado profundo para a detecção e segmentação automática de lesões de melanoma, utilizando Rede Neural Convolutinal (RNC) para extração de características e o classificador *softmax* para classificação das lesões. O método alcançou uma precisão de 95% de acurácia, 97% de sensibilidade e 96% de especificidade.

[Banerjee et al. \(2020\)](#) propõe um algoritmo baseado em aprendizagem profunda *You Only Look Once* (YOLO), que é baseado na aplicação de *deep convolutional neural network* (DCNN) para detectar melanoma a partir de imagens dermatoscópicas e digitais, aplicando algoritmo baseado em grafos para segmentação e regra do ABCD para extração de características. Resultando em 97,5% de acurácia, especificidade e sensibilidade.

Na metodologia de [Kumar et al. \(2020\)](#) propõe um sistema que classifica uma imagem de lesão da pele como cancerosa ou não cancerosa utilizando o agrupamento *fuzzy C-means* para segmentação. Para classificação uma a rede neural artificial (RNA) é treinada usando o algoritmo de evolução diferencial (DE). O resultado mostra que a técnica proposta detecta efetivamente o câncer de pele e apresenta acurácia de 97,4% de acurácia, 92% de especificidade e 90% de sensibilidade.

Os trabalhos relacionados acima apresentaram resultados promissores na detecção e diagnóstico do melanoma e lesões de pele, porém alguns apresentaram dificuldades na execução do método. Como no trabalho de [Saba et al. \(2019\)](#), que poderia ter sido aplicado outras métricas para validar suas pesquisas. No projetos de [Li e Shen \(2018\)](#) e [Ünver e Ayan \(2019\)](#) poderia ser empregado técnicas para melhorar a sensibilidade.

A maioria dos trabalhos citados acima, utilizam descritores baseados em forma, seguindo as regras ABCD e descritores baseados em redes neurais, com isso, nesta pesquisa pretende-se utilizar uma nova abordagem com descritores de textura baseado em índices de diversidade para verificar a relevância da análise de textura em comparação aos descritores de forma e aos baseados em redes neurais, também aumentar a acurácia e melhorar os valores da sensibilidade e especificidade.

14.4 Metodologia

A metodologia proposta neste trabalho segue as etapas da Figura 104. Na primeira etapa, é feita a aquisição das imagens oriundas da base PH2 ([MENDONCA et al., 2015](#)). A segunda etapa, é o pré-processamento, onde técnicas de processamento de imagens são aplicadas de modo a evidenciar com mais destaque as estruturas contidas nas imagens. Na terceira etapa, é realizada a extração de características, através dos índices de diversidade. Na quarta etapa, a classificação em melanoma ou não melanoma, foram utilizados os classificadores *Random Forest*, SMO, MLP e IBK. O processo é finalizado com a validação dos resultados.

14.4.1 Aquisição das Imagens

O banco de dados PH2 foi criado principalmente para disponibilizar um conjunto de dados comum que pode ser usado para a avaliação de desempenho de diferentes sistemas de diagnóstico auxiliado por computador de imagens dermatoscópicas ([MENDONCA et al., 2015](#)). A base possui 200 imagens dermatoscópicas divididas em 40 images de melanoma e 160 imagens de não melanoma, com 8bits e resolução de 768x560 pixels.

Este trabalho utiliza *Region of Interest* (ROI) adquiridas da base PH2. Para a concretização desse trabalho, foram utilizadas 200 ROIs de imagens de lesões na pele; sendo 160 ROIs sem a presença de melanoma e 40 ROIs com melanoma. Na Figura 105 temos duas amostras da base.

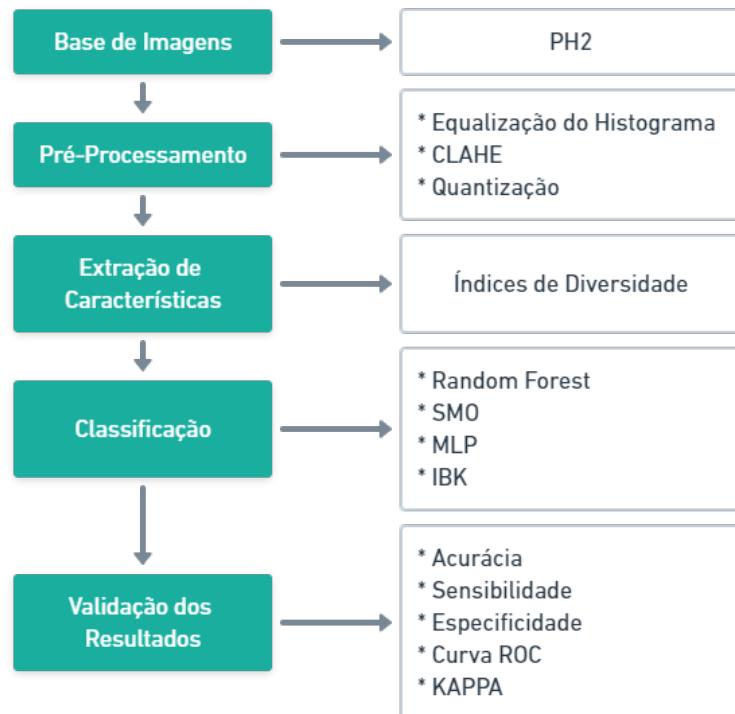


Figura 104 – Metodologia proposta.

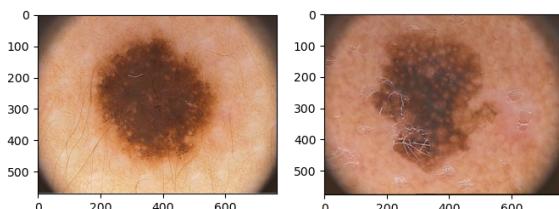


Figura 105 – Exemplos de imagens contidas na base PH2.

14.4.2 Pré-Processamento

Na metodologia utiliza-se técnica de Equalização de Histograma para um espalhamento da distribuição dos níveis de cinza ao longo de toda a escala de contraste. Tal técnica tem a finalidade de aumentar a detectabilidade de aspectos da imagem. Na Figura 106 vemos a imagem original (a) e a imagem com a técnica aplicada (b). Na Figura 107 vemos seus respectivos histogramas.

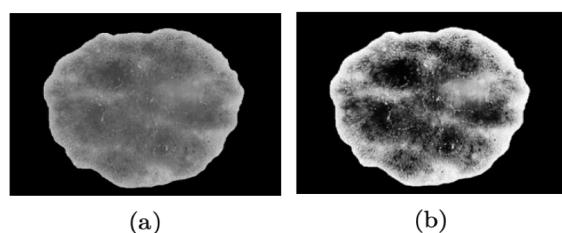


Figura 106 – (a) Imagem Original e (b) Imagem Equalizada.

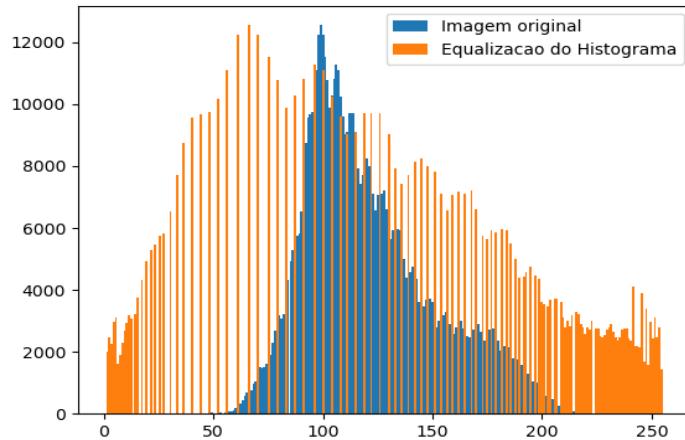


Figura 107 – Histograma da Imagem Original e da Imagem Equalizada.

A técnica de equalização adaptativa de histograma com limitação de contraste (CLAHE) foi utilizada para gerar uma nova representação da imagem com realce de contraste. Essa técnica separa a imagem em regiões contextuais e aplica a equalização de histograma em cada uma delas. Na Figura 108 vemos a imagem original (a) e a imagem com a técnica aplicada (b). Na Figura 109 vemos seus respectivos histogramas.

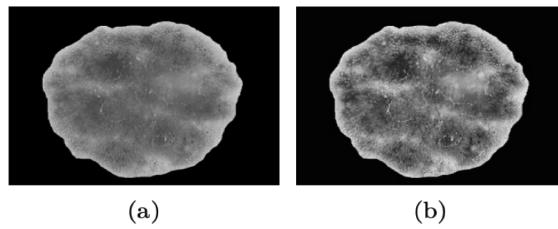


Figura 108 – (a) Imagem Original e (b) Imagem Realçada.

A metodologia utiliza de quantização (Figura 110). A proposta desta abordagem é investigar se uma região com menos espécies produz um resultado mais homogêneo e, desta forma, auxiliar os índices de diversidade. Neste trabalho, os valores de 7, 6, 5 e 4 bits foram definidos para aplicação da técnica, com o objetivo de obter uma variância gradual. As imagens foram geradas a partir da divisão dos *pixels* em subgrupos, onde, os elementos de cada subgrupo assumem o valor do *pixel* central, tornando a população mais homogênea.

14.4.3 Extração de Características

Após o pré-processamento das imagens, inicia-se a fase de extração de características baseada em textura. Nesta etapa serão obtidos valores que caracterizam a textura da lesão na pele. Neste trabalho, são utilizados quatro índices de diversidade, sendo eles: índice de Margalef, McIntosh, Shannon e Simpson.

Para a utilização destes índices em imagens, é necessário um mapeamento de conceitos de tal forma, que uma comunidade passa a ser interpretada como uma imagem e as espécies são as intensidades de

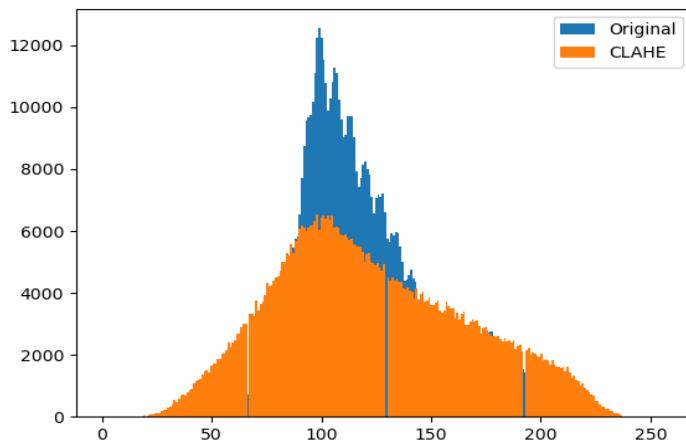


Figura 109 – Histograma da Imagem Original e da Imagem Realçada.

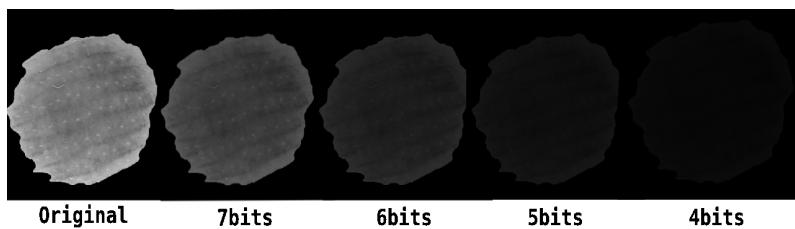


Figura 110 – Quantizações aplicadas na metodologia.

pixels presentes na imagem (REIS et al., 2012).

De posse desse mapeamento de conceitos, é aplicada segmentação para evidenciar a região de interesse (ROI), onde foram obtidos os índices de diversidade, criando um vetor de características. A ROI cobre regiões significativas da imagem que, para esse trabalho, trata-se da região da lesão na pele. Para a geração do vetor de características, foram calculados todos os índices para a ROI selecionada gerando um vetor de 4 elementos, um para cada índice de diversidade.

Por fim, esses índices descrevem as texturas das imagens e os atributos fornecidos buscam reconhecer as medidas de homogeneidade ou heterogeneidade da ROI, servindo como parâmetros de entrada para os classificadores utilizados e permitindo a categorização em melanoma e não melanoma.

14.4.4 Classificação e Validação dos Resultados

Para a classificação, foi utilizado a ferramenta suíte de algoritmos de mineração de dados e Aprendizado de Máquina WEKA, que contém algoritmos para pré-processamento de dados, classificação, regressão, agrupamento, regras de associação e visualização.

Foram selecionados os algoritmos *Random Forest*, SMO, MLP e IBK, por apresentarem bons resultados e pelo fato de serem rigorosos na presença de ruídos no processo de classificação, também utilizam descritores que se baseiam em comportamento dentro de comunidades, sendo uma boa escolha para a classificação da metodologia.

Utilizando os classificadores com parametrização padrão juntamente com a validação cruzada (*Cross Validation*) de k -folds, atribuindo o $k = 10$. Esse método tem como procedimento dividir os indivíduos em 10 grupos, separando o treino em 9 grupos e 1 grupo para teste, ocorrendo 10 cruzamentos alterando o grupo de teste, após realizar todos os cruzamentos é gerado uma média atribuída como resultado.

Por fim, após a etapa do reconhecimento de padrões é necessário a utilização de métricas para validar os resultados e analisar possíveis melhorias do mesmo. A metodologia proposta, utiliza métricas de validação recorrentes na área de processamento digital de imagens, sendo baseadas em estatísticas, respectivamente: Acurácia (A), Sensibilidade (S), Especificidade (E), Coeficiente Kappa (K) e Área Sobre a Curva (ROC).

14.5 Resultados e Discussões

Este capítulo apresenta e discute os resultados obtidos da aplicação da metodologia proposta neste trabalho, obtendo o resultado da classificação das ROIs das imagens de pele da base PH2 em melanoma e não melanoma. Foram feitos testes dos descritores de características para imagens em níveis de cinza, do canal vermelho, do canal verde e do canal azul, com o intuito de verificar o potencial dos descritores de textura em cada uma dessas representações da imagem. Para cada imagem são calculados quatro atributos, para todos os experimentos.

As Tabelas apresentadas neste Capítulo, são uma síntese com os melhores resultados obtidos entre os classificadores *Random Forest*, IBK, MLP e SMO nesta metodologia, através da extração de características de textura realizadas a partir dos índices de diversidade especificados na Seção 14.2.6 e utilizando os classificadores descritos na Seção 14.2.7 para o reconhecimento de padrões das imagens de pele.

A Tabela 96 apresenta os resultados dos testes para as imagens originais, com e sem quantização.

Sem Quantização						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	78	90	30	0,717	0.2254	Random Forest
Canal Vermelho	81,5	93,13	35	0,750	0.3273	Random Forest
Canal Azul	75,5	84,38	40	0,622	0.2415	IBK
Canal Verde	77	86,25	40	0,631	0.2675	IBK
Com Quantização de 7 bits						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	78	95	10	0,563	0,0678	Random Forest
Canal Vermelho	79	90	35	0,778	0,2759	Random Forest
Canal Azul	71,5	83,13	25	0,541	0,0836	IBK
Canal Verde	77	93,13	12,5	0,658	0,0726	Random Forest

Tabela 96 – Resultado dos testes utilizando imagens sem nenhum melhoramento de contraste.

De acordo com a Tabela 96, o classificador *Random Forest* obteve o melhor resultado em imagens com canal vermelho e sem quantização, com uma taxa de acurácia de 81,5%, uma sensibilidade de 93,13%, especificidade de 35%, uma curva ROC de 0,750 e um Kappa de 0,3273, indicando que o classificador apresenta uma razoável capacidade de diagnóstico, de acordo com a Tabela 95. O resultado menos significativo foi do classificador IBK nas imagens comuns com canal azul e com quantização, com taxa de acurácia de 71,5%, curva ROC de 0,541 e Kappa de 0,0836. Como podemos ver na Tabela 96, o classificador IBK foi capaz de identificar um número razoável de casos com a doença, como mostra a sensibilidade de 83,13%. Um número baixo de casos que não têm a doença, com especificidade de 25%.

A Tabela 97 apresenta os resultados dos testes para as imagens com Equalização do histograma, com e sem quantização.

Dos resultados exibidos na Tabela 97, o classificador *Random Forest* em conjunto com as imagens com canal azul e com quantização obteve o melhor resultado, com uma taxa de acurácia de 88%, uma sensibilidade de 95%, especificidade de 60%, uma curva ROC de 0,816 e Kappa de 0,5946, indicando que o classificador apresenta uma moderada capacidade de diagnóstico, de acordo com a Tabela 95. O classificador MLP, tanto para as imagens em níveis de cinza quanto para o canal verde, sem quantização, obtiveram o resultado menos significativo, com uma taxa de acurácia de 86%, curva ROC 0,885 e Kappa de 0,4737. Foi observado na Tabela 97, que o classificador foi capaz de identificar um número alto de casos com doença, como mostra a sensibilidade de 86%, porém, uma quantidade mediana de casos que não têm a doença, com especificidade de 42,5%.

Sem Quantização						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	86	96,88	42,5	0.885	0.4737	MLP
Canal Vermelho	85	92,5	55	0.807	0.5033	Random Forest
Canal Azul	87,5	95,63	55	0.887	0.5645	MLP
Canal Verde	86	96,88	42,5	0.893	0.4737	MLP
Com Quantização de 7 bits						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	88	96,25	55	0.873	0.5775	MLP
Canal Vermelho	87	95	55	0.883	0.5517	MLP
Canal Azul	88	95	60	0.816	0.5946	Random Forest
Canal Verde	86,5	95,63	50	0.87	0.5196	MLP

Tabela 97 – Resultado dos testes utilizando imagens realçadas com Equalização do Histograma.

Imagens Originais e CLAHE						
Sem Quantização						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	89,5	96,88	60	0.784	0.6341	SMO
Canal Vermelho	90	96,88	62,5	0.797	0.6552	SMO
Canal Azul	89	96,25	60	0.781	0.6207	SMO
Canal Verde	89,5	96,88	60	0.784	0.6341	SMO
Com Quantização de 7 bits						
Espaço de Análise	A(%)	S(%)	E(%)	ROC	Kappa	Classificador
Níveis de cinza	90,5%	97,5%	62,5%	0.800	0.669	SMO
Canal Vermelho	90%	97,5%	60%	0.788	0.6479	SMO
Canal Azul	90%	96,88%	62,5%	0.797	0.6552	SMO
Canal Verde	90%	97,5%	60%	0.788	0.6479	SMO

Tabela 98 – Resultado dos testes utilizando imagens realçadas com CLAHE.

A Tabela 98 demonstra os resultados para as imagens com equalização de CLAHE, com e sem quantização. Entre os resultados obtidos, o classificador SMO obteve o melhor resultado nas imagens em níveis de cinza e com equalização, com uma taxa de acurácia de 90,5%, uma sensibilidade de 96,88%, especificidade de 62,5%, uma curva ROC de 0,800 e uma Kappa de 0,669, indicando que o classificador apresenta uma capacidade de diagnóstico substancial, de acordo com a Tabela 95. O pior resultado foi do classificador SMO nas imagens com canal azul e sem quantização, com taxa de acurácia de 89%, curva ROC de 0,781 e Kappa de 0,6207. Como podemos observar na Tabela 98, o classificador identificou um número alto de casos com doença, como mostra a sensibilidade de 89% e uma quantidade razoável de casos sem a doença, com especificidade de 60%.

A Tabela 99 apresenta uma comparação da metodologia deste trabalho com os trabalhos relacionados citados no Capítulo 14.3.

A Tabela 99 faz um comparativo entre a metodologia proposta e os trabalhos relacionados, avaliando a acurácia, especificidade e sensibilidade. Este comparativo visa apenas ilustrar as previsões obtidas em cada metodologia. No que diz respeito às métricas de desempenho utilizadas neste trabalho, o ideal para um sistema CADx é ter um bom equilíbrio entre as três métricas de avaliação (acurácia, sensibilidade e especificidade), pois, um bom método deve ser capaz de classificar com sucesso, tanto os casos de positivos (que possuem a doença) como os casos de negativos (que não possuem a doença).

É importante destacar a capacidade do método proposto em classificar corretamente as imagens que possuem melanoma, ou seja, alcançando uma sensibilidade de 97,5%.

Os resultados da metodologia proposta mostram-se promissores em comparação aos demais trabalhos.

Tabela 99 – Comparativo com os Trabalhos Relacionados.

Trabalho	Metodologia	Base	A(%)	S(%)	E(%)
(LI; SHEN, 2018)	LIN CNN	ISIC	85,7	49	96,6
(MAIA et al., 2019)	Inception + Resnet + Random Forest	PH2	92	91,8	92,5
(ÜNVER; AYAN, 2019)	GrabCut + YOLO	PH2	92,99	83,63	94,02
(SABA et al., 2019)	DCNN + MLP	PH2	95,4	-	-
(ADEGUN; VIRIRI, 2019)	RNC + Softmax	PH2	95	97	95
(KUMAR et al., 2020)	FUZZY + ANN-DE	PH2	97,4	90	92
(BANERJEE et al., 2020)	ABCD + YOLO	PH2	97,5	97,5	97,5
Metodologia	Índices de diversidade + Quantização + SMO	PH2	90,5	97,5	62,5

Na metodologia de [Li e Shen \(2018\)](#) foi obtido uma acurácia de 85,7%, sendo a menor taxa entre os trabalhos relacionados. O trabalho de [Banerjee et al. \(2020\)](#) obteve as melhores valoress preditivos entre os trabalhos relacionados. Por fim, baseado nos valores das predições, a metodologia proposta apresentou resultados satisfatórios diante da bibliografia.

14.5.1 Discussões

O método proposto apresentou-se promissor, no entanto, com algumas limitações em relação a outros trabalhos descritos no Capítulo 14.3. No entanto, é importante ponderar alguns aspectos, a saber:

1. Foi proposto e apresentado neste trabalho, técnicas capazes de descrever a textura das imagens, a partir de técnicas de outras áreas de estudo;
2. Os índices propostos nesse trabalho não necessitam de parametrização inicial e nem é necessário o redimensionamento das imagens, assim como, nos métodos utilizando CNN.
3. O método proposto é rápido, e consegue caracterizar cada imagem com uma média de 20 milésimos de segundo;
4. O método proposto produz apenas 4 (quatro) características para cada imagem, e conseguiu com isso, resultados bem promissores, uma vez que, descritores baseados em CNNs produzem milhares de atributos;
5. Por fim, acredita-se que os descritores propostos são robustos e capazes de caracterizar com sucesso, as imagens de melanoma e não-melanoma;

14.6 Conclusão

O presente estudo apresentou um método automático, com o uso dos Índices de Diversidade, em conjunto com reconhecimento de padrões, capaz de discriminar e classificar lesões na pele em melanoma ou não.

Os índices de diversidade, apresentaram-se moderados na tarefa de caracterização das regiões em melanoma e não melanoma.

Os resultados obtidos nas imagens quantizadas, demonstraram o desempenho próspero das técnicas de extração de textura pelos índices de diversidade com o classificador SMO, sendo confirmados pelo resultados da estatística *Kappa*, que é representado na Tabela 95.

No entanto, este trabalho contribuiu de forma positiva em duas distintas áreas. Sendo que na área da saúde, disponibilizou uma metodologia automática para auxílio no diagnóstico lesões na pele em melanoma e não melanoma, que apesar de não fornecer excelentes resultados mostram-se promissores. E na computação, com a adaptação de técnicas de outras áreas do conhecimento que serviram como descritores de textura, permitindo então, que sejam utilizados para caracterizações em outros tipos de imagens.

Ademais, como sugestão para trabalhos futuros, pretendendo-se:

- Implementar novos testes buscando melhorar ainda mais os resultados;
- Aumentar a quantidade e variabilidade das amostras de lesões utilizando mais bases de imagens;
- Aplicar os índices propostos para investigação de outras doenças; e,
- Aplicar os índices propostos em conjunto com outros descritores, inclusive, com abordagens de aprendizado profundo.

Segmentação de Nódulos Pulmonares em Imagens de Tomografia Computadorizada

Autores: Açucena Rodrigues dos Santos Soares; Flávio Henrique Duarte de Araújo.

Resumo: O câncer de pulmão é uma das principais causas de morte por câncer em todo o mundo. O diagnóstico precoce aumenta as chances de cura do paciente. O principal exame para o diagnóstico da doença é a tomografia computadorizada, um exame de imagem que fornece informações precisas sobre o nódulo. No entanto, depende de muitos fatores externos, desde a qualidade do equipamento até a fadiga do especialista que analisa. Técnicas de processamento de imagem podem ser grandes aliadas na detecção precoce do nódulo, uma vez que não tem limitações humanas. Este estudo apresenta uma avaliação de duas abordagens de aprendizagem profunda, 3D U-Net e 3D V-Net, com diferentes configurações de arquiteturas, parâmetros e distribuição de aumento de dados aplicados à segmentação dos nódulos pulmonares. Os melhores resultados obtidos correspondem a Dice Score de 0.846 e Mean IoU 0.499 para 3D U-Net e Dice 0.685 e Mean IoU 0.987 para a V-Net. A primeira rede obteve os melhores resultados para as imagens de menor dimensão porque há uma maior delimitação das regiões de interesse, removendo elementos que possam influenciar negativamente no resultado.

Keywords: 3D U-Net. V-Net; Segmentação de nódulos pulmonares; Imagens Médicas; Câncer de Pulmão; LUNA16 Database.

15.1 Introdução

Nas últimas décadas, o câncer de pulmão tem evoluído do status de doença rara para o de maior causa de mortes por câncer no mundo e isso se deve, principalmente, ao estilo de vida da população. Dentre as principais causas da doença encontram-se o tabagismo, a poluição do ar e fatores genéticos. Nos EUA, entre os anos de 1991 e 2018, houve um aumento estimado de mais de 70 mil casos de câncer (GROOT et al., 2018).

Segundo dados do GLOBOCAN 2018, que é um projeto da *International Agency for Research on Cancer* (IARC) para a avaliação de incidência e mortalidade por câncer no mundo, o câncer de pulmão foi o tipo de câncer com maior incidência no ano. Neste ano, ocorreram pouco mais de 2,09 milhões de casos da doença (correspondente a 11,6%) e a maior causa de mortes com mais de 2,76 milhões de mortes (BRAY et al., 2018b). No Brasil, no ano de 2015, foi responsável por 28.495 mortes e é o segundo tipo mais comum (INCA, 2019a).

O Diagnóstico precoce da doença aumenta as chances de sobrevivência. Dentre os vários métodos para o diagnóstico de câncer de pulmão está a tomografia computadorizada (TC) que se trata de um exame de imagem que permite a visualização de informações precisas como a localização e o tamanho do nódulo. Entretanto, necessita da análise de um especialista em um processo demorado e cansativo, o que pode levar a um diagnóstico errado.

Muitos pesquisadores têm se empenhado nos últimos anos em busca de formas de melhorar resultados e reduzir o número de falsos positivos. Um método muito trabalhado atualmente utiliza técnicas de Processamento Digital de Imagens (PDI) e visão computacional para a detecção e classificação de nódulos pulmonares. Essas técnicas possuem alta taxa de acurácia, além disso, são de baixo custo. Algoritmos de PDI compreendem três etapas principais, sendo elas: segmentação, extração de características e classificação.

A segmentação é a primeira fase da visão computacional, ou seja, é a porta de entrada para o total entendimento (ou processamento) de uma imagem. Trata-se de um processo que separa a imagem entre suas partes constituintes ou seus objetos. É uma fase muito importante para a visão computacional, uma má segmentação leva a um resultado final quase sempre ruim, pois, o resultado dessa etapa serve como base para as etapas de extração de características e de classificação (GONZALEZ; WOODS, 2000).

Com base no acima exposto, este estudo visa propor um método de segmentação automática dos nódulos pulmonares em imagens de TC do tórax baseado em técnicas de segmentação disponíveis na literatura. Como objetivos específicos temos:

1. Levantamento bibliográfico das principais técnicas que estão sendo utilizadas na área do trabalho.
2. Realização de uma análise comparativa entre métodos já desenvolvidos.
3. Desenvolvimento de um método para a segmentação de nódulos pulmonares em imagens de TC de tórax.
4. Estimar o desempenho do método desenvolvido utilizando métricas de avaliação.

As seções deste trabalho estão dispostas da seguinte forma: na Seção 15.2 são apresentados conceitos importantes ao desenvolvimento e compreensão do trabalho bem como as técnicas e as métricas utilizadas. Já a Seção 15.3 apresenta, de forma breve, descrições dos principais trabalhos relacionados encontrados na literatura. Logo em seguida, a Seção 15.4 aborda as principais características deste trabalho, tais as principais abordagens para cada etapa do desenvolvimento, resultados alcançados e outras informações úteis a respeito da metodologia aqui implementada. Por fim, na Seção 15.5 são apresentadas as conclusões deste trabalho, bem como possíveis trabalhos futuros.

15.2 Referencial Teórico

Este capítulo apresenta o referencial teórico que serve como base para o total entendimento do conteúdo abordado neste trabalho. Inicialmente, são apresentados alguns conceitos acerca do câncer de pulmão e em seguida, tratamos sobre segmentação de imagens. Além desses temas, são abordadas algumas técnicas encontradas na literatura que estão sendo investigadas para uso neste trabalho.

15.2.1 Câncer de Pulmão

O Câncer de pulmão é a principal causa de mortes por câncer no mundo. Assim como outros tipos da doença, ela se caracteriza pelo crescimento descontrolado de células que sofreram algum tipo de mutação genética ao serem expostas aos fatores de risco. Essas células se agrupam formando um nódulo que possui características próprias, diferentes dos demais tecidos (HOSPITAL DE CÂNCER DE BARRETOS, 2016).

Existem diferentes tipos de câncer que podem acometer os pulmões, entretanto, quase sempre o câncer de pulmão é um carcinoma, que por sua vez, pode ser subdividido em dois tipos, são eles: carcinoma de pequenas células e de não pequenas células, sendo o segundo o subtipo mais comum. O diagnóstico da doença em estágios iniciais aumenta as chances de cura do paciente (HOSPITAL DE CÂNCER DE BARRETOS, 2016).

Uma das formas para o diagnóstico é o exame de tomografia computadorizada (TC) do tórax. Trata-se de um exame de imagem que traz uma visualização completa, em 3D, da região dos pulmões, sendo possível identificar a localização e o tamanho de um possível nódulo. Para tal, é necessária a análise de um médico especialista que possui algumas limitações inerentes ao ser humano, o que pode influenciar no diagnóstico. A Figura 111 mostra um exemplo de uma imagem de tomografia computadorizada de tórax.

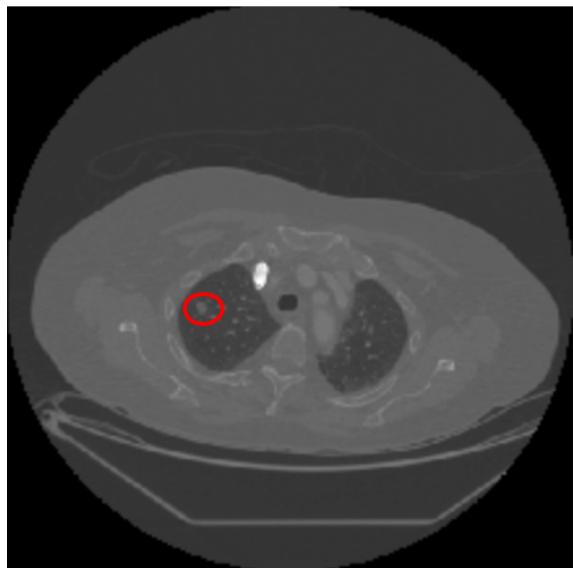


Figura 111 – Imagem de TC presente na base LUNA 16, o círculo vermelho representa um nódulo.

15.2.2 Segmentação

A segmentação é uma das principais etapas em sistemas de visão computacional. Trata-se da porta de entrada para a total compreensão (ou processamento) de uma imagem. Essa etapa consiste na separação de regiões de interesse de uma imagem original. A Figura 112 exemplifica a etapa de segmentação. É uma das fases mais importantes para a visão computacional, pois, os resultados das etapas de extração de características e de classificação, que usam como base os resultados dessa etapa, dependem de uma boa segmentação. (GONZALEZ; WOODS, 2000).

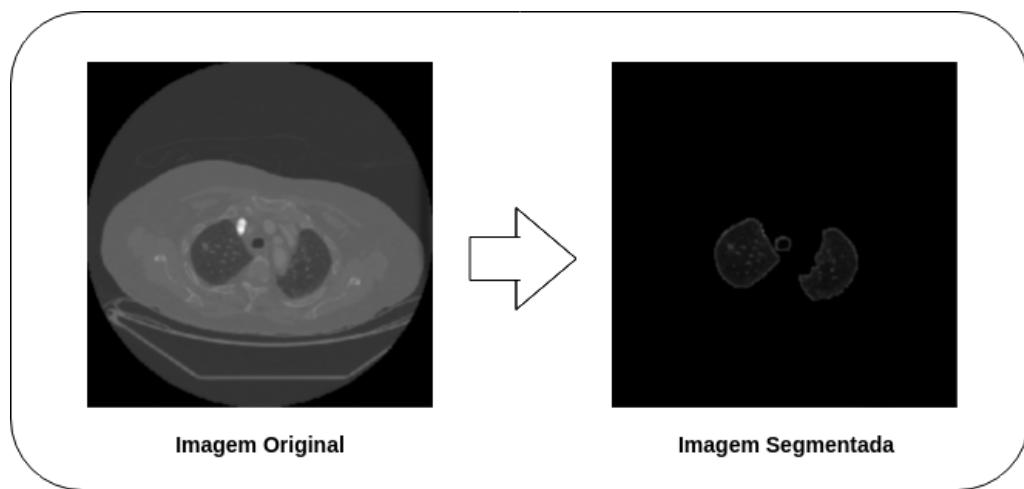


Figura 112 – Resultado de um processo de segmentação.

Uma boa segmentação é aquela em que os pixels da região de cada categoria possuem valores de escala de cinza similares, formando uma região conectada. Regiões vizinhas que não fazem parte do objeto possuem valores de pixels dissemelhantes. Ou seja, é uma segmentação em que os objetos podem ser claramente identificados dentro da imagem (GLASBEY; HORGAN, 1998).

Existem vários algoritmos para a segmentação de imagens, o mais comum e simples dentre eles é a limiarização que divide uma imagem em níveis de cinza em diferentes regiões com base em um limiar. A limiarização pode ser global, que divide a imagem em região alvo e fundo ou, local, que separa a imagem em várias regiões de alvo e fundo com base em diferentes limiares (YUHENG; HAO, 2017). Outra categoria de algoritmos de segmentação é a baseada em agrupamento, sendo o principal representante o *K Means*.

15.2.3 K-Means

O *K Means* (ARTHUR; VASSILVITSKII, 2007) é um algoritmo de aprendizado não supervisionado para o agrupamento de dados. Um algoritmo de agrupamento geralmente utiliza um conceito de similaridade que, por sua vez, realiza cálculos de medidas de distância de atributos para encontrar a similaridade de um ponto com os demais elementos do conjunto de dados.

Esse algoritmo tem por objetivo o agrupamento dos dados em k grupos definidos com base na similaridade dos seus atributos, para tal, atribui aleatoriamente k valores, chamados de centróides, que são os valores centrais de cada um dos clusters gerados e servem como base para a separação inicial dos dados. Essa separação é feita através do cálculo da distância euclidiana dos atributos entre os centróides e cada um dos pixels da imagem. Após a separação, recalcula o valor do centróide e repete o processo até que os clusters se tornem estáticos ou que um valor máximo de iterações seja atingido.

A aplicação do *K Means* permite a visualização de regiões separadas da imagem, permitindo a identificação de possíveis regiões de nódulos nas imagens. A Figura 113 mostra um exemplo de uma máscara gerada utilizando *K Means*. Além dos algoritmos de limiarização e de agrupamento, outro tipo muito utilizado tanto para segmentação como para a extração de características e classificação, são as Redes Neurais Convolucionais (CNNs).

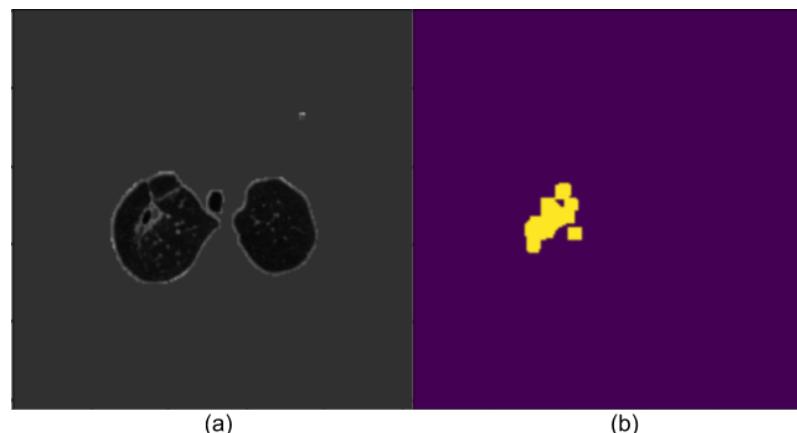


Figura 113 – Exemplo de utilização do *K Means*: (a) Imagem original e (b) Clusters obtidos após a aplicação do *K Means*.

15.2.4 Redes Neurais Convolucionais

As Redes Neurais Convolucionais são algoritmos em camadas com funções específicas que utilizam *deep learning*. Elas se diferem dos tradicionais algoritmos de *Machine Learning* por possuir métodos que aprendem através de uma composição de funções, onde a saída de uma camada é a entrada para a camada seguinte, em busca de uma função que melhor represente um conjunto de dados (PONTI; COSTA, 2018).

Essas redes são caracterizadas pela utilização de pelo menos uma camada convolucional. A Figura 114 mostra a estrutura básica de uma CNN.

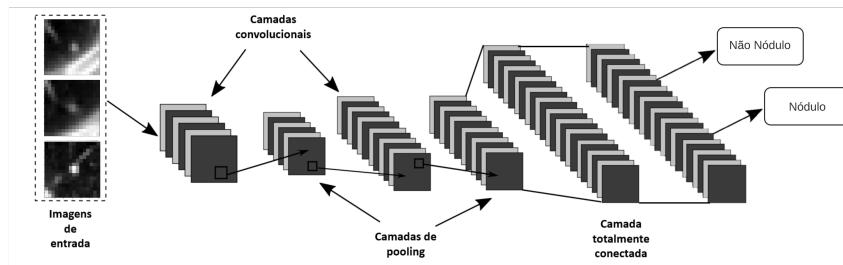


Figura 114 – Ilustração da arquitetura de uma LeNet que classifica imagens de entradas em nódulo ou não nódulo e suas três principais camadas: convolucionais, de pooling e totalmente conectadas. Fonte: adaptado de (ARAÚJO et al., 2017a).

15.2.5 3D U-Net

A 3D U-Net (ÇIÇEK et al., 2016) é uma adaptação da U-Net (RONNEBERGER; FISCHER; BROX, 2015) e é uma CNN com uma arquitetura em forma de U composta por dois caminhos. O caminho de contração (ou codificador) é responsável pela redução da imagem com aumento de volume, captando informação de contexto enquanto o caminho de expansão (ou decodificador) recupera a imagem ao seu tamanho original, reduzindo o volume e recuperando informação da região segmentada. A Figura 115 mostra a arquitetura da rede.

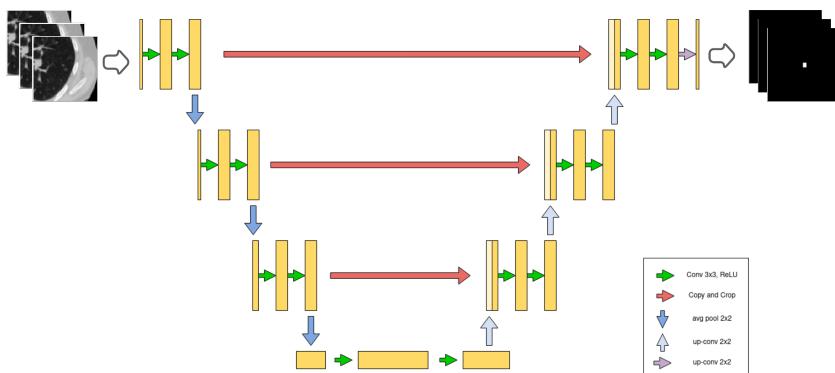


Figura 115 – A arquitetura da 3D U-Net.

15.2.6 V-Net

Semelhante à U-Net, citada acima, a V-Net (MILLETARI; NAVAB; AHMADI, 2016) é uma CNN composta por caminhos de contração e expansão, contudo, em cada camada, a rede aprende uma função residual que garante uma melhor convergência, diferente de uma rede não-residual como a U-Net. A Figura 116 mostra a arquitetura da V-Net.

15.2.7 Métricas de Avaliação

Para a avaliação da metodologia proposta, utilizaremos duas métricas amplamente utilizadas na literatura para avaliar o desempenho de algoritmos de segmentação, sendo elas: Coeficiente Dice (D) e Intersection over Union (IoU).

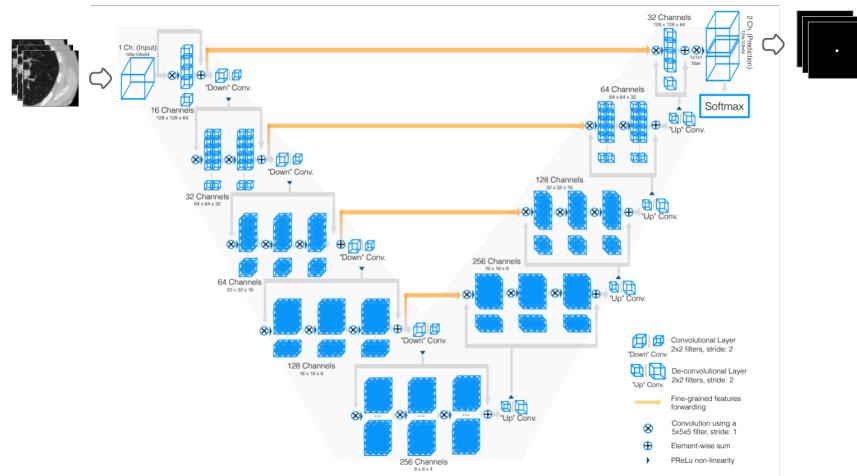


Figura 116 – Arquitetura da V-Net. Fonte: adaptado de ([MILLETARI; NAVAB; AHMADI, 2016](#)).

15.2.7.1 Coeficiente Dice

O Coeficiente Dice é uma métrica de avaliação que calcula a similaridade entre uma imagem segmentada (IS) e a sua máscara correspondente (M). Ela consiste na razão entre o número de verdadeiros positivos (VP) (pixel de valor 1 que foi predito como 1) dividido pela soma dos pixels da imagem segmentada com os pixels da máscara e multiplicado por 2. A Equação 15.1 mostra como essa métrica é calculada.

$$D = 2 * \frac{|VP|}{|IS| + |M|}. \quad (15.1)$$

15.2.7.2 Intersection over Union

IoU é uma métrica utilizada para medir a precisão dos algoritmos de segmentação de imagem que consiste na razão entre a área de intersecção e união da região da máscara (M) e a imagem segmentada (IS). A Equação 15.2 mostra a fórmula utilizada nesta métrica.

$$IoU = \frac{M \cap IS}{M \cup IS}. \quad (15.2)$$

15.3 Trabalhos Relacionados

Nos parágrafos abaixo são descritos alguns trabalhos relacionados ao tema em estudo, tratando principalmente de métodos de segmentação utilizando processamento de imagens e redes neurais convolucionais. A Tabela 15.3 traz um resumo dos trabalhos aqui apresentados.

Tabela 100 – Trabalhos Relacionados

Trabalho	Método Utilizado	Resultados
(GU et al., 2018)	Otsu, region growing, DBSCAN e CNN	Sensibilidade 87.94 - 92.93%
(KIDO; HIRANO; HASHIMOTO, 2018)	AlexNet, SVM e R-CNN	Acurácia 81.1 - 84.7%
(NÓBREGA et al., 2018)	CNN	Acurácia 89.91%
(FILHO et al., 2017a)	OTSU	Acurácia 85.3%
(ARESTA et al., 2019b)	iW-Net, baseada na U-Net	IoU 0.55
(SINGADKAR et al., 2020)	Abordagem 2D, CNN residual	Dice Score 0.9497 Jaccard 0.8868

O trabalho de (GU et al., 2018) propõe um método de segmentação dividido em duas etapas. Na primeira etapa, o processo é iniciado utilizando limiarização de Otsu (*Otsu's threshold*) para segmentar apenas a

parte da imagem correspondente aos pulmões, em seguida, utilizaram uma tecnologia de crescimento da região 3D (*3D region growing*) para remover a traqueia e os brônquios da imagem. Na segunda etapa, regiões candidatas a presença de nódulos são separadas em cubos multi-escala, esses cubos são agrupados utilizando o algoritmo DBSCAN. Os cubos passam por um processo de *data augmentation* e são classificados entre nódulo e outros tecidos utilizando CNN baseada no modelo C3D e no modelo de Julian de Wit para a detecção e classificação de nódulos. O método é avaliado utilizando a métrica de avaliação de performance FROC (free-response receiver operating characteristic) que analisa a sensibilidade por número de falsos positivos. O trabalho obteve sensibilidade de 87.94% e 92.93%.

(KIDO; HIRANO; HASHIMOTO, 2018) propõem um sistema de diagnóstico auxiliado por computador (CADx) baseado em imagens utilizando CNN, em uma abordagem que não utiliza uma prévia extração de características para o diagnóstico de anormalidade em tomografias computadorizadas de tórax. A própria CNN é responsável pela extração de características. Para tal, utiliza o modelo AlexNet pré-treinado com dados do ImageNet. Inicialmente, o modelo usado extrai as características que, em seguida, são utilizadas para treinar uma SVM (*Support vector machine*) multiclasse. Também propõe um sistema de detecção auxiliada por computador (CADe) para detecção de objetos em regiões específicas da imagem utilizando o framework R-CNN. O sistema proposto utiliza uma abordagem baseada em regiões, com uso de características obtidas por CNN (R-CNN). O framework R-CNN é usado para classificar regiões de uma imagem, regiões essas que possuem possíveis candidatos a nódulos. Esse método realiza uma pesquisa seletiva e seleciona as regiões utilizando caixas delimitadoras ou propostas de regiões. Cada região detectada é redimensionada para um tamanho padrão e posteriormente passada para o modelo de CNN. O método proposto obteve 95.2% de acurácia para a classificação dos nódulos em benigno e maligno sem *data augmentation* e 99.4% com *data augmentation*. Para a classificação de diferentes doenças, obteve uma acurácia média de 81.1% sem *data augmentation* e 84.7% com *data augmentation*.

(NÓBREGA et al., 2018) propõem a classificação de nódulos pulmonares utilizando transferência de aprendizado, assim provando a eficácia dessa metodologia, também visa verificar qual a melhor combinação CNN/classificador para esse método. Neste trabalho, foi realizada a extração de características das imagens utilizando 11 diferentes modelos de CNN, sendo todos eles pré-treinados através de transferência de aprendizado com base nos dados do ImageNet. No trabalho também foi realizada a classificação com as características extraídas utilizando 5 classificadores. Após as etapas de extração e classificação, foram feitas comparações entre as combinações apresentadas e trabalhos relacionados para identificar quais as melhores combinações utilizando 5 diferentes métricas de avaliação. A melhor combinação encontrada entre modelo de CNN e classificador, considerando o valor de acurácia, foi entre o modelo Xception para extração com a classificação usando *MultiLayer Perceptron* (MLP), obtendo um resultado de 89.91% de acurácia. Entretanto, em comparação com outros trabalhos relacionados e levando em consideração todas as métricas de avaliação utilizada, a melhor combinação foi entre ResNet50 e SVM com kernel RBF.

(FILHO et al., 2017a), propõem um sistema CAD para classificação de nódulos pulmonares utilizando processamento de imagem e técnicas de reconhecimento de padrões. O trabalho usa o método de segmentação com base no limiar de otsu para remover excessos de tecido não afetado em áreas marcadas por especialistas criando uma região de interesse. Para diferenciar entre nódulos malignos e benignos, utilizam alguns descriptores de forma do nódulo, dentre eles, a funcional de *minkowski*, que são algumas medidas capazes de caracterizar a forma do nódulo, medições de distância, medidas de triangulação e outros. Para a classificação, utilizam um algoritmo genético para selecionar o melhor modelo e o algoritmo SVM. O método proposto encontrou uma acurácia de 85.3%.

(ARESTA et al., 2019b), desenvolveram uma nova arquitetura de CNN chamada iW-Net que permite a segmentação automática ou opcional de nódulos pulmonares 3D. Baseada na 3D U-Net, a rede é composta por dois blocos em formato U, com uma redução na quantidade de filtros em comparação com a U-Net original, o primeiro bloco é responsável por uma segmentação automática dos nódulos enquanto o segundo recebe a saída do primeiro juntamente com um mapa de pesos de pixels e realiza uma correção na segmentação dessas imagens. Utilizando a base LIDC-IDRI, obtiveram um IoU de 0.55 ± 0.14 , além de, com a etapa de correção, permitir a segmentação de nódulos pequenos e não sólidos.

(SINGADKAR et al., 2020) propõem uma nova arquitetura para a segmentação de nódulos pulmonares em imagens de CT baseada em uma rede de deconvolução residual profunda com a combinação de conexões

de salto longo e curto buscando capturar a maior diversidade de nódulos em uma abordagem 2D. A metodologia proposta foi testada utilizando a base LIDC-IDRI e obteve Coeficiente Dice de 0.9497 e índice Jaccard de 0.8868.

A partir das análises dos trabalhos relacionados, foi possível concluir que as CNNs são as técnicas mais utilizadas para este tipo de aplicações, pois produzem bons resultados. Por esta razão, neste trabalho foi investigado o uso das CNNs 3D U-Net e V-Net, por serem duas das principais arquiteturas utilizadas dentro do contexto da segmentação de imagens 3D.

15.4 Metodologia, Resultados e Discussão

Nesta seção, são apresentados os métodos utilizados para a obtenção dos resultados. A abordagem proposta está dividida em subpartes, a partir da aquisição dos dados até a utilização métricas de avaliação para validar os resultados obtidos na segmentação. A Figura 117 mostra as etapas da metodologia proposta.

15.4.1 Aquisição de Imagens

A primeira etapa trata da aquisição das imagens para a aplicação do método proposto. A base de imagens utilizada neste trabalho foi a *Lung Nodule Analysis 2016 (LUNA16 Challenge)* ([SETIO et al., 2017](#)) que consiste em 888 tomografias de tórax com 1186 nódulos retirados do *Lung Image Database Consortium - Image Database Resource Initiative* (LIDC-IDRI) ([III et al., 2011a](#)). A base tem um conjunto de imagens 3D com fatias de tamanho de 512x512 com uma profundidade que varia por exame. A Figura 118 representa um dos exames presentes no conjunto de dados.

15.4.2 Pré-Processamento

Na sequência foi realizada uma etapa de pré-processamento. Imagens 3D como as presentes no conjunto de dados utilizado neste trabalho requerem uma grande quantidade de recursos computacionais para a sua manipulação. Para mitigar este problema, foi utilizada a biblioteca RadIO ([R. EMELYANOV K., 2017](#)), uma ferramenta poderosa que permite a manipulação de imagens reduzindo os custos computacionais numa estratégia de indexação. Para além de reduzir custos, tem um grande número de funções e redes neurais implementadas para utilização em imagens pulmonares.

O conjunto de dados foi dividido em subconjuntos com 70% para treino, 20% para teste e 10% para validação. Em seguida, foi criado um pipeline de pré-processamento que foi aplicado a cada um dos subconjuntos separadamente. A primeira etapa do pipeline consistiu na padronização das imagens para unidades hounsfield (HU), que é uma escala de cinza amplamente utilizada no campo da radiografia.

O passo seguinte é redimensionar as imagens e recortar pequenas regiões que são candidatas à presença de nódulos. As imagens foram redimensionadas para 128x256x256 e depois foram selecionados voxels de sete tamanhos diferentes, sendo eles: 16x16x16, 16x32x32, 16x64x64, 32x32x32, 32x64x64, 64x32x32 e 64x64x64. As imagens foram delimitadas para um tamanho menor a partir da hipótese de que pelo fato dos nódulos, ou regiões de interesse, serem pequenos em tamanho, uma imagem de menor dimensão resultaria em melhores resultados. O resultado deste processo pode ser visto na Figura 119.

O conjunto de treino possuía 1.380 voxels de cada uma das dimensões acima referidas, dos quais 832 eram amostras de nódulos e 548 eram não-nódulos. Foi realizado um processo de aumento de dados nestas pequenas regiões que compunham o conjunto de treino a fim de equilibrar os dados e aumentar a quantidade de amostras. Para o aumento de dados, foram utilizadas 3 técnicas: rotação, que consistiu em rodar a imagem em 10 graus; técnica de recorte, onde a imagem foi aumentada em 20% e depois é feito um corte do mesmo tamanho que a imagem original perto da região central da imagem; e técnica de aumento na escala de cinza, que introduz um aumento global aleatório no valor dos pixels da imagem. Para obter um número equilibrado de imagens, o aumento foi feito numa proporção de 1x para as regiões com nódulo e 2x para as regiões sem nódulo. Após este processo, foram obtidas 1.664 amostras com nódulo e 1.644 de não-nódulo.

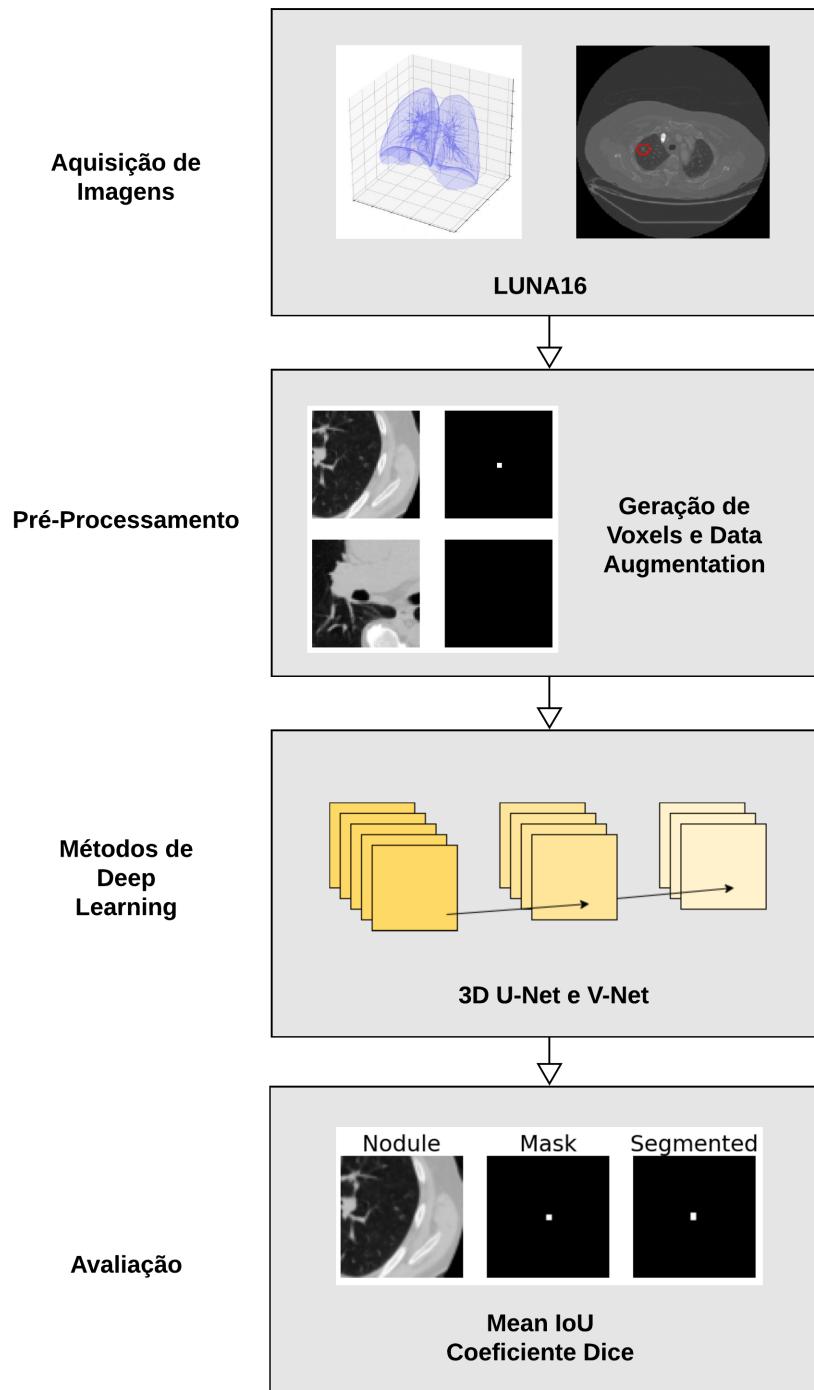


Figura 117 – Etapas do Método Proposto.

15.4.3 Avaliação/Estudos de Caso

Para a 3D U-Net foram realizados 14 testes utilizando todas as 7 diferentes dimensões de imagens geradas em etapas anteriores. Para cada dimensão, foram testados com e sem aumento dos dados. A arquitetura usada para os teste foi composta por duas camadas em cada um dos caminhos, taxa de aprendizado de 0.00001 e foi treinada por 60 épocas com um batch size de 1. Os resultados podem ser vistos na Tabela

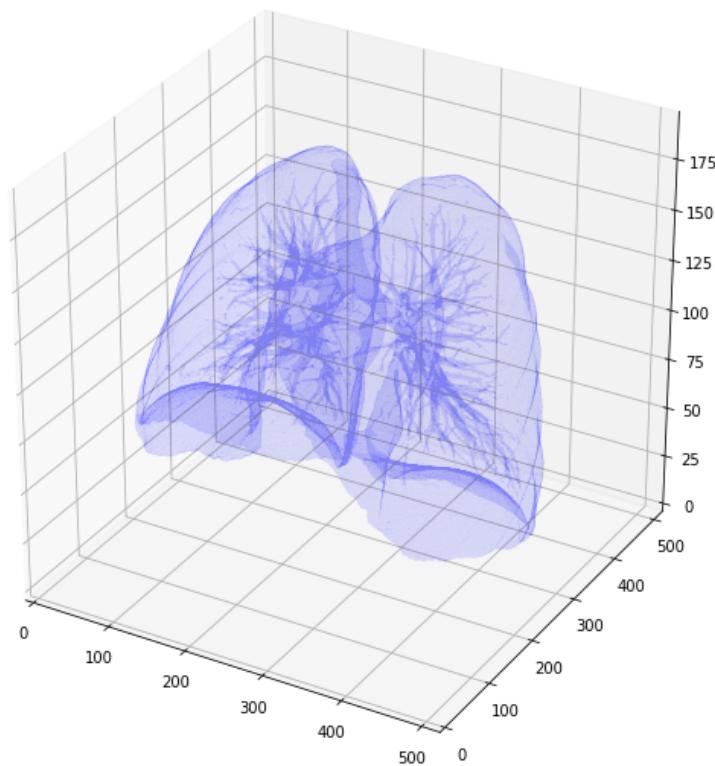


Figura 118 – Exemplo de exame presente na base de imagens

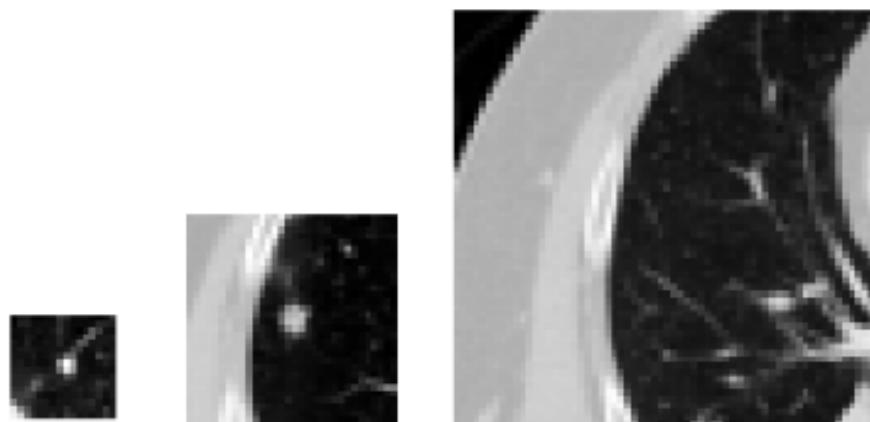


Figura 119 – Amostra das imagens após a etapa de pré-processamento.

15.4.3.

A partir da tabela, é possível observar que, a medida em que as imagens diminuem em dimensão, o Coeficiente Dice aumenta e isso se deve principalmente ao fato de que um nódulo corresponde apenas a uma pequena região de uma imagem de tórax. Com a delimitação, foi possível reduzir a dimensionalidade do problema removendo regiões grande como vasos sanguíneos que poderiam ser confundidos com nódulos.

Um outro ponto que pode ter auxiliado na melhoria dos resultados é o fato de o processamento ser mais rápido com as imagens de menor tamanho, o que permitiu que a rede fosse treinada com um maior número de épocas.

Nesse contexto, o melhor resultado obtido foi com as imagens de dimensão 16x16x16 sem *augmentation*, obtendo Coeficiente Dice de 0.846 e IoU de 0.499. A Figura 120 mostra algumas regiões segmentadas por esse modelo. Nos testes com *augmentation*, foram obtidos melhores resultados com as imagens das dimensões maiores que 16x32x32, sendo que as outras duas dimensões obtiveram melhores resultados sem *augmentation*.

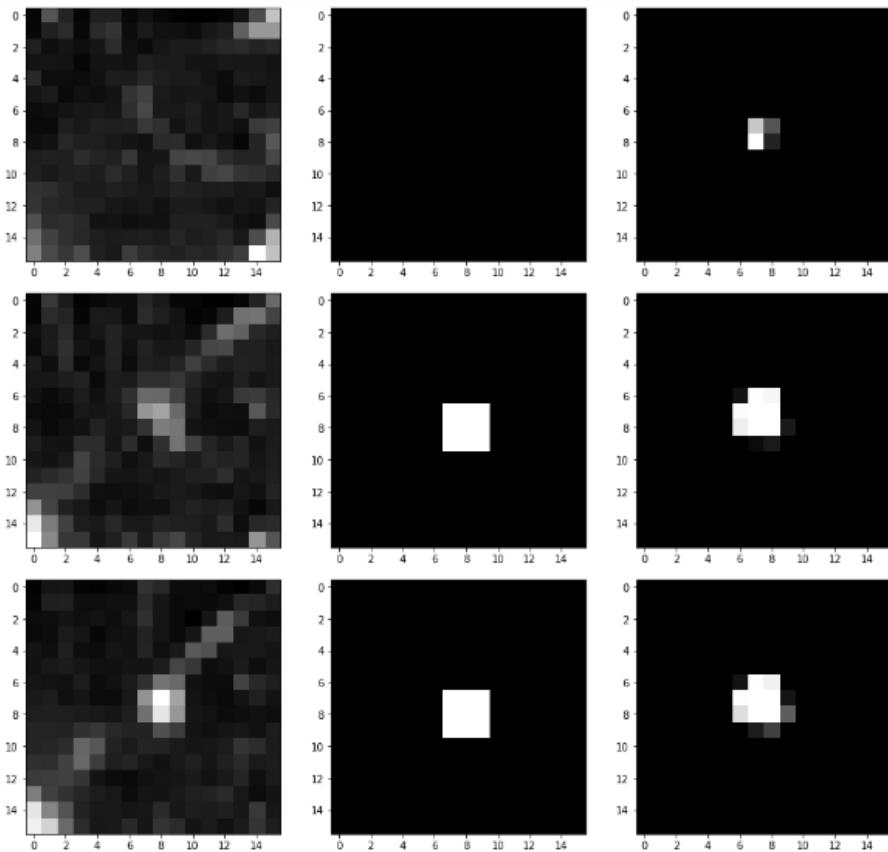


Figura 120 – Amostra das segmentadas pela melhor configuração da 3D U-Net. Na primeira coluna estão as imagens originais, na segunda as máscaras e na terceira as regiões segmentadas pela rede.

Tabela 101 – Resultados obtidos com a utilização da 3D U-Net

Dimensões da Imagem	Sem Augmentation		Com Augmentation	
	Dice	IoU	Dice	IoU
16x16x16	0.846	0.499	0.844	0.494
16x32x32	0.842	0.499	0.838	0.546
16x64x64	0.753	0.577	0.794	0.551
32x32x32	0.801	0.501	0.812	0.543
32x64x64	0.629	0.566	0.795	0.535
64x32x32	0.656	0.561	0.785	0.536
64x64x64	0.716	0.553	0.748	0.536

Para a V-Net, foi utilizada uma arquitetura com 4 camadas em cada caminho com uma taxa de aprendizado de 0.0001 e foi treinada com 10 épocas e batch size também de 1. Para esse modelo, foram realizados os mesmos 14 testes com as amostras de imagens geradas durante o pré-processamento, com e sem *augmentation*. Os resultados podem ser analisados a partir da Tabela 315.4.3.

Para esse caso a rede não conseguiu obter resultados tão expressivos, ou seja, não conseguiu segmentar tão bem uma região de nódulo. O melhor resultado obtido foram um Coeficiente Dice de 0.685 com IoU de

Tabela 102 – Resultados obtidos com a utilização da V-Net

Dimensões da Imagem	Sem Augmentation		Com Augmentation	
	Dice	IoU	Dice	IoU
16x16x16	0.685	0.987	0.685	0.987
16x32x32	0.685	0.996	0.685	0.996
16x64x64	0.664	0.999	0.664	0.999
32x32x32	0.675	0.998	0.675	0.998
32x64x64	0.664	0.999	0.661	0.999
64x32x32	0.674	0.999	0.674	0.999
64x64x64	0.635	0.999	0.631	0.999

0.978 para as imagens com dimensão 16x16x16.

Para as duas redes, quanto menor a dimensão, melhor o resultado. Isso se deve ao fato de que uma menor dimensão da imagem delimita melhor o problema, uma vez que um nódulo é uma região pequena, removendo muitos dos ruídos que podem ser confundidos com nódulos. Para os teste com *data augmentation* não obtivemos melhorias significativas nesse modelo.

Para esses testes, obtivemos o IoU para a 3D U-Net menor em comparação com o IoU da V-Net devido ao fato de que o IoU considera os acertos para pixels 0 e 1, onde calcula-se o resultado para a região 0 e para a região 1 separadamente e faz uma média entre as duas. A 3D U-Net conseguiu segmentar melhor o nódulo, conseguindo valores próximos na proporção para cada pixel, obtendo um resultado mais confiável de IoU. Enquanto a V-Net teve uma diferença muito grande nessas proporções porque não segmentou tão bem a região de nódulo, fazendo com que a média fosse influenciada por um valor alto na proporção de Os. Isso também explica as diferenças entre os valores do Dice, uma vez que essa métrica considera apenas as regiões segmentadas corretamente.

A Tabela 15.4.3 traz uma comparação dos resultados obtidos com os trabalhos relacionados. Um ponto a ser destacado para a análise da mesma é o fato de que os trabalhos utilizam bases e métricas diferentes e, portanto, a tabela foi colocada aqui apenas para uma base de comparação.

Tabela 103 – Comparação dos Resultados

Autor	Base	Acurácia	Sensibilidade	IoU	Dice
(GU et al., 2018)	LIDC-IDRI	-	0.879-0.929	-	-
(KIDO; HIRANO; HASHIMOTO, 2018)	LIDC-IDRI	0.811-0.853	-	-	-
(NÓBREGA et al., 2018)	LIDC-IDRI	0.899	-	-	-
(FILHO et al., 2017a)	LIDC-IDRI	0.853	-	-	-
(ARESTA et al., 2019b)	LIDC-IDRI	-	-	0.55	-
(SINGADKAR et al., 2020)	LIDC-IDRI	-	-	-	0.949
Método Proposto (V-Net)	LUNA16	-	-	0.996	0.685
Método Proposto (3D U-Net)	LUNA16	-	-	0.499	0.846

15.4.4 Discussão

Os melhores resultados obtidos para a 3D U-Net foram um Coeficiente Dice de 0.846 e uma mean IoU de 0.499, resultados esses que foram positivamente influenciados pela diminuição das dimensões da imagem. Essa diminuição foi responsável por uma melhor delimitação das regiões de interesse, removendo muito do ruído e das regiões de não nódulo que poderiam ser segmentadas como nódulo. Os experimentos com *data augmentation* obtiveram os melhores resultados para alguns casos de teste.

Para a V-Net, os melhores resultados obtidos foram Coeficiente Dice de 0.685 e mean IoU de 0.499. Esses resultados também foram muito influenciados pela redução das imagens, entretanto, não atingiu bons resultados, apesar de ser uma CNN que geralmente funciona melhor para imagens como as utilizadas no trabalho. Um dos motivos que pode ser causador deste problema é a quantidade de camadas na arquitetura,

tendo em vista que devido ao tamanho da imagem, uma arquitetura menor pode ser mais eficiente. Para esse modelo, os testes com *data augmentation* não obtiveram melhorias significativas.

15.5 Conclusão

Neste trabalho, foram apresentadas algumas abordagens comuns à segmentação de imagens aplicadas a imagens de tomografia computorizada do tórax para a detecção de nódulos pulmonares. Foram avaliadas duas abordagens de aprendizagem profunda, 3D U-Net e V-Net, com imagens de entrada em diferentes dimensões. Os melhores resultados obtidos correspondem para a 3D U-Net em uma arquitetura com duas camadas em cada lado, para as imagens com as dimensões de 16x16x16 sem *augmentation*. Isso se deve principalmente ao fato de que as imagens de menor dimensão representam melhor o problema trabalhado além de remover possíveis regiões de não nódulo que podem ser confundidas com os nódulos.

Como trabalhos futuros, pretende-se testar novas configurações de parâmetros para a 3D V-Net e posteriormente classificar cada região obtida em nódulo e não nódulo, a fim de avaliar a precisão da segmentação. Para tal, será realizada a extração das características de forma e textura destas regiões para uma classificação posterior. Em seguida, após a fase de classificação, a avaliação quantitativa do método proposto será realizada utilizando métricas de avaliação de classificação. Além disso, pretende-se realizar os mesmos testes apresentados para outras arquiteturas de CNN para segmentação.

Redes Neurais Convolucionais para Identificação de Metástase em Imagens Histopatológicas.

Autores: Renesio Joaquim de Oliveira Costa; Daniel de Sousa Luz; Flávio Henrique Duarte de Araújo.

Resumo: Metástase é um tumor formado a partir de células cancerígenas que, se separaram de um primeiro tumor e migraram através dos vasos linfáticos ou vasos sanguíneos para outra parte do corpo, onde se estabeleceram. Atualmente, a detecção de metástases é realizada por patologistas revisando grandes extensões de tecidos biológicos. Este processo de detecção é tido na literatura como trabalhoso e propenso a erros. Com base na problemática de detecção, este trabalho apresenta três estratégias de *deep learning* para identificação automática, capaz de localizar tecido tumoral em imagens de microscopia da base PCam do desafio *Camelyon16*. A PCam é uma base com 327.680 imagens coloridas, destas 262.144 são dedicadas para treinamento, 32.768 são destinadas para teste de aprendizado. Com uma base dessa extensão é possível criar um modelo de classificação automática que poderá auxiliar na decisão do patologista. Para a construção desse método propomos uma arquitetura de Rede Neural Convolucional com 9 camadas, como também utilizar outros modelos já prontos de redes neurais pré-treinadas, e também técnicas de aperfeiçoamento de aprendizagem como *fine-tuning*, para assim determinarmos a melhor técnica entre essas estratégias. Obteve-se um resultado promissor com um AUC de 96,00% utilizando a técnica de *fine-tuning*, mostrando que modelos de *deep learning*, podem ser usados, para tarefas de classificação de imagens.

Keywords: Histopatologia. Redes. Neurais Convolucionais. Aprendizado Profundo. Transferência de Aprendizado. Ajuste Fino. PCam.

16.1 Introdução

Conforme o Instituto Nacional do Câncer (INCA), a metástase é um tumor formado a partir de células cancerígenas que se separaram de um primeiro tumor e migraram através dos vasos linfáticos ou vasos sanguíneos para outra parte do corpo onde se estabeleceram. Há muitos anos, o sistema de estadiamento clínico de tumores TNM, tem sido adotado como padrão de classificação para caracterizar os tumores através de imagens histopatológicas, propondo a terapia mais adequada e, assim, estimar a sobrevida dos pacientes.

Tendo em vista a importância do exame histopatológico do tumor, é necessária uma observação detalhada na imagem da região afetada. Com auxílio desse tipo de exame onde se analisa imagens a respeito da região afetada, métodos computacionais de classificação com base em Inteligência Artificial (IA) se tornam essenciais para gerar resultados, que auxiliam no diagnóstico das imagens histopatológicas.

O Aprendizado de Máquina (ML) é uma área de estudo da Inteligência Artificial (IA) que foca no aprendizado a partir de dados, estudos nessa linha de pesquisa tornaram-se cada vez mais onipresentes no estudo de dados de domínio médico. No campo da visão computacional o uso de estruturas de Redes Neurais Convolucionais (CNN) e outros modelos de Aprendizado Profundo (DL) são utilizados para tarefas relacionadas com classificação e segmentação de imagens médicas. Por exemplo, o uso de tais estruturas para detectar metástases em manchas extraídas de imagens histopatológicas levam a resultados que ultrapassam muitas vezes o de patologistas humanos (VEELING et al., 2018), (LIU et al., 2017).

A importância de tais resultados é aumentada pelo fato de que especialistas em patologia ao realizarem a mesma tarefa, podem discordar em mais de 20% dos casos (BEJNORDI et al., 2017). Essa tarefa também é descrita na literatura como tediosa e demorada, o que torna mais propensa a erros humanos.

A existência de sistemas computacionais que podem executar esta tarefa de classificação em níveis muito altos de precisão, mostra ser uma maneira de superar este problema. Como a demanda por patologistas é extremamente alta em todo o mundo, esses sistemas de IA poderiam ajudar muito no balanceamento de sua carga de trabalho e permitir um diagnóstico mais preciso e consistente. Esta questão, no entanto, se estende a muitas outras áreas onde as decisões apoiadas pela IA são utilizadas como ferramentas de segunda opinião a respeito de um determinado problema.

Este Capítulo aborda técnicas de classificação com base em aprendizado profundo, assim como um conjunto de métodos e procedimentos de Transferência de Aprendizado e o uso da tática de *fine-tuning* para obtenção de melhores resultados, aplicando técnicas e estratégias computacionais para o problema da classificação de metástases em Imagem de Slide Inteira (ISI). O conjunto de dados relacionado a essa tarefa é o *benchmark* Patch Camelyon (PCam) lançado em março de 2019 em um desafio de classificação pela patchcamelyon.grand-challenge.org (VEELING et al., 2018).

Este capítulo está organizado em cinco Seções. Na presente Seção são detalhados o contexto e proposta do trabalho. Na Seção 16.2 são detalhados os principais trabalhos relacionados. Na Seção 16.3 são detalhados os materiais e métodos, iniciando pela aquisição das imagens e embasamentos teóricos a respeito das técnicas e métricas utilizadas nesse projeto. Na Seção 16.5 são apresentados e discutidos os resultados obtidos com a metodologia proposta. E por fim, na Seção 16.8 são apresentadas as conclusões e trabalhos futuros.

16.2 Trabalhos Relacionados

No que diz respeito aos trabalhos voltados a base escolhida, pode se notar que por se tratar de uma base bem recente, poucos utilizaram esses dados para análise, contudo há uma tendência de crescimento em estudos baseados nesta linha de pesquisa. Foram analisados trabalhos com o mesmo intuito e objetivos semelhantes aos deste projeto, ou seja, uso de métodos computacionais para detecção de metástase, entretanto esses trabalhos utilizam outras bases de dados, ou uma versão mais antiga da base utilizada nesse projeto. Pôde-se observar que há uma tendência de crescimento no uso de técnicas computacionais, tanto para extração das características, quanto para classificação de imagens histopatológicas.

A Tabela 104 resume os principais trabalhos encontrados na literatura e relacionados com o objetivo dessa pesquisa. Na Tabela 1 também são mostradas as bases de imagens utilizadas por cada autor, assim como as arquiteturas e os resultados de acurácia(Acc.) ou Área sob a curva ROC (AUC).

Wang et al. (2016) propuseram um sistema de aprendizado profundo para a detecção automática de câncer metastático de slide inteiro em imagens de linfonodos. Os autores avaliaram o desempenho de quatro arquiteturas de redes de aprendizado profundo conhecidas, para esta tarefa de classificação: GoogLeNet, AlexNet, VGG16 e FaceNet na base de imagens Camelyon16. Eles conseguiram o melhor resultado com o GoogLeNet, atingindo uma precisão de 98.4%.

Tabela 104 – Tabela contendo a descrição dos trabalhos relacionados área de classificação de imagens histológicas. Acurácia (Acc) e Área sob a curva ROC (AUC).

Referências	Base de Imagens	Arquitetura	Resultados
(WANG et al., 2016)	Camelyon16	GoogleNet	Acc 98,4%
(LIU et al., 2017)	Camelyon16	Inception V3	AUC 97,5%
(VEELING et al., 2018)	Camelyon16 , Pcam e BreakHi	CNN Tradicional	Acc 96,1%
(LINMANS et al., 2018)	PCam	P4M U-Net, Unet	Acc 83,7%
(JAISWAL et al., 2019)	PCam	DenseNet201	Acc 97,94%

Liu et al. (2017) desenvolveram uma abordagem para classificação de imagens histológicas, onde os autores utilizam a arquitetura de rede pré-treinada *Inception* (V3) para extração das características, usando o classificador *Random forest*. O conjunto de imagens é o *Camelyon16*, do desafio de detecção de tumor ao nível de lesão. O método proposto obteve 97,5% de AUC.

Veeling et al. (2018) desenvolveram um novo modelo de classificação treinando o modelo com a base de imagens histopatológicas *Camelyon16* e com foco na estabilidade das previsões. Para avaliar o modelo proposto os autores utilizaram a base PCam e testaram a arquitetura com a base de dados *BreakHis*. O modelo proposto alcançou a precisão de 96,1%.

Linmans et al. (2018) propuseram um modelo de segmentação semântica que explora simetrias de rotação, onde são utilizadas duas arquiteturas de redes (Unet e P4M U-Net), além disso, o aumento de dados é empregado. O conjunto de imagens utilizadas nesse projeto é o PCam fornecido como parte do desafio de classificação de imagens histológicas. Eles conseguiram com a rede P4M U-Net 83,7% de precisão.

Jaiswal et al. (2019) propuseram um modelo baseado em CNN, validado no conjunto de dados da PCam, onde usaram uma abordagem de aprendizagem semi-supervisionada em diferentes modelos pré-treinados como VGG16, InceptionResNetV2, InceptionV3, Xception, ResNet101 e DenseNet201 no sentido de transferência de aprendizagem fazendo a diferenciação entre classes de baixa densidade. O DenseNet201 obteve uma acurácia de 97%.

Pesquisas efetuadas mostraram que, as técnicas de aprendizado profundo estão se tornando primordiais, na resolução de diagnóstico médico através de imagens, por esta razão foi determinado o uso de técnicas envolvendo arquiteturas de CNN.

Neste Capítulo são avaliadas três técnicas baseadas em aprendizado profundo. A primeira técnica é uma CNN própria treinada com a inicialização aleatória de pesos. Como segunda técnica foram utilizadas as arquiteturas pré-treinadas VGG16, VGG19, e ResNet50 para extrair características e com ajuda de classificadores, classificar os dados. Como terceira estratégia, foi utilizada a técnica de *fine-tuning*. Além disso, os modelos foram treinados e testados com a versão mais atual da base PCam que foi disponibilizada em março de 2019, no desafio de classificação *Camelyon16 Challenge*. Os melhores resultados obtidos nesses testes serão comparados com os presentes no desafio, visto que até então não há poucos trabalhos publicados que utilizaram a versão mais atual da base PCam.

16.3 Materiais e Métodos

Nesta seção são apresentados a base de imagens, conceitos importantes para uma melhor compreensão deste Capítulo, e as principais arquiteturas e técnicas de CNNs utilizadas para o processamento das imagens histológicas.

16.3.1 Base de Dados

O conjunto de imagens utilizadas nesse trabalho consiste em uma base de dados lançada em março de 2019, a base conta com 327.680 imagens coloridas com dimensão (96 x 96px) extraídas de exames histopatológicos de seções de linfonodos. Cada imagem é anotada com um rótulo binário indicando a presença ou não tecido metastático.

O conjunto de imagens é dividido em treinamento com 262.144(2^{18}) exemplos e um teste de 32.768 (2^{15}) exemplos. Não há sobreposição em ISIs entre as divisões e todas as divisões têm um equilíbrio de 50/50 entre exemplos positivos e negativos.

A PCam é derivada do Camelyon16 Challenge, que contém WSIs com 400 H&E de seções de linfonodo sentinelas. As lâminas foram adquiridas e digitalizadas em dois centros diferentes usando uma objetiva de 40x (resolução de píxeis resultante de 0,243 micrônios). A Figura 121 apresenta exemplos de imagens das duas classes, classe 0 representa tecido não metastático e classe 1 tecido metastático.

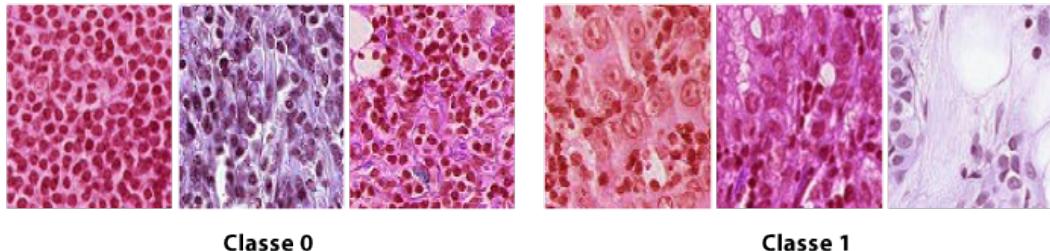


Figura 121 – Exemplo de imagens da base PCam divididas duas classes: classe 0, que não apresenta tecido metastático e classe 1 que apresenta tecido metastático. Fonte: Banco de dados PCam (PCam) 2019.

16.3.2 Aprendizado Profundo

O aprendizado profundo é uma sub área do aprendizado de máquina que trata de modelos vagamente inspirados no cérebro humano. Tecnicamente, toda Rede Neural com mais de uma camada escondida é chamada de Rede Neural Profunda e tudo que ela aprende é por meio do Aprendizado Profundo. Para Lecun, Bengio e Hinton (2015) o aprendizado profundo é uma área que contem uma pilha de multicamadas simples, todos (ou a maioria) estão sujeitos à aprendizagem, e muitos dos quais calcula mapeamentos de entrada-saída não lineares.

Tradicionalmente, a qualidade dos algoritmos depende muito da representação dos dados em certas características (as chamadas *features*). Assim, a criação de recursos válidos e pré-processamento consomem grande parte dos esforços dos especialistas. Já os algoritmos de aprendizado profundo têm uma abordagem inovadora, pois dispensam grande parte desse pré-processamento e geram automaticamente propriedades invariantes em suas camadas hierárquicas de representação (HELABS, 2019).

Brigade (2016) afirma que atualmente o reconhecimento de imagens por máquinas treinadas através de aprendizado profundo em alguns cenários possuem uma taxa de acerto maior que a de humanos. Várias arquiteturas de aprendizado profundo, tais como CNN têm sido aplicadas em áreas como visão computacional, reconhecimento automático de fala, processamento de linguagem natural, reconhecimento de áudio e bioinformática, onde elas têm mostrado-se capazes de produzir resultados significantes.

A seguir são abordadas as fundamentações sobre as Redes Neurais Convolucionais (CNN) utilizadas nesse projeto.

16.3.3 Redes Neurais Convolucionais

CNN ou *ConvNets* são uma categoria de redes neurais usadas no reconhecimento e processamento de imagens que é especificamente projetada para processar dados de pixel. PONTI et al. (2018) versa que sua característica marcante é ser composta basicamente de camadas convolucionais, que processa as entradas considerando campos locais. Essas redes constituem uma escolha de topologia ou arquitetura projetadas para reduzir o número de parâmetros a serem aprendidos otimizando o tempo de treinamento.

As CNNs são projetadas para processar dados que vêm na forma de múltiplos *arrays*, por exemplo, uma imagem colorida composta por três matrizes contendo intensidades de pixel nos três canais de cores (LECUN; BENGIO; HINTON, 2015). Elas são capazes de capturar com sucesso as dependências espaciais e

temporais em uma imagem através da aplicação de filtros. Kovaleski (2018) afirma que a mesma se utiliza desses filtros para analisar pequenos grupos de dados, extraíndo suas características mais relevantes e relata que apenas estas características são passadas adiante para as próximas camadas, como apresentado na Figura 122.

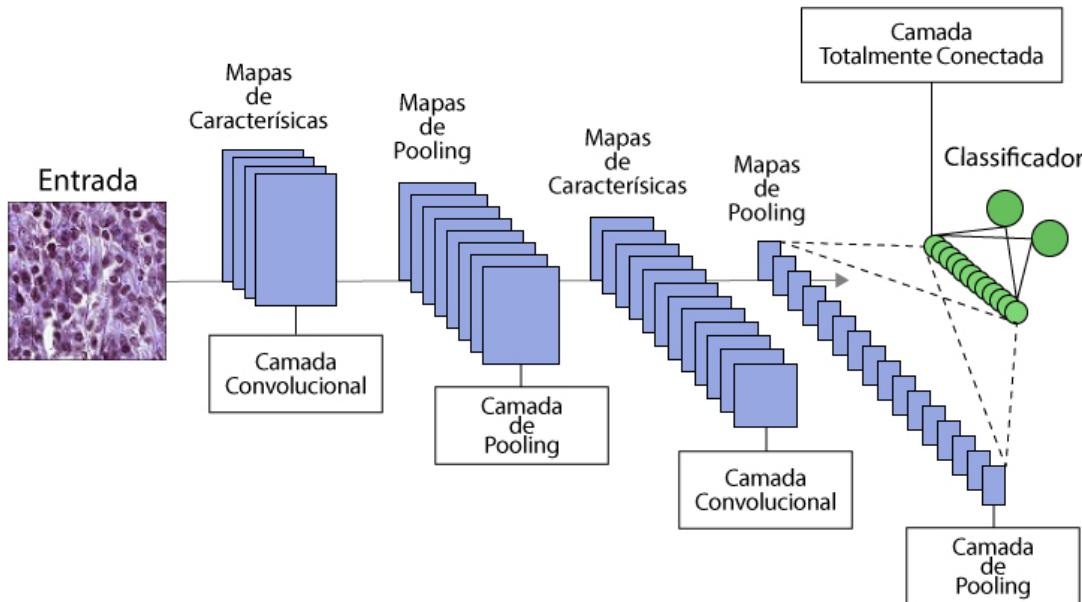


Figura 122 – Ilustração da arquitetura de uma CNN com duas camadas convolucionais, duas de *pooling*, uma totalmente conectada e a de saída. Fonte: Adaptação de Saha (2018).

Ao contrário de uma Rede Neural regular, as camadas de uma CNN possuem neurônios dispostos em 3 dimensões: largura, altura e profundidade. Além disso, os neurônios em uma camada não se conectam a todos os neurônios da camada seguinte, mas apenas a uma pequena região dela. Por fim, a saída final será reduzida a um único vetor de pontuações de probabilidade, organizadas ao longo da dimensão de profundidade.

A arquitetura executa um melhor ajuste ao conjunto de dados da imagem devido à redução no número de parâmetros envolvidos e à capacidade de reutilização dos pesos. A grande vantagem da aplicação das CNNs para o Processamento Digital de Imagens (PDI) é sua grande capacidade de extrair características, por trabalhar com convoluções, ou seja, criando “sub-quadros de bits” de forma que seja possível realçar contornos e áreas de interesse de cada imagem (ROSEBROCK, 2017b).

A seguir são apresentadas as fundamentações e principais características das mais diversas arquiteturas de redes convolucionais utilizados nesse trabalho para a extração de características das imagens.

16.3.3.1 Arquitetura das Redes Avaliadas

Após analisar a bibliografia que resultou da pesquisa a respeito do tema, foram encontrados diversos modelos de arquitetura compatíveis com a proposta desse trabalho. Com isso, foi possível selecionar dois dos principais modelos, visto que essas duas arquiteturas são as mais comuns na literatura. Para um estudo mais profundo e consequente avaliação do melhor desempenho, os modelos utilizados são descritos a seguir.

- **Modelo 1 - Visual Geometry Group - VGG** A VGG é um modelo de CNN inicialmente proposto por Simonyan e Zisserman (2014a). A mesma garantiu a primeira e segunda posições nas tarefas de localização e classificação, respectivamente, no desafio ImageNet 2014. Ela faz melhorias sobre o

AlexNet construída por Krizhevsky, Sutskever e Hinton (2012) substituindo grandes filtros do *kernel* por múltiplos filtros do tamanho 3×3 um após o outro. As profundidades mais comuns desse modelo são de 16 e 19 camadas.

Com a introdução da VGG, elas trouxeram alguns padrões: sugeriu-se que todos os filtros tivessem tamanho de 3×3 , *poolings* máximos deveriam ser colocados após cada 2 convoluções e o número de filtros deveria ser dobrado após cada *max-pooling*. A VGGNet é considerada a rede preferida pela comunidade para Transferência de Aprendizado, pois sua arquitetura uniforme é boa para extrair características de imagens. Assim, a VGG tem sido utilizada como modelo na construção de novas redes, como Unet, TernausNet e várias outras redes.

- **Modelo 2 - ResNet**

A ResNet é uma arquitetura que tem como principal característica suas conexões residuais. Na aprendizagem residual, em vez de tentar aprender algumas características, tal arquitetura é voltada para aprender alguns resíduos. O termo residual pode ser simplesmente entendido como subtração de características aprendidas da entrada dessa camada. A ResNet faz isso usando conexões de atalho, que são conexões em que o gradiente é diretamente retro-propagado para camadas anteriores. Na Figura 123 é demonstrado o funcionamento de um bloco residual.

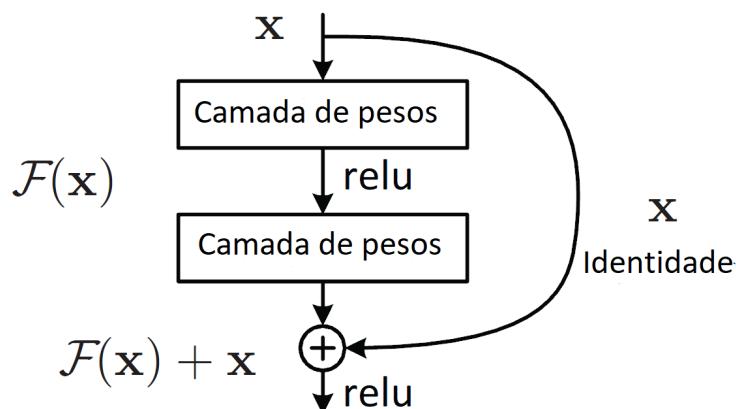


Figura 123 – Ilustração de um bloco residual na arquitetura da rede ResNet. Fonte: Adaptação de He et al. (2016).

Um dos problemas que a ResNet resolve é o famoso desaparecimento de gradiente. Isso ocorre porque, quando a rede é muito profunda, os gradientes de onde a função de perda é calculada diminuem facilmente para zero. Este resultado nos pesos nunca atualiza seus valores, portanto nenhum aprendizado será realizado.

Com tal arquitetura, os gradientes podem fluir diretamente através das conexões de salto. Para Chen et al. (2017) a ResNet fornece uma estrutura de treinamento para facilitar formação de redes substancialmente mais profundas que as utilizadas anteriormente.

16.3.4 Transferência de Aprendizado

A Transferência de Aprendizado é um método de aprendizado de máquina, em que, um modelo desenvolvido para uma tarefa é reutilizado como ponto de partida para um modelo em uma segunda tarefa (PONTI et al., 2018). É uma abordagem popular em aprendizado profundo onde os modelos pré-treinados são usados como fase inicial em tarefas de visão computacional.

Normalmente, precisa-se de muitos dados para treinar uma Rede Neural, mas nem sempre é possível ter acesso a dados suficientes. É aí que a Transferência de Aprendizado pode ser utilizada, porque com ela pode-se construir um modelo de aprendizado de máquina sólida com comparativamente poucos dados de treinamento, já que o modelo já é pré-treinado (TORREY; SHAVLIK, 2010).

Embora as redes profundas tenham se mostrado promissoras na representação de informações, tais modelos profundos supervisionados, exigem grandes quantidades de dados rotulados e recursos de computação, considerados caros (DU et al., 2017b). O processo de Transferência de Aprendizado é demonstrado pela Figura 124.

Na Transferência de Aprendizado, os modelos pré-treinados são usados para extrair características de imagens, sendo estes modelos treinados em milhões de imagens pertencentes a milhares de classes, em várias GPUs de alta potência por vários dias e todos esses recursos já aprendidos são submetidos para resolver problemas relacionados.

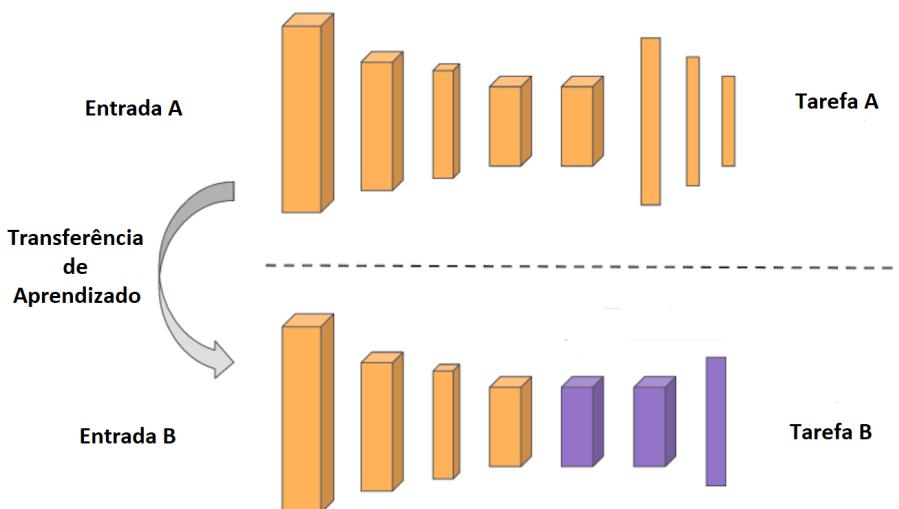


Figura 124 – Exemplo de Transferência de Aprendizado. Fonte: Adaptação de Subodh (2018).

Hu et al. (2015) corrobora em afirmar que as vantagens da Transferência de Aprendizado incluem aceleração do tempo de treinamento, redução parâmetros e uma arquitetura de redes neurais profundas que busca manter alto desempenho de classificação, especialmente em um conjunto limitado de imagens. Existem duas abordagens principais para implementar na Transferência de Aprendizado; são elas: inicialização de peso e extração de características (BROWNLEE, 2019). No caso da primeira abordagem os pesos nas camadas reutilizadas podem ser usados como ponto de partida para o processo de treinamento e adaptados em resposta ao novo problema, essa técnica é conhecida como *fine-tuning*.

A estratégia de *fine-tuning* consiste em dar continuidade ao treinamento de uma rede pré-treinada utilizando o algoritmo *Backpropagation* e uma nova base de imagens. Em outras palavras, os pesos de todas as camadas de uma rede pré-treinada, com exceção da última camada, são utilizados para a inicialização de uma nova CNN (ARAÚJO et al., 2017a). É possível fazer o *fine-tuning* de todas as camadas de uma CNN, ou somente das últimas camadas. Isso é motivado pelo fato que as primeiras camadas da rede contém extractores mais genéricos que podem ser utilizados para diferentes tarefas, como detectores de bordas e de cores, porém, as camadas mais profundas possuem detalhes específicos da base com a qual a rede foi originalmente treinada (ARAÚJO et al., 2017a).

Na segunda abordagem pode-se remover a camada de saída (aquele que fornece as probabilidades de estar em cada uma das 1000 classes) e, em seguida, usar toda a rede como um extrator de características fixo para o novo conjunto de imagens. Depois disso um classificador externo é usado para classificação das imagens. Nesse trabalho utilizamos a segunda abordagem envolvendo os classificadores (*Random Forest* e *SVM*), porque os mesmos eram os mais abordados nos trabalhos pesquisados.

Portanto, ela também economiza muito tempo de treinamento, que às vezes pode levar dias ou até semanas para treinar uma Rede Neural Profunda do zero em uma tarefa complexa. Além desse benefício a

Transferência de Aprendizado resulta em menor erro de generalização (GOODFELLOW; BENGIO; COURVILLE, 2016).

16.4 Métricas de Avaliação

Para a avaliação do método proposto nesse trabalho foram utilizadas as seguintes métricas: Sensibilidade (S), Especificidade (E), Acurácia (A), Área Sob a Curva ROC (AUC) e *Kappa* (K). Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em função de quatro valores, sendo eles: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

1. Acurácia

Esta é a métrica mais simples. Consiste basicamente o número de acertos (positivos) dividido pelo número total de exemplos. Essa métrica é mais apropriada para bases com a mesma proporção de exemplos para cada classe, e quando as penalidades de acerto e erro para cada classe forem as mesmas (MISHRA, 2018). Em problemas com classes desproporcionais, ela causa uma falsa impressão de bom desempenho. A acurácia pode ser calculada conforme a Equação 124.

$$A = \frac{VP + VN}{VP + VN + FP + FN}. \quad (16.1)$$

2. Sensibilidade

A sensibilidade é uma medida da proporção de casos positivos reais que foram previstos como positivos (ou positivos verdadeiros). Em outras palavras, um teste altamente sensível é aquele que identifica corretamente pacientes com uma doença, o mesmo pode ser útil para descartar uma doença se uma pessoa tiver um resultado negativo (KUMAR, 2018). A sensibilidade pode ser calculada conforme a Equação 16.2

$$S = \frac{VP}{VP + FN}. \quad (16.2)$$

3. Especificidade

A especificidade de um teste (também chamado de Taxa Negativa Verdadeira) é a proporção de pessoas sem a doença que terão um resultado negativo. Em outras palavras, a especificidade de um teste se refere a quanto bem um teste identifica pacientes que não têm uma doença (KUMAR, 2018). A especificidade pode ser calculada com a Equação 16.3

$$E = \frac{VN}{VN + FP}. \quad (16.3)$$

4. *Kappa*

O coeficiente *Kappa* é uma medida de concordância estatística formulada por Jacob Cohen, tal métrica mede concordância entre dois avaliadores que classificam cada N itens em C categorias mutuamente exclusivas. O *Kappa* de Cohen é forte dependente de vários fatores, incluindo, por exemplo, número de classes e número de exemplos a serem classificados. O *Kappa* pode ser calculado conforme a Equação 16.4

$$K = \frac{\theta_1 - \theta_2}{1 - \theta_2}, \quad (16.4)$$

onde

$$\theta_1 = \frac{VP + VN}{VP + VN + FP + FN}, \quad (16.5)$$

e

$$\theta_2 = \frac{[(VP + FN)(VP + FP)] + [(VN + FN)(VN + FP)]}{(VP + VN + FP + FN)^2}. \quad (16.6)$$

Sobre a avaliação dos classificadores, Landis e Koch (1977b) partitionaram o nível de exatidão do Índice *Kappa* conforme a Tabela 105.

Tabela 105 – Nível de exatidão de uma classificação, conforme o valor de índice *Kappa*

Índice <i>Kappa</i> (K)	Qualidade
$K < 0.2$	Ruim
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito Boa
≥ 0.8	Excelente

5. Área Sob a Curva ROC (*Area Under ROC*)

É uma métrica para classificação binária, é provavelmente uma das mais populares. Braga (2001) afirma que a curva ROC é um método gráfico simples e robusto, permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. AUC é uma abreviação da área sob a curva. É usado na análise de classificação para determinar qual dos modelos utilizados prediz melhor as classes. Quanto mais próximo o AUC de um modelo chegar a 1, melhor ele será (GIANNINI, 2012). Assim, os modelos com AUCs mais altos são preferidos em relação àqueles com AUCs mais baixos.

16.5 Modelos Propostos e Resultados

Neste Capítulo é um método automático para a classificação de imagens extraídas de exames histopatológicos de seções de linfonodos é apresentado. Para determinar a arquitetura e/ou técnica que produzisse os melhores resultados, testamos três estratégias: uma CNN elaborada do zero para classificar as imagens que são divididas em duas classes, a que não apresenta e a que apresenta tecido metastático; Redes Neurais pré-treinadas usadas para extrair características das imagens, com ajuda de classificadores para determinar a classe das imagens. Foram utilizados os classificadores SVM e seus diferentes *kernels* e o *Random forest*; Como terceira estratégia utilizamos o *fine-tuning*, utilizando a arquitetura clássica inicializada com os pesos do modelo treinado para a base ImageNet, e este treinamento foi continuado utilizando a base utilizada nesse projeto, juntamente com as camadas da CNN proposta. Abaixo são detalhadas as estratégias, juntamente com os resultados obtidos nos testes.

16.6 Rede treinada do zero

Abaixo são mostrados os resultados obtidos de acordo com os testes realizados durante a implementação do zero da Rede Neural Convolucional, logo em seguida é discutida a implementação em cada modelo. A Tabela 3 apresenta o modelo implementado (Modelo), com a quantidade de imagens utilizadas para o treinamento (Quant. Imagens). Estão também listadas as seguintes métricas: Acurácia(Acc.), Especificidade (Esp.), Sensibilidade(Sens.), Área sob a curva ROC (AUC), e o coeficiente *Kappa*(*Kappa*). Vale ressaltar que após cada camada de convolução havia uma camada de *pooling* com filtro de tamanho 2X2. Todos os modelos foram elaborados utilizando a API *Keras*, uma API de alto nível para redes neurais usando a biblioteca *tensorflow* como *backend*.

O Modelo 1 foi uma rede criada sem camadas de convolução ou *pooling*, apenas com uma camada de entrada com 32 neurônios, uma camada oculta com 16 neurônios, e 2 neurônios na camada de saída. Para o treinamento usamos um total de 10.000 (dez mil) imagens, com um tamanho de lote igual a 32, isso combinado com um treinamento de 300 épocas. Com a base de testesm no modelo 1, atingimos uma acurácia de 49,63%, com uma especificidade de 00,00% e sensibilidade de 00,00%, conseguindo um AUC

Tabela 106 – Tabela de testes da arquitetura treinada do zero.

Modelo	Quant. Imagens	Acc.%	Esp.%	Sens.%	AUC%	Kappa
Modelo 1	10,000	49,63	0,0	0,0	50,00	0,19
Modelo 2	10.000	73,68	51,08	96,26	82,00	0,37
Modelo 3	10.000	79,81	64,77	94,84	84,00	0,59
Modelo 4	100.000	82,26	68,15	93,85	90,00	0,63
Modelo 5	262.144	83,63	72,49	94,76	93,00	0,67

50,00%. Ou seja, com essa implementação não foi possível aprender o problema e a rede mandou todos os exemplos de teste para na mesma classe.

No modelo 2 foram feitas modificações na rede do modelo 1, adicionamos 2 camadas de convolução e 2 de *pooling*, todas com filtros de tamanho 3x3 e 2x2 respectivamente. Para treinamento foram separadas 10.000 (dez mil) imagens. O modelo continha uma camada de entrada com 32 neurônios, uma camada oculta com 16 neurônios e uma camada de saída com 2 neurônios, um tamanho de lote igual a 32, com um total de 300 épocas. Na base de testes, no modelo 2, atingiu-se uma acurácia de 73,68%, com uma especificidade de 51,08%, e sensibilidade de 96,26%, conseguindo um AUC 82,00%. Com isso determinamos que ao adicionar as camadas de convolução a rede começava a aprender o problema.

No modelo 3 foi adicionado 3 camadas de convolução e 3 de *pooling*, todas com filtros de tamanho 3x3 e 2x2 respectivamente. Para treinamento foram separadas 10.000 (dez mil) imagens. O modelo continha uma camada de entrada com 64 neurônios, uma camada oculta com 32 neurônios e uma camada de saída com 2 neurônios, um tamanho de lote igual a 32, com um total de 500 épocas, tendo cada época um número de passos por época igual ao número de imagens para treinamento dividido pelo tamanho do lote. Na base de testes, no modelo 3 atingiu-se uma acurácia de 79,68%, com uma especificidade de 64,77% e sensibilidade de 94,26%, conseguindo um AUC 84,00%. Ao aumentarmos o número de épocas e a quantidade de neurônios, notamos um aumento na Acurácia como também nas demais métricas.

Após a análise dos resultados, de acordo com as implementações, foi encontrado um modelo que se saia melhor em aprender o problema, e que gerava baixo custo computacional. Após acharmos a melhor arquitetura decidiu-se aumentar a quantidade de imagens de treinamento, como mostrados os resultados do modelo 4 e 5.

No modelo 4 foram utilizadas 3 camadas de convolução e 3 de *pooling*, todas com filtros de tamanho 3x3 e 2x2 respectivamente, porém com uma diferença do modelo anterior foi adicionado uma camada de normalização entre cada camada de *pooling*. Para treinamento foram separadas 100.000 (cem mil) imagens. O modelo continha uma camada de entrada com 128 neurônios, uma camada oculta com 64 neurônios e uma camada de saída com 2 neurônios, diminuímos o tamanho de lote pela metade do modelo 3 ficando igual a 16, com um total de 2500 épocas, tendo cada época um número de passos por época igual 1000. Na base de testes, no modelo 4, atingimos uma acurácia de 82,26%, com uma especificidade de 67,33% e sensibilidade de 93,85%, conseguindo um AUC 90,00%. Concluiu-se que a normalização entre cada camada fez com que a rede mostrasse uma performance de classificação mais acurada.

No modelo 5 foram utilizadas 3 camadas de convolução e 3 de *pooling*, todas com filtros de tamanho 3x3 e 2x2 respectivamente, tendo uma camada de normalização entre cada camada de *pooling*. Para treinamento utilizamos todas as imagens disponíveis na base para treinamento um total de 262.144 (duzentos e sessenta e dois mil e cento e quarenta e quatro) imagens. O modelo continha uma camada de entrada com 192 neurônios, uma camada oculta com 128 neurônios e uma camada de saída com 2 neurônios, com um total de 3000 épocas. Na base de testes, no modelo 5, atingiu-se uma acurácia de 83,63%, com uma especificidade de 72,49% e sensibilidade de 94,76%, conseguindo um AUC 93,00%. Esse método se mostrou melhor em todas as métricas depois do aumento da base de treino, e da quantidade de épocas, com auxílio das camadas de convolução e *pooling* juntamente com a camada de normalização.

16.6.1 Redes Pré-treinadas

Baseado nos resultados das redes implementadas anteriormente, foram feitos testes com arquiteturas pré-treinadas, ou seja, que já tinham um modelo e pesos definidos, arquiteturas estas que já passaram por um treinamento como dito na seção de referencial teórico.

Para a obtenção dos resultados foram usadas apenas 10.000 (dez mil) imagens da base de testes, para a extração de características, visto que, um número maior de imagens elevava muito o custo computacional. Foram usadas as redes neurais VGG16, VGG19 e a ResNet50 para a obtenção de características da imagem. Extraídas as características das imagens, utilizamos os classificadores SVM e *Random Forest*. O classificador *Random Forest* foi implementado com um total de 1000 árvores de estimativas, um número tido na literatura como ideal. O classificador SVM foi usado, em conjunto com cada uma de suas funções chamadas *kernel*, onde se testou cada uma dessas funções matemáticas para obter os resultados da classificação.

As Tabelas 107, 108, 109 mostram os resultados obtidos com as arquiteturas pré-treinadas. Cada tabela mostra o classificador utilizado e qual *kernel* selecionado. Vale ressaltar que em todas as arquiteturas utilizamos a mesma quantidade de imagens e que nos classificadores utilizou-se uma taxa de *cross-validation*, com K igual a 10.

Tabela 107 – Tabela de resultados da arquitetura pré-treinada VGG16.

Classificador	kernel	Acc.%	Esp.%	Sens.%	AUC%	Kappa
SVM	PolyKernel	89,00	88,00	90,00	90,00	0,79
SVM	RBF	89,00	88,00	89,00	79,00	0,78
SVM	Sigmoid	77,00	77,00	77,00	78,00	0,75
SVM	Linear	84,00	85,00	83,00	84,00	0,68
<i>random forest</i>	-	87,00	86,00	87,00	87,00	0,74

Tabela 108 – Tabela de resultados da arquitetura pré-treinada VGG19.

Classificador	kernel	Acc.%	Esp.%	Sens.%	AUC%	Kappa
SVM	PolyKernel	88,00	88,00	89,00	89,00	0,77
SVM	RBF	89,00	88,00	89,00	89,00	0,78
SVM	Sigmoid	81,00	81,00	79,00	81,00	0,61
SVM	Linear	83,00	84,00	82,00	90,00	0,66
<i>random forest</i>	-	86,00	86,00	86,00	87,00	0,73

Tabela 109 – Tabela de resultados da arquitetura pré-treinada ResNet50.

Classificador	kernel	Acc.%	Esp.%	Sens.%	AUC%	Kappa
SVM	PolyKernel	90,00	89,00	91,00	90,00	0,80
SVM	RBF	90,00	89,00	91,00	90,00	0,80
SVM	Sigmoid	75,00	77,00	74,00	76,00	0,51
SVM	Linear	87,00	88,00	87,00	88,00	0,75
<i>random forest</i>	-	88,00	88,00	87,00	88,00	0,76

Como se percebe, os resultados descritos na Tabela 6 foram os melhores resultados e apresentam valores de acurácia ótimos de 90,00 % na taxa média de acertos, 91,00% para sensibilidade média e 89,00% para especificidade. Sendo que o maior valor alcançado dentre todos os experimentos foi de acurácia 90,0%, também apresentando os bons resultados de 90,00% para AUC e 0,8 para o coeficiente *kappa*, utilizando a arquitetura pré-treinada ResNet50 e usando o classificador SVM juntamente com o Kernel RBF ou *PolyKernel*. Com base nos resultados obtidos, constata-se que a arquitetura e o classificador configurado com umas das duas funções conseguiu os resultados mais promissores e equilibrados em comparação as outras arquiteturas. Já a arquitetura VGG16 apresentou no geral menores resultados em comparação as outras arquiteturas.

16.7 Fine-tuning

Com o intuito de melhorar os resultados obtidos, tanto na arquitetura criada do zero, como nos modelos pré-treinados utilizados, recorremos à técnica de *fine-tuning*.

Realizamos testes com as arquiteturas pré-treinadas, ou seja, as redes VGG 16 e 19 e também a ResNet50 todas elas em conjunto com as camadas da rede treinada do zero. A Tabela 110 mostra os resultados obtido com a técnica *fine-tuning*, onde estão descritos modelo, que se refere a arquitetura utilizada para ser treinada com a nova base.

Tabela 110 – Tabela de resultados da aplicação da técnica de *fine-tuning*

Modelo	Acc.%	Esp.%	Sens.%	AUC%	Kappa
VGG16	84,99	83,58	86,40	93,00	0,69
VGG19	86,12	76,83	95,41	96,00	0,72
ResNet50	81,93	67,62	96,24	91,00	0,63

Como se percebe, os resultados da técnica *fine-tuning*, foram bem próximos dos resultados da rede treinada a partir do zero, considerando como métricas principais a Acurácia e a AUC. A técnica usada aqui mostrou uma acurácia média, em relação com as outras duas estratégias testadas, ficando maior que o melhor resultado da arquitetura implementada do zero e menor que o melhor resultado da rede pré-treinada. A Tabela 111 mostra os melhores resultados de cada uma das estratégias testadas. Para comparação dos resultados decidiu-se adotar as seguintes métricas, com base na métrica exigida no desafio lançado em março de 2019 pela grand-challenge.org, que mede a precisão do modelo a partir dos resultados da métrica da AUC.

Tabela 111 – Tabela dos melhores resultados das estratégias testadas.

Estratégia	Acc.%	AUC%	Kappa
Rede do zero	83,63	93,00	0,67
Pré-treinada Resnet50	90,00	90,00	0,80
<i>fine-tuning</i> com VGG19	86,12	96,00	0,72

Considerando a métrica utilizada no desafio que é a Área Sob a Curva ROC a estratégia de *fine-tuning* conseguiu um resultado ótimo podendo se colocar como um dos 12 melhores do desafio da *grand-challenge.org*, alcançando um AUC de 96,00%. Já o resultado da arquitetura treinada do zero ficou colocada na vigésima terceira posição do desafio. A Tabela 112 mostra os principais resultados submetidos para o desafio juntamente com o melhor resultado obtido nesta pesquisa.

Tabela 112 – Tabela de comparação dos resultados deste trabalho com os resultados do desafio *fine-tuning*.

Equipe	Posição	AUC%
<i>zhongqiaoyong (Dr. Strange)</i>	DenseNet201	97.37
GeertLitjens	DenseNet with TTA	96.91
<i>ISMI19-KarmaCamelyon</i>	NASNET	96.23
<i>e.boeijenk (ISMI19-KarmaCamelyon)</i>	Ensemble convnet, nasnet, recnn	96.20
Este trabalho	<i>fine-tuning com VGG19</i>	96.00

16.8 Conclusão

Neste Capítulo foram apresentadas e testadas três estratégias de *deep learning* para classificação de imagens histopatológicas para identificação do câncer metastático. A estratégia de melhor resultado foi a que aplicamos a técnica de *fine-tuning* que ao ser treinada com os pesos da rede pré-treinada juntamente com as camadas da rede própria implementada do zero, obteve como melhor resultado na base de teste chegando a 96,00% na área sob a curva ROC e uma sensibilidade de 95,41%, que é um desempenho melhor que os

resultados obtidos por um patologista humano que conseguiu 72.00% de sensibilidade de acordo com (VEELING et al., 2018). Com base nos resultados obtidos o método apresentado mostra boa eficácia no âmbito de classificação automática de imagens.

Dessa maneira este Capítulo apresenta contribuições em duas áreas de estudo. Na área da saúde oferecendo uma metodologia como segunda opinião a profissionais patologistas. Esse método pode ajudar no diagnóstico de tumores metastáticos em imagens histopatológicas de seções linfonodais. A outra área é a da computação, fornecendo informações importantes de processamento de imagens para auxiliar pesquisas futuras. Em trabalhos futuros retende-se testar novos modelos de extração de características como também de classificação, tratando como trabalhos futuros: utilizar outras bases de pesos de redes neurais pré-treinadas, para comparação de resultados e testar outras estratégias de classificação de imagens, de modo a melhorar os resultados.

Redes Neurais Convolucionais Aplicadas na Detecção de Carcinoma Mamário em Imagens Histológicas

Autores: Mateus Feitosa Lima; Thiago José Barbosa Lima; Flávio Henrique Duarte de Araújo.

Resumo: O Câncer de Mama é um dos tipos de cânceres mais comuns entre as mulheres no Brasil e no mundo, respondendo por cerca de 28% dos novos casos a cada ano. Estatísticas indicam aumento da sua incidência tanto nos países desenvolvidos quanto nos em desenvolvimento. Quando os pacientes são diagnosticados com o câncer, precisam saber se o mesmo mostra-se invasivo ou não, no sentido de ocorrência da metástase e essa informação pode ser obtida por meio de alguns exames, por exemplo, biopsia na qual são realizados testes histológicos para confirmar o potencial cancerígeno. Nesse trabalho desenvolvemos um método computacional para classificação de imagens histológicas diferenciando carcinoma invasivo do não invasivo (*in situ*), para tal método testamos diferentes arquiteturas de Redes Neurais Convolucionais que são: ResNet, VGG19, VGG16, Xception, InceptionV3. Além destas arquiteturas também utilizamos descritores clássicos da literatura como: HOG - (*Histogram of Oriented Gradients*), LBP - (*Local Binary Patterns*) e Dayse. Além disso, a Transferência de Aprendizado foi empregada para lidar com a limitação da pequena quantidade de imagens na base de dados. O conjunto de imagens utilizado foi o de microscopia de histologia da mama corados com hematoxilina e eosina que é fornecido como parte do Desafio ICIAR 2018 em Imagens Histológicas de Câncer de Mama. Com base nos testes realizados, a ResNet foi a arquitetura que obteve o melhor resultado, com acurácia de 96%, AUC 96%, sensibilidade 95,5% e especificidade de 96,4%, superando as demais arquiteturas e descritores avaliados.

Keywords: Câncer de Mama; Descritores de imagem; Aprendizado Profundo; Transferência de Aprendizado.

17.1 Introdução

O câncer de mama é um tumor maligno que se desenvolve na mesma como consequência de alterações genéticas em algum conjunto de células mamárias, que passam a se dividir descontroladamente. A proporção em homens e mulheres é de 1:100 ou seja, para cada 100 mulheres com câncer, um homem terá a doença.

Para o ano de 2018 foram estimados 59.700 casos novos, que representam uma taxa de incidência de 51,29 casos por 100.000 mulheres (INCA, 2018a). Segundo dados da (SBM, 2018), a cada ano, aproximadamente 60 mil brasileiras são diagnosticadas com câncer de mama, onde a incidência em mulheres mais jovens nos últimos anos tem chamado a atenção dos pesquisadores.

A maioria dos cânceres de mama são carcinomas e (VIEIRA et al., 2008) afirma que esse tipo é a neoplasia maligna mais frequente em mulheres. Os carcinomas são tumores que começam nas células epiteliais que revestem órgãos e tecidos por todo o corpo. Às vezes, um termo ainda mais específico é usado, por exemplo, a maioria dos cânceres de mama é um tipo de carcinoma chamado adenocarcinoma. Este termo refere-se a um tipo de cancro que começa nas células do tecido glandular que tem uma função secretora.

O carcinoma de mama é geralmente classificado por sua aparência histológica e origina-se do epitélio de revestimento interno dos ductos ou dos lóbulos que suprem os ductos com leite (MAKKI, 2015). Para o estudo morfológico do carcinoma de mama uma das principais informações avaliadas pelo patologista é a presença ou não de uma característica conhecida com invasão da membrana basal que é o rompimento dos limites dos ductos e lóbulos que dispersam-se, invadindo outros tecidos da mama. De acordo com as características que o médico patologista encontre, o tratamento pode mudar de pessoa para pessoa.

Ainda não é possível prevenir o câncer de mama em função da multiplicidade de fatores relacionados ao surgimento da doença, desta forma, o diagnóstico precoce é a melhor forma de combater a doença dentre os métodos para detecção e classificação de câncer de mama. (TRAMONTE et al., 2016) aponta que o diagnóstico do câncer de mama, em países com recursos limitados, é frequentemente feito em um estágio já avançado.

Especificamente, durante o procedimento de diagnóstico, os especialistas avaliam a organização do tecido por meio de imagens microscópicas. No entanto, a grande quantidade de dados e a complexidade das imagens tornam essa tarefa custosa e não trivial. Por causa disso, o desenvolvimento de ferramentas automáticas de diagnóstico é desafiador, mas também essencial para o campo da saúde.

(MIRANDA, 2011) afirma que o método tradicional de inspeção microscópica é importantíssimo em analisar a estrutura do tecido. No entanto, esta técnica de inspeção manual é altamente demorada e propensa a erros manuais, principalmente no processo de classificação dos carcinomas *in situ* e invasivos, pois é nesse diagnóstico que é fornecido a informação de espalhamento de células cancerosas para os tecidos circundantes a partir de seu ponto de ocorrência (INCA, 2011).

Em vista do contexto apresentado, este trabalho tem como objetivo geral o desenvolvimento de um método computacional para a classificação automática de imagens histológicas em carcinoma invasivo e não invasivo. Para a concretização do objetivo geral, tem-se os seguintes objetivos específicos:

- Avaliar diferentes descritores de imagens para a extração de características das mesmas.
- Avaliar a viabilidade do uso das CNNs por meio do teste de diferentes arquiteturas de redes e o uso da Transferência de Aprendizado nas arquiteturas aplicadas ao problema descrito.

17.2 Trabalhos Relacionados

No que concerne aos trabalhos desta linha de pesquisa, pode-se observar uma tendência de crescimento no uso de técnicas de Aprendizado Profundo, tanto para extração das características, quanto para classificação de imagens médicas.

17.2.1 Descrição dos Trabalhos

Estudos conhecidos sobre o problema da classificação de imagens histológicas vêm se tornando relevantes com o passar dos anos e se popularizando no meio científico. Com isso, a Tabela 113 mostra um resumo das principais informações dos trabalhos relacionados para a classificação de imagens histológicas da mama.

(GOLATKAR; ANAND; SETHI, 2018) propuseram um método baseado na Aprendizagem Profunda, onde sua estratégia é extrair manchas com base na densidade de núcleos em vez de amostragem aleatória ou em grade, juntamente com a rejeição de patches que não são ricos em regiões de núcleos. O conjunto de imagens

Tabela 113 – Tabela contendo a comparação entre os trabalhos de classificação de imagens histológicas.

Referências	Base de Imagens	Algorítmos	Resultados
(ARAÚJO et al., 2017)	ICIAR 2015	CNN Própria	77,8% 4 classes e 83,3% em 2
(GOLATKAR; ANAND; SETHI, 2018)	ICIAR 2018	Inception-v3	85% 4 classes e 93% em 2
(RAKHLIN et al., 2018)	ICIAR 2018	ResNet, VGG e Inception	87,2% 4 classes e 93% em 2
(VESAL et al., 2018)	ICIAR 2018	Inception-V3 e ResNet	97.5% 4 classes
(WANG et al., 2018)	(INESC TEC), i3S, Universidade do Porto	VGG	85% 4 classes
(GUO et al., 2018)	ICIAR 2018	GoogleNet	87.5% 4 classes

empregadas nesse trabalho é o da ICIAR 2018 (*International Conference on Image Analysis and Recognition*) que são fornecidas como parte do desafio de imagens histológicas mamárias em quatro subtipos de tecido, são elas: tecidos normais, lesões benignas, carcinomas *in situ* e invasivos. O método proposto obteve uma acurácia média de 85 % nas quatro classes: e uma acurácia média de 93% em duas classes (sem câncer versus carcinoma).

(RAKHLIN et al., 2018) propuseram uma abordagem computacional baseada em redes neurais profundas, onde são utilizadas algumas arquiteturas de redes (ResNet, VGG e Inception) pré-treinadas na base ImageNet. Na tarefa de classificação nas quatro classes da base de imagens da ICAR 2018, o método proposto alcançou 87,2% de precisão. Para a tarefa de classificação de 2 classes para detectar carcinomas, os autores relataram 93,8% de precisão, AUC 97,3% e sensibilidade de 96,5 e especificidade de 88,0%.

(VESAL et al., 2018) utilizaram uma abordagem baseada na Transferência de Aprendizado, para a tarefa de classificação da imagem histológica da mama em quatro subtipos de tecido, são elas: tecidos normais, lesões benignas, carcinomas *in situ* e invasivos. As imagens histológicas, fornecidas como parte do desafio ICIAR 2018, foram primeiramente normalizados para corrigir variações de cor resultantes de inconsistências durante a preparação da lâmina. Eles conseguiram com a rede ResNet50 (com base na aprendizagem residual) uma precisão de classificação de 97,50% para quatro classes, superando a rede Inception-V3 que alcançou uma precisão de 91,25%.

(ARAÚJO et al., 2017) propuseram um método para a classificação de imagens de biópsia da mama, onde é projetada uma arquitetura de rede para recuperar informações em diferentes escalas, incluindo organização de núcleos e tecidos em geral. As características extraídas pela CNN também são usadas para treinar um classificador *Support Vector Machine*. O método proposto alcançou acurácia de 77,8% para quatro classes da base de imagens ICIAR 2015 (sendo esse um conjunto de treinamento estendido de 249 imagens e um conjunto de teste separado de 20 imagens) e 83,3% para carcinoma / não carcinoma.

(WANG et al., 2018) propuseram um método de CAD (*computer-aided diagnosis*), que é um sistema que auxilia os médicos na interpretação de imagens médicas, esse sistema trabalha com pool global (usada para extração de informações globais a partir de mapas de recursos de alta resolução) diretamente nas imagens de entrada e o mesmo emprega a técnica de aumento de dados. O conjunto de imagens utilizadas nesse trabalho é fornecido pela Universidade do Porto, pelo Instituto de Engenharia e Sistemas e Computadores, Tecnologia e Ciência (INESC TEC) e pelo Instituto de Investigação e Inovação em Saúde (i3S) contendo 400 imagens de alta resolução com aumento de 200x em cada imagem. O método proposto relatou acurácia de 85% em imagens rotuladas com quatro classes: Normal, Benign, *In situ*, e Invasivo.

(GUO et al., 2018) propuseram uma estratégia que utiliza CNN híbrida que é projetada com base na arquitetura GoogLeNet para mesclar mais informações importantes na decisão, onde a Transferência de Aprendizado e aumento de dados foram empregados para lidar com as limitações da pequena quantidade de imagens na base ICIAR 2018. A precisão de classificação do método adotado é de 87,5% para as quatro classes dessa base.

Pesquisas efetuadas mostraram que as técnicas de Aprendizagem Profunda estão se tornando primordiais na resolução de diagnóstico médico por meio de imagens, por isso resolvemos investigar o uso dessas técnicas envolvendo algumas arquiteturas de redes convolucionais. Também decidimos testar outros descriptores clássicos para confirmar se as Redes Neurais Convencionais realmente apresentavam um melhor desempenho com uma quantidade de imagens reduzidas. A seguir serão descritos os métodos executados nesse trabalho.

O principal diferencial desse trabalho é a utilização dos descriptores de textura e também das Redes Neurais Convencionais, sendo realizada uma comparação entre eles na etapa de extração de características das imagens. Além disso é empregado dois métodos de divisão da base de imagens sendo eles: *cross validation* e o método de divisão (75/25) para treino e teste.

17.3 Materiais e Métodos

Nessa Seção são apresentados a base de imagens, conceitos importantes para uma melhor compreensão deste trabalho, as principais arquiteturas de CNNs e descriptores clássicos utilizados para o processamento das imagens histológicas.

17.3.1 Base de Dados

O conjunto de imagens utilizadas nesse trabalho é composto por imagens de alta resolução (2040 x 1536 pixels) e modelo de cores: RGB, descomprimidas e anotadas com H&E, que é a principal técnica de coloração de tecidos em histologia. As imagens empregadas no projeto fazem parte do desafio de classificação da ICIAR 2018 (*International Conference on Image Analysis and Recognition*), que é uma conferência que visa reunir pesquisadores e profissionais na área de processamento de imagem. Todas as imagens são digitalizadas com ampliação de 200 vezes e tamanho de pixel de $0,42\mu \times 0,42\mu$. Essa base contém um total de 400 imagens microscópicas, distribuídas da seguinte forma: carcinoma in situ, carcinoma invasivo, tecido normal e lesão benigna, sendo 100 imagens para cada classe.

A rotulagem dessa base foi realizada por dois patologistas, que apenas forneceram um diagnóstico a partir do conteúdo da imagem, sem especificar a área de interesse para a classificação. Casos de desacordo entre especialistas foram descartados. O objetivo do desafio é fornecer uma classificação automática das quatro classes mencionadas.

Tradicionalmente as mais variadas tecnologias são utilizadas para oferecer um ótimo desempenho no processamento de imagens e uma dessas está rapidamente se tornando um dos mais estudados e procurados campos dentro da ciência da computação moderna, onde ela oferece facilidades e poder computacional, tal tecnologia é denominada Aprendizagem Profunda e ela será apresentada a seguir.

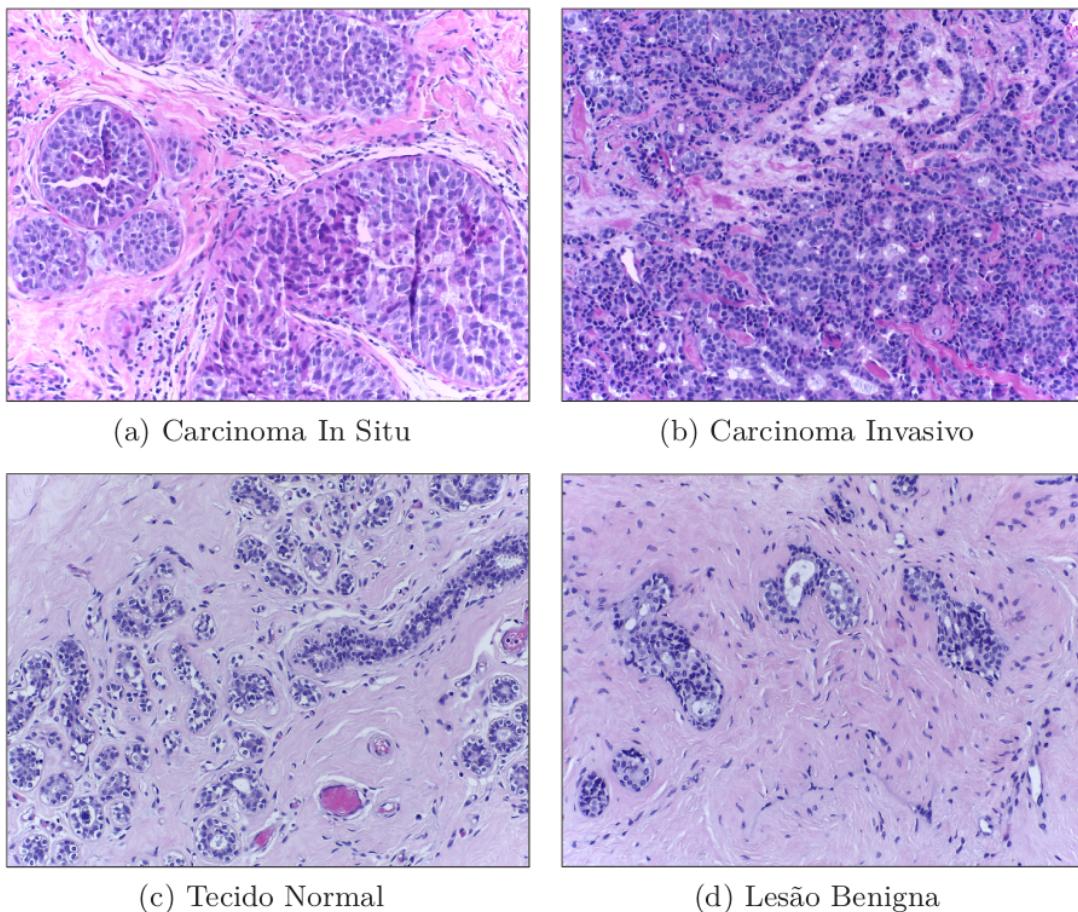
17.3.2 Aprendizagem Profunda

A Aprendizagem Profunda é uma sub área do aprendizado de máquina que trata de modelos vagamente inspirados no cérebro humano. Tecnicamente, toda Rede Neural com mais de uma camada escondida é chamada de Rede Neural Profunda e tudo que ela aprende é por meio do Aprendizado Profundo. Para (LECUN; BENGIO; HINTON, 2015) a Aprendizagem Profunda é uma área que contém uma pilha de multicamadas simples, todos (ou a maioria) estão sujeitos à aprendizagem, e muitos dos quais calculam mapeamentos de entrada-saída não lineares.

Tradicionalmente, a qualidade dos algoritmos depende muito da representação dos dados em certas características (as chamadas *features*). Assim, a criação recursos válidos e pré-processamento consomem grande parte dos esforços dos especialistas. Já os algoritmos do tipo Aprendizado Profundo têm uma abordagem inovadora, pois dispensam grande parte desse pré-processamento e geram automaticamente propriedades invariantes em suas camadas hierárquicas de representação (HELABS, 2019).

(BRIGADE, 2016) afirma que atualmente o reconhecimento de imagens por máquinas treinadas através de Aprendizado Profundo em alguns cenários possuem uma taxa de acerto maior que a de humanos. Várias arquiteturas de Aprendizagem Profunda, tais como Redes Neurais Convencionais têm sido aplicadas em

Figura 125 – Ilustração dos exemplos contidos na base de imagens ICIAR 2018. Fonte: Banco de Dados da ICIAR 2018.



áreas como visão computacional, reconhecimento automático de fala, processamento de linguagem natural, reconhecimento de áudio e bioinformática, onde elas têm se mostrado capazes de produzir resultados significantes.

A seguir será abordado as fundamentações sobre as redes neurais convolucionais utilizadas nesse projeto.

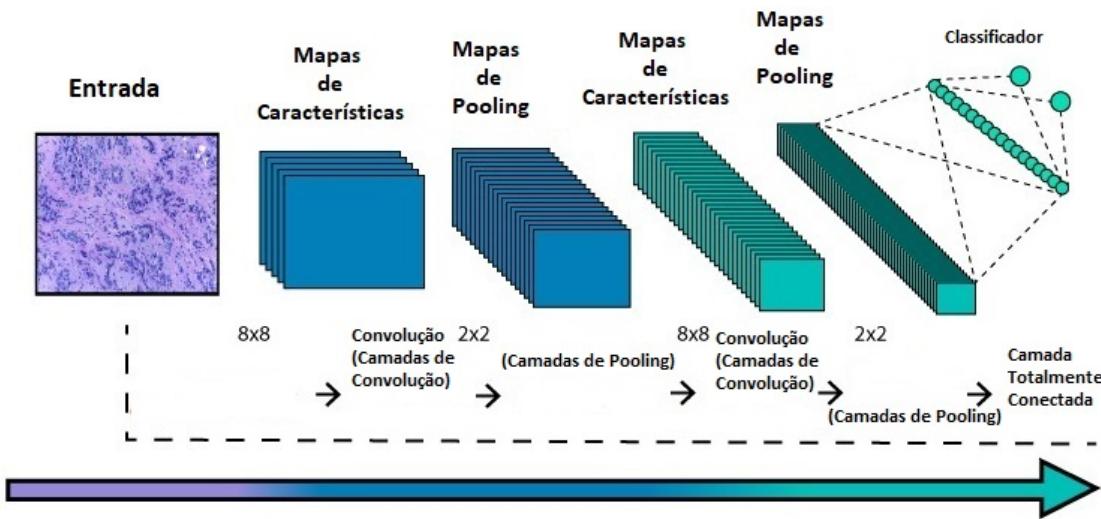
17.3.2.1 Redes Neurais Convolucionais

Redes Neurais Convolucionais, também conhecidas como CNN (do inglês, *Convolutional Neural Network*) ou ConvNets são uma categoria de redes neurais usadas no reconhecimento e processamento de imagens que é especificamente projetada para processar dados de pixel. (PONTI et al., 2018) versa que sua característica marcante é ser composta basicamente de camadas convolucionais, que processa as entradas considerando campos locais. Essas redes constituem uma escolha de topologia ou arquitetura projetadas para reduzir o número de parâmetros a serem aprendidos otimizando o tempo de treinamento.

As CNNs são projetadas para processar dados que vêm na forma de múltiplos arrays, por exemplo, uma imagem colorida composta por três matrizes contendo intensidades de pixel nos três canais de cores (LECUN; BENGIO; HINTON, 2015). Elas são capazes de capturar com sucesso as dependências espaciais e temporais em uma imagem através da aplicação de filtros. (KOVALESKI, 2018) afirma que a mesma se utiliza desses filtros para analisar pequenos grupos de dados, extraíndo suas características mais relevantes e relata que apenas estas características são passadas adiante para as próximas camadas, como apresentado na Figura 126.

Ao contrário de uma Rede Neural regular, as camadas de uma CNN possuem neurônios dispostos em

Figura 126 – Ilustração da arquitetura de uma CNN com duas camadas convolucionais, duas de pooling, uma totalmente conectada e a de saída. Fonte: Adaptação de ([SAHA, 2018](#)).



3 dimensões: largura, altura e profundidade. Além disso, os neurônios em uma camada não se conectam a todos os neurônios da camada seguinte, mas apenas a uma pequena região dela. Por fim, a saída final será reduzida a um único vetor de pontuações de probabilidade, organizadas ao longo da dimensão de profundidade.

A arquitetura executa um melhor ajuste ao conjunto de dados da imagem devido à redução no número de parâmetros envolvidos e à capacidade de reutilização dos pesos. A grande vantagem da aplicação das CNNs para o Processamento Digital de Imagens é sua grande capacidade de extrair características, por trabalhar com convoluções, ou seja, criando “sub-quadros de bits” de forma que seja possível realçar contornos e áreas de interesse de cada imagem ([ROSEBROCK, 2017b](#)).

A seguir será apresentada as fundamentações e principais características das mais diversas arquiteturas de redes convolucionais utilizados nesse trabalho para a extração de características de textura.

17.3.2.1.1 Arquitetura das Redes Avaliadas

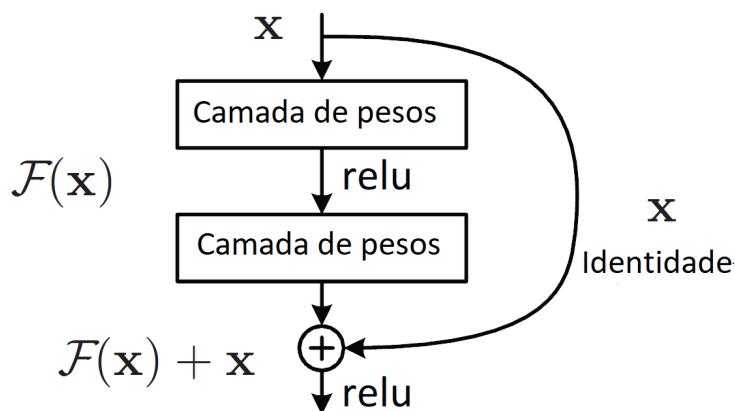
Após analisar a bibliografia que resultou da pesquisa a respeito do tema, foram encontrados diversos modelos de arquitetura compatíveis com a proposta desse trabalho. Com isso, foi possível selecionar cinco principais modelos para estudo mais profundo e consequente avaliação da melhor performance. Tais modelos são descritos à seguir.

1. Modelo 1 - ResNet

A ResNet é uma arquitetura que tem como principal característica suas conexões residuais. Na aprendizagem residual, em vez de tentar aprender algumas características, tal arquitetura é voltada para aprender alguns resíduos. Residual pode ser simplesmente entendido como subtração de características aprendidas da entrada dessa camada e a ResNet faz isso usando conexões de atalho, que são conexões em que o gradiente é diretamente retropropagado para camadas anteriores. Na Figura 127 é demonstrado o funcionamento de um bloco residual.

Um dos problemas que a ResNet resolve é o famoso desaparecimento de gradiente. Isso ocorre porque, quando a rede é muito profunda, os gradientes de onde a função de perda é calculada diminuem facilmente para zero. Este resultado nos pesos nunca atualiza seus valores e, portanto, nenhum aprendizado será realizado.

Figura 127 – Ilustração de um bloco residual na arquitetura da rede ResNet. Fonte: Adaptação de (HE et al., 2016).



Com tal arquitetura, os gradientes podem fluir diretamente através das conexões de salto. Para (CHEN et al., 2017) a ResNet fornece uma estrutura de treinamento para facilitar formação de redes substancialmente mais profundas que as utilizadas anteriormente.

2. Modelo 2 - Visual Geometry Group - VGG

A VGG é um modelo de CNN inicialmente proposto por (SIMONYAN; ZISSERMAN, 2014a). A mesma garantiu a primeira e segunda posições nas tarefas de localização e classificação, respectivamente, no desafio ImageNet 2014. Ela faz a melhoria sobre o AlexNet construída por (Krizhevsky; Sutskever; Hinton, 2012) substituindo grandes filtros do kernel por múltiplos filtros do tamanho 3x3 um após o outro. Nela as profundidades mais comuns são as de 16 e 19 camadas de profundidade.

Com a introdução da VGG, elas trouxeram alguns padrões: sugeriu-se que todos os filtros tivessem tamanho de 3x3, pools máximos deveriam ser colocados após cada 2 convoluções e o número de filtros deveria ser dobrado após cada max-pooling. A VGGNet é considerada a rede preferida pela comunidade para Transferência de Aprendizado, pois sua arquitetura uniforme é boa para extrair características de imagens. Assim, a VGG tem sido utilizada como modelo na construção de novas redes, como Unet, TernausNet e várias outras redes.

3. Modelo 4 - Xception

A arquitetura Xception foi proposta por (CHOLLET, 2017) e significa *Extreme version of Inception* sendo uma versão da arquitetura Inception, onde essa arquitetura tem 36 camadas convolucionais formando a base de extração de características da rede. A sua principal característica é que ela substitui o módulo inception por convoluções separáveis em profundidade, isso faz com que o número de conexões sejam menores e o modelo se torne mais leve.

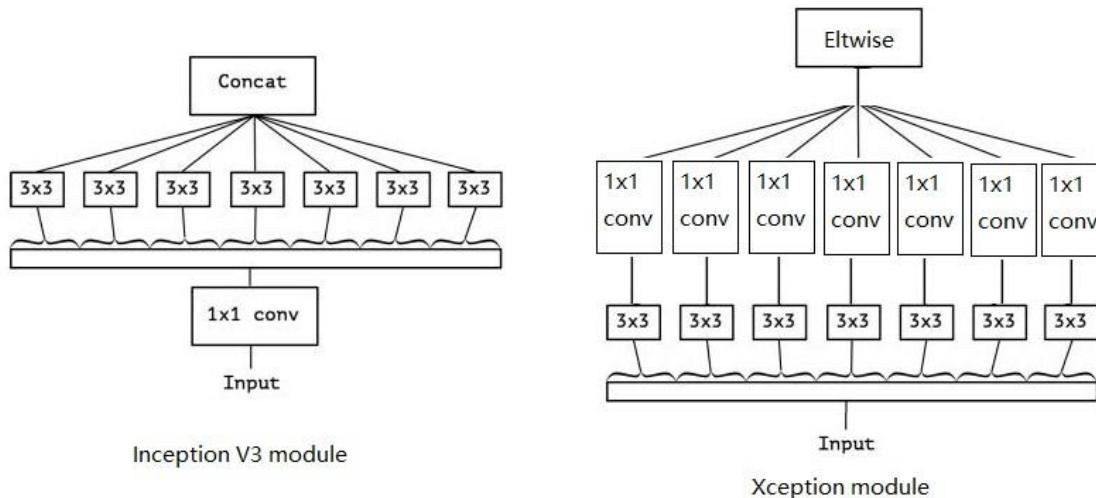
A Figura 128 mostra que a arquitetura Xception em vez de partitionar dados de entrada em vários blocos compactados, a mesma leva isso um passo adiante, ela mapeia as correlações espaciais para cada canal de saída separadamente e, em seguida, executa uma convolução de 1x1 em profundidade para capturar a correlação entre canais.

4. Modelo 5 - Inception V3

O Inception V3 baseia-se no artigo original (SZEGEDY et al., 2016) só que com algumas melhorias desenvolvidas pelos pesquisadores. Os mesmos observaram que os classificadores auxiliares não contribuíram muito até o final do processo de treinamento, quando as precisões estavam se aproximando da saturação fazendo que o modelo não fosse tão eficiente.

Com tal problema a arquitetura passou a utilizar suavização de rótulo (um tipo de componente de regularização adicionado à fórmula de perda que impede a rede de se tornar muito confiante em

Figura 128 – Diferença na ordem das operações de convolução da Inception e Xception. Fonte: ([PROGRAMMERSOUGHT, 2017](#)).



relação a uma classe. Impede o ajuste excessivo), convoluções de 7x7 fatoradas e outras melhorias. O modelo em si é composto de blocos de construção simétricos e assimétricos, incluindo convoluções, agrupamentos médios, agrupamentos máximos, concats, dropouts e camadas totalmente conectadas.

A quantidade de imagens é um fator preponderante para treinamento de uma CNN, mas às vezes nem sempre é possível lidar com bases de grande escala e para contornar tal empecilho uma técnica chamada de transferência de aprendizado vem ganhando força, pois ela torna os modelos que a usam eficientes na manipulação de dados insuficientes. A seguir serão abordados pontos relevantes sobre a Transferência de Aprendizado.

17.3.3 Transferência de Aprendizado

A Transferência de Aprendizado é um método de aprendizado de máquina em que um modelo desenvolvido para uma tarefa é reutilizado como ponto de partida para um modelo em uma segunda tarefa([PONTI et al., 2018](#)). É uma abordagem popular em Aprendizado Profundo na qual os modelos pré-treinados são usados como fase inicial em tarefas de visão computacional.

Normalmente, precisa-se de muitos dados para treinar uma Rede Neural, mas nem sempre é possível ter acesso a dados suficientes. É aí que a Transferência de Aprendizado entra em jogo, porque com ela pode-se construir um modelo de aprendizado de máquina sólida com comparativamente poucos dados de treinamento, porque o modelo já é pré-treinado ([TORREY; SHAVLIK, 2010](#)).

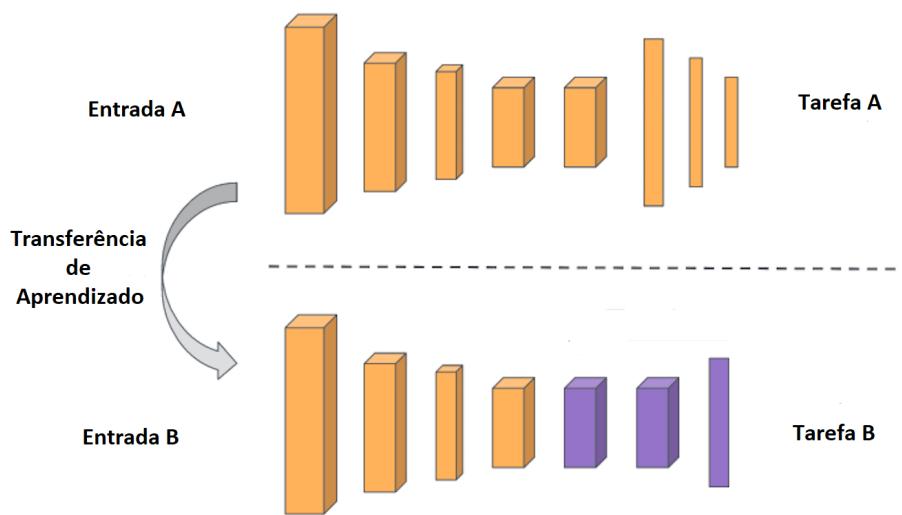
Embora as redes profundas tenham mostrado-se promissoras na representação de informações, tais modelos profundos supervisionados, exigem grandes quantidades de dados rotulados e recursos de computação, que são caros ([DU et al., 2017b](#)). O processo de Transferência de Aprendizado é demonstrado pela Figura 129

Existem duas abordagens principais para implementar na Transferência de Aprendizado; são elas: inicialização de peso e extração de características ([BROWNLEE, 2019](#)). No caso da primeira abordagem os pesos nas camadas reutilizadas podem ser usados como ponto de partida para o processo de treinamento e adaptados em resposta ao novo problema.

Na segunda abordagem pode-se remover a camada de saída (aquele que fornece as probabilidades de estar em cada uma das 1000 classes) e, em seguida, usar toda a rede como um extrator de características fixo para o novo conjunto de imagens. Depois disso um classificador externo é usado para classificação das imagens. Nesse trabalho utilizamos a segunda abordagem envolvendo os classificadores (*Random Forest* e *SVM*), por causa que os mesmos eram os mais abordados nos trabalhos pesquisados.

Portanto, ela também economiza muito tempo de treinamento, porque às vezes pode levar dias ou até semanas para treinar uma Rede Neural Profunda do zero em uma tarefa complexa. Além desse be-

Figura 129 – Exemplo de Transferência de Aprendizado. Fonte: Adaptação de ([SUBODH, 2018](#)).



nenfício a Transferência de Aprendizado resulta em menor erro de generalização ([GOODFELLOW; BENGIO; COURVILLE, 2016](#)).

Na próxima Seção serão apresentadas as fundamentações dos descritores textuais utilizados nesse projeto, ditos anteriormente como clássicos para comparar se as CNNs realmente produzem melhores resultados sobre os tais descritores.

17.3.4 Descritores Clássicos

São descrições das características visuais do conteúdo em imagens e representam recursos homogêneos em uma imagem ou subimagem. Eles descrevem características elementares, como a forma, a cor, a textura. A seguir serão apresentados os descritores escolhidos para o trabalho proposto que são eles:

1. *Histogram of Oriented Gradients - HOG*

A implementação desse descritor exige a divisão da imagem em pequenas regiões conectadas, chamadas de células, e, em seguida, para cada célula, o cálculo de um histograma de gradientes orientados para os pixels dentro de cada célula. Pode-se então acumular esses histogramas através de múltiplas células para formar nosso vetor de características.

No descritor de característica HOG, a distribuição (histogramas) de direções de gradientes (gradientes orientados) é usada como características. Os gradientes (derivadas x e y) de uma imagem são úteis porque a magnitude dos gradientes é grande em torno de bordas e cantos (regiões de mudanças bruscas de intensidade) e sabemos que bordas e cantos contêm muito mais informações sobre a forma do objeto do que as regiões planas ([MALLICK, 2016](#)).

O descritor HOG é então o vetor concatenado dos componentes dos histogramas de células normalizadas de todas as regiões do bloco. Esses blocos normalmente se sobrepõem, o que significa que cada célula contribui mais de uma vez para o descritor final. Existem duas geometrias de blocos principais: blocos R-HOG retangulares e blocos C-HOG circulares ([PRATES, 2014](#)).

2. *Local Binary Patterns - LBP*

O LBP é um tipo de descritor visual usado para classificação em visão computacional. O LBP calcula uma representação local da textura e essa representação local é construída comparando cada pixel com sua vizinhança de pixels. ([TOPI et al., 2000](#)) descreve que o LBP contém a densidade de cada

recurso em uma região e que, para cada pixel, a saída de apenas a melhor correspondência de recurso é utilizado.

O uso de uma vizinhança circular e a interpolação bilinear de valores em coordenadas de pixel não inteiro permitem qualquer raio e número de pixels na vizinhança (HUANG et al., 2011). Outra extensão importante é a definição dos chamados padrões uniformes, que pode ser usado para reduzir o comprimento do vetor de recursos e implementar um descritor simples de rotação invariante. Um padrão binário local é chamado uniforme se o padrão binário contiver no máximo duas transições bit a bit de 0 a 1 ou vice-versa

3. Dense Daisy

O descritor de imagem local Daisy é baseado em histogramas de orientação de gradiente semelhantes ao descritor SIFT (*Scale Invariant Feature Transform*), que é usado para detectar e descrever recursos locais de imagens. Ele é formulado de uma maneira que permite a extração rápida e densa. (ZHU; BICHOT; CHEN, 2011) afirma que o Daisy usa uma configuração de vizinhança circular em vez da retangular, além do mais a parte do descritor é feito pela concatenação de todos os vetores do círculos, depois que eles são normalizados para a norma da unidade.

O descritor Daisy usa o histograma de gradiente filtrado pelo filtro de Gauss como uma descrição de recurso, para que o desfoque da imagem tenha um efeito mais proeminente (LI; ZIC, 2014). Além disso, eles são invariantes a rotação sem depender de uma orientação de referência, melhorando ainda mais sua robustez. Ele pode ser aplicado densamente ou aplicado a remendos extraídos em torno de pontos de interesse.

A escolha dos descritores se deve por que os mesmos são capazes de trabalhar com a imagem inteira sem precisar de nenhum processo de segmentação para utilização das imagens, partindo assim em ponto de igualdade com as arquiteturas de CNNs escolhidas anteriormente.

17.3.5 Métricas de Avaliação

Para a avaliação do método proposto nesse trabalho foram utilizadas as seguintes métricas: Sensibilidade (S), Especificidade (E), Acurácia (A), Área Sob a Curva ROC (AUC) e *Kappa* (K). Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em função de quatro valores, sendo eles: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

1. Acurácia

Esta é a métrica mais simples. Consiste basicamente o número de acertos (positivos) dividido pelo número total de exemplos. Essa métrica é mais apropriada para bases com a mesma proporção de exemplos para cada classe, e quando as penalidades de acerto e erro para cada classe forem as mesmas (MISHRA, 2018). Em problemas com classes desproporcionais, ela causa uma falsa impressão de bom desempenho. A acurácia pode ser calculada conforme a Equação 17.1.

$$A = \frac{VP + VN}{VP + VN + FP + FN}. \quad (17.1)$$

2. Sensibilidade

A sensibilidade é uma medida da proporção de casos positivos reais que foram previstos como positivos (ou positivos verdadeiros). Em outras palavras, um teste altamente sensível é aquele que identifica corretamente pacientes com uma doença, o mesmo pode ser útil para descartar uma doença se uma pessoa tiver um resultado negativo (KUMAR, 2018). A sensibilidade pode ser calculada conforme a Equação 17.2

$$S = \frac{VP}{VP + FN}. \quad (17.2)$$

3. Especificidade

A especificidade de um teste (também chamado de Taxa Negativa Verdadeira) é a proporção de pessoas sem a doença que terão um resultado negativo. Em outras palavras, a especificidade de um teste se refere a quão bem um teste identifica pacientes que não têm uma doença (KUMAR, 2018). A especificidade pode ser calculada com a Equação 17.3

$$E = \frac{VN}{VN + FP}. \quad (17.3)$$

4. Kappa

O coeficiente *Kappa* é uma medida de concordância estatística formulada por Jacob Cohen, tal métrica mede concordância entre dois avaliadores que classificam cada N itens em C categorias mutuamente exclusivas. O *Kappa* de Cohen é forte dependente de vários fatores, incluindo, por exemplo, número de classes e número de exemplos a serem classificados. O *Kappa* pode ser calculado conforme a Equação 17.4

$$K = \frac{\theta_1 - \theta_2}{1 - \theta_2}, \quad (17.4)$$

onde

$$\theta_1 = \frac{VP + VN}{VP + VN + FP + FN}, \quad (17.5)$$

e

$$\theta_2 = \frac{[(VP + FN)(VP + FP)] + [(VN + FN)(VN + FP)]}{(VP + VN + FP + FN)^2}. \quad (17.6)$$

Sobre a avaliação dos classificadores, (LANDIS; KOCH, 1977b) partitionaram o nível de exatidão do Índice *Kappa* conforme a Tabela 114.

Tabela 114 – Nível de exatidão de uma classificação, conforme o valor de índice *Kappa*

Índice <i>Kappa</i> (K)	Qualidade
$K < 0,2$	Ruim
$0,2 \leq K < 0,4$	Razoável
$0,4 \leq K < 0,6$	Bom
$0,6 \leq K < 0,8$	Muito Boa
$\geq 0,8$	Excelente

5. Área Sob a Curva ROC (Area Under ROC)

É uma métrica para classificação binária, é provavelmente uma das mais populares. (BRAGA, 2001) afirma que a curva ROC é um método gráfico simples e robusto, permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. AUC é uma abreviação da área sob a curva. É usado na análise de classificação para determinar qual dos modelos utilizados prediz melhor as classes. Quanto mais próximo o AUC de um modelo chegar a 1, melhor ele será (GIANNINI, 2012). Assim, os modelos com AUCs mais altos são preferidos em relação àqueles com AUCs mais baixos.

Após apresentar fundamentações, métodos e técnicas ao longo desse trabalho, a seguir são descritos os resultados obtidos ao fim das etapas de treinamento e testes avaliativos, além disso será abordado uma discussão sobre o processo utilizado para resolver o problema proposto.

17.4 Resultados e Discussões

Este capítulo mostra e discute os resultados obtidos com a metodologia proposta por esse trabalho, obtendo assim, a classificação das regiões de interesse de imagens dos carcinomas mamários. Além disso, ele apresenta o comparativo dos descritores e arquiteturas de redes convolucionais.

17.4.1 Treinamento e Avaliação

Nesse trabalho desenvolvemos um método para classificação de imagem histopatológica de câncer de mama em duas classes (*in situ* e *invasiva*) que foram coletadas da base ICIAR 2018 usando abordagens de Aprendizagem Profunda e Transferência de Aprendizado. A segunda parte consistiu em extrair características dessas imagens utilizando as redes pré-treinadas com pesos da ImageNet, disponíveis na biblioteca Keras¹. Devido a Transferência de Aprendizado, as camadas de classificação foram removidas, com isso, cada arquitetura produziu a quantidade de características mostradas na Tabela 115.

Tabela 115 – Tabela descritiva da quantidade obtida de características por cada algoritmo utilizado nesse trabalho. Fonte: Elaborada pelo autor.

Algoritmos	Características
ResNet	(100352)
VGG16	(25088)
VGG19	(25088)
InceptionV3	(51200)
Xception	(100352)
LBP	(18)
HOG	(576)
Dense Daisy	(45864)

Na terceira etapa utilizamos os descritores clássicos para a extração de características da base, onde as imagens foram redimensionadas oferecendo tamanho de entrada padrão (64 x 64) para os mesmos, pois esse formato foi o que demonstrou melhor relevância nos resultados em relação as métricas avaliadas no Seção 17.3.5.

No processo final de classificação é utilizado os classificadores *Support Vector Machine* (SVM) e o *Random Forest*, para o SVM avaliamos diferentes kernels (Linear, Sigmoid, RBF, Polynomial) para oferecer diversidade de comparações preliminares. No classificador *Random Forest* o número de estimadores utilizado foi 200 (esse parâmetro é empregado para melhorar a precisão preditiva e controlar o ajuste excessivo de erros), onde foram testados vários valores dentro de um intervalo de 50 a 500 árvore. Primeiro utilizamos o método de divisão de base de dados onde ela foi dividida em 75% para treino e 25% para teste. Outro método de avaliação importante empregado nesse trabalho é o *cross validation*, onde ele consiste em dividir a base de imagens em k partes, usando k-1 partes para treino e a parte remanescente para teste, fazendo isso k vezes.

17.4.2 Testes Avaliativos

Para a realização dos experimentos aqui descritos, utilizou-se um computador, cujas especificações são descritas na Tabela 116.

Tabela 116 – Descrição do Ambiente de Desenvolvimento

Características	Descrição
Modelo	NP370E4K
RAM	4,00 GB
Processador	Intel(R) core(TM) i3
Sistema Operacional	Windows 10

¹ Disponível em <https://keras.io/applications/>

A seguir, os resultados obtidos com a metodologia proposta serão apresentados e discutidos. As Tabelas 117 e 118 mostram os resultados obtidos com as arquiteturas e descritores, respectivamente, mencionados no Capítulo 17.3 utilizando o método de divisão da base de dados (75% para treino e 25% para teste). Já as Tabelas 119 e 120 mostram os resultados obtidos utilizando o método de *cross validation*.

A Tabela 117 apresenta os resultados dos algoritmos com o classificador SVM em que o kernel utilizado é o Linear, pois ele foi o kernel com melhor eficiência dentre os outros utilizados. Nessa Tabela é possível notar que a ResNet50 obteve os melhores resultados entre os algoritmos descritos, com acurácia de 96%, sendo esse o maior resultado do projeto. Além disso, nota-se que os descritores não conseguiram discriminar as imagens das duas classes, muito provavelmente por que os parâmetros dos kernels não foram alterados nas trocas dos mesmos.

Tabela 117 – Tabela comparativa dos resultados (divisão da base) utilizando o classificador SVM com Kernel Linear

Arquiteturas				
Algoritmos	Accuracy	Sensitivity	Specificity	Kappa
ResNet50	0.96	0.95	0.96	0.91
InceptionV3	0.82	0.93	0.77	0.62
VGG16	0.84	0.72	0.92	0.66
VGG19	0.90	0.81	0.96	0.79
Xception	0.86	0.85	0.86	0.71
Descritores				
LBP	0.44	1.0	0.0	0.0
HOG	0.44	1.0	0.0	0.0
Dense Daisy	0.44	1.0	0.0	0.0

A Tabela 118 apresenta os resultados dos algoritmos com o classificador *Random Forest*. Nessa Tabela é possível notar que a VGG19 obteve os melhores resultados entre os algoritmos utilizados, com acurácia de 90%. Além disso, também é possível notar que os descritores conseguiram valores bem próximos das arquiteturas de redes profundas, principalmente o descritor LBP que chegou na faixa dos 80% de acurácia..

Tabela 118 – Tabela comparativa dos resultados (divisão da base) utilizando o classificador *Random Forest*

Arquiteturas				
Algoritmos	Accuracy	Sensitivity	Specificity	Kappa
VGG19	0.90	0.86	0.92	0.79
InceptionV3	0.86	0.89	0.83	0.71
VGG16	0.88	0.86	0.89	0.75
ResNet50	0.88	0.83	0.92	0.75
Xception	0.82	0.78	0.85	0.63
Descritores				
LBP	0.80	0.72	0.85	0.59
HOG	0.76	0.68	0.82	0.50
Dense Daisy	0.70	0.63	0.75	0.38

A Tabela 119 apresenta os resultados dos algoritmos no método *cross validation* com o classificador SVM em que o kernel utilizado é o Linear, pois ele novamente foi o kernel com melhor eficiência nos resultados obtidos. Nesse método é utilizado o K com valor 5 para divisão da base, por causa da baixa quantidade de imagens. Nessa Tabela é possível notar que a ResNet50 obteve os melhores resultados entre os algoritmos descritos.

A Tabela 120 apresenta os resultados dos algoritmos no método *cross validation* com o classificador *Random Forest*. Nessa Tabela a ResNet50 obteve o maior resultado dentre esse método de divisão de base, com acurácia de 91%, muito superior aos descritores que ficaram abaixo da faixa de 80% de acurácia.

Como pode ser observado, a ResNet50 obteve o melhor resultado em três das quatro tabelas de resultados apresentados. Um ponto de destaque nos mesmos é o valor do *Kappa*, pois com base na Tabela 114 o

Tabela 119 – Tabela comparativa dos resultados (*cross validation*) utilizando o classificador SVM com Kernel Linear

Arquiteturas				
Algoritmos	Acuracy	ROC	Precision	Recall
ResNet50	0.89	0.96	0.88	0.92
InceptionV3	0.87	0.93	0.85	0.88
VGG16	0.83	0.90	0.80	0.89
VGG19	0.86	0.92	0.83	0.90
Xception	0.77	0.88	0.79	0.76
Descretores				
LBP	0.58	0.74	0.65	0.64
HOG	0.63	0.70	0.59	0.94
Dense Daisy	0.61	0.80	0.57	0.97

Tabela 120 – Tabela comparativa dos resultados (*cross validation*) utilizando o classificador *Random Forest*

Arquiteturas				
Algoritmos	Acuracy	ROC	Precision	Recall
ResNet50	0.91	0.94	0.93	0.87
InceptionV3	0.86	0.94	0.87	0.84
VGG16	0.84	0.93	0.88	0.86
VGG19	0.86	0.93	0.88	0.86
Xception	0.76	0.85	0.74	0.76
Descretores				
LBP	0.68	0.80	0.71	0.66
HOG	0.77	0.84	0.78	0.78
Dense Daisy	0.72	0.78	0.69	0.73

algoritmo utilizado tem um excelente nível de exatidão na classificação. Os demais resultados obtidos foram satisfatórios de acordo com as métricas especificadas na Seção 17.3.5, indicando uma excelente capacidade discriminativa e classificatória do algoritmo para as classes definidas.

A Tabela 121 apresenta uma breve comparação entre os resultados encontrados utilizando a metodologia proposta e alguns trabalhos citados no Capítulo 17.2, que realizam a classificação de mama, apenas em duas classes. Porém, alguns dos trabalhos utilizaram metodologias e bases diferentes da usada neste trabalho. Portanto, os dados apresentados na Tabela 121, são apenas para ilustrar/comparar os resultados obtidos com os trabalhos da área, sem desmerecer nenhum dos métodos citados.

Tabela 121 – Tabela comparativa dos resultados dos trabalhos da área com esse trabalho.

Trabalhos	A (%)	S (%)	E (%)	AUC
(GOLATKAR; ANAND; SETHI, 2018)	93	-	-	-
(RAKHLIN et al., 2018)	93.8	-	-	-
(ARAÚJO et al., 2017)	83.3	-	-	-
Método Proposto	96	96	95.5	96.4

17.5 Conclusão

Nesse trabalho avaliamos o desempenho de algoritmos selecionados para classificação dos carcinomas em imagens de biopsia da mama. Os resultados obtidos na abordagem Transferência de Aprendizado proposta foram eficientes e eficazes para a classificação automática da histologia do câncer de mama em imagens.

As arquiteturas de redes investigadas transferiram com sucesso o conhecimento do ImageNet codificados como características convolucionais para o problema da histologia na presença de dados limitados de

treinamento. A rede residual (ResNet50) investigada, superou as demais arquiteturas profundas e descritores clássicos, em função das métricas utilizadas para avaliação.

O trabalho apresentado confirma a poderosa capacidade de classificação das CNNs conjuntamente com a Transferência de Aprendizado evidenciando uma eficiente ferramenta para a análise automática de imagens histológicas, mesmo com um número limitado de imagens e recursos computacionais escassos.

Assim, acredita-se que a metodologia apresentada neste trabalho poderá integrar uma ferramenta CAD, que é uma ferramenta que auxilia os médicos em diagnósticos por meio de imagens, sendo ela aplicada em casos reais e atuais na classificação dos carcinomas mamários. Assim, trazendo benefícios ao especialista, que poderá contar com uma segunda opinião durante o processo de diagnóstico do potencial cancerígeno do câncer, bem como garantindo ao paciente melhor qualidade de vida e um prognóstico mais favorável quanto a sua saúde.

Como trabalhos futuros, pretende-se: aumentar a quantidade e variabilidade das amostras histológicas mamárias utilizando mais bases de imagens e também técnicas de aumento de dados, testar outras técnicas para melhorar o desempenho das CNNs, tais com o *fine-tuning*.

Uso de Transferência de Aprendizado para Detecção Automática de Melanoma

Autores: Aroldo de Moura Santos Júnior; Lucas Marques de Sousa Silva; Flávio Henrique Duarte de Araújo

Resumo: Atualmente, o câncer de pele é visto como uma das formas mais perigosas dos cânceres encontrados em seres humanos. Ele pode ser classificado em vários tipos, como melanoma, carcinoma de células basais ou escamosas e outros, onde o primeiro é o mais imprevisível. Quanto mais precoce for o diagnóstico do melanoma maior as chances de cura, por isso foram desenvolvidas técnicas de computação baseadas em aprendizagem profunda que podem desempenhar um papel importante no diagnóstico de imagens médicas. Porém, elas necessitam de uma grande quantidade de imagens o que acaba se tornando um desafio. Neste trabalho utilizamos a transferência de aprendizado, pois não foi possível treinar uma arquitetura própria devido a pequena quantidade de imagens da base de dados coletadas do arquivo ISIC e limitações de poder computacional de hardware. Foram testadas diferentes arquiteturas de Redes Neurais Convolucionais (CNN) utilizando a transferência de aprendizado e também comparamos seus desempenhos com outros classificados utilizando descritores clássicos da literatura como o HOG, LBP e o Dayse. O modelo que obteve o melhor resultado foi a ResNet50 que atingiu uma acurácia de 88,5%, sensibilidade de 89,2%, especificidade de 87,7% e uma área sob a curva ROC de 88 ultrapassando os descritores clássicos utilizados nesse trabalho.

Keywords: Processamento de Imagens, Aprendizagem Profunda, Redes Neurais Convolucionais.;

18.1 Introdução

O melanoma e o câncer de pele não-melanoma (carcinoma basocelular e de células escamosas) são exemplos de alguns dos tipos de cânceres de pele diagnosticados atualmente. Embora o segundo tipo tenha uma maior incidência, o primeiro tipo possui uma taxa de fatalidade bem maior. De acordo com Tuccitto et al. (2019) o melanoma representa apenas 4% dos cânceres de pele, mas é responsável por 80% das mortes relacionadas a lesões de pele.

O termo melanoma se aplica a um tipo de câncer de pele grave que pode ocorrer em qualquer parte do corpo que possua melanócitos. A principal causa do melanoma é a excessiva exposição à radiação ultravioleta

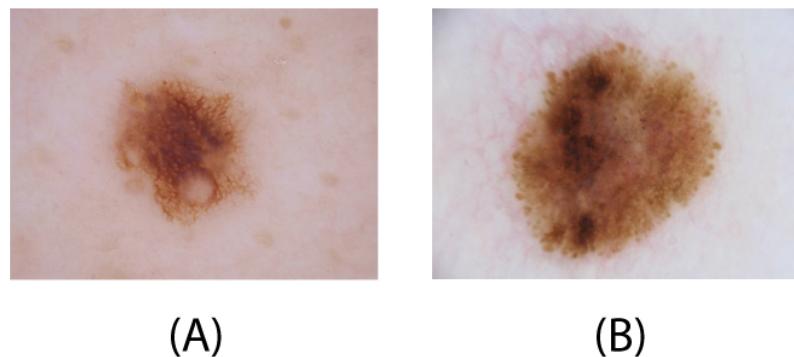
(UV), que pode acabar prejudicando as células da pele afetando o seu crescimento. As pessoas mais afetadas por esse tipo de câncer são as de pele clara, devido ao baixo teor de melanina.

Estima-se que, em anos anteriores, 105.000 pessoas foram diagnosticadas com melanoma e que 33.000 morreram da doença em todo o mundo (SHEHA; MABROUK; SHARAWY, 2012). De acordo com o Instituto Nacional de Câncer (INCA)¹, no ano de 2018, estima-se que foram diagnosticados 6.260 novos casos de câncer de pele melanoma no Brasil.

Segundo Ercal et al. (1994), o diagnóstico de melanoma maligno é um verdadeiro desafio, pois possui características bem semelhantes a outros tipos de lesões na pele. Em muitos casos, os dermatologistas devem realizar uma biópsia (um procedimento médico de laboratório) para analisar de forma detalhada se um tumor é maligno ou benigno. Como este é um procedimento que envolve um certo nível de despesa e morbidade, particularmente em pacientes com múltiplos sinais atípicos, técnicas alternativas de detecção precoce estão sendo procuradas para uma rápida e conveniente detecção do câncer de pele.

Os médicos geralmente confiam na sua experiência profissional e avaliam as lesões de cada paciente, caso a caso, levando em conta os padrões de lesão local do paciente em comparação com as demais existentes no corpo. No entanto, as diferenças visuais entre melanoma e lesões cutâneas benignas podem ser muito sutis, como podemos observar na Figura 1, tornando difícil distinguir os dois casos, mesmo para médicos especialistas (LOPEZ et al., 2017). A Figura 35.5 mostra dois exemplos de imagens dermatoscópicas da base de dados da Colaboração Internacional de Imagem de Pele (*International Skin Imaging Collaboration – ISIC*)²: uma de melanoma (A) e outra de nevos (B), onde pode-se perceber que ambos possuem características semelhantes, como a coloração e a textura.

Figura 130 – Melanoma (A) e nevos (B). Fonte: ISIC.



Os sistemas de detecção auxiliada por computador (*Computer-Aided Design - CAD*) foram desenvolvidos para reduzir os gastos e melhorar a capacidade de inúmeros profissionais que lidam com imagens. Estudos recentes, como por exemplo Elsheikh, Mustafa e Ibraheem (2018), comprovaram que os sistemas CAD são capazes de revelar lesões de pele, como o melanoma. Embora não sejam utilizados para emitir o diagnóstico final, eles servem como uma segunda opinião para os especialistas.

Com base no exposto, o objetivo geral desse trabalho é desenvolver um método computacional para a detecção automática de melanoma em imagens médicas de pele. Como objetivos específicos temos:

1. Analisar a performance de diferentes arquiteturas de redes e também descritores clássicos aliado a transferência de aprendizado para diagnosticar casos de melanoma.

¹ Disponível em ;(<https://www.inca.gov.br/tipos-de-cancer/cancer-de-pele-melanoma>) ; acesso em 29/04/19

² O ISIC é uma parceria acadêmica e industrial projetada para facilitar a aplicação de imagens digitais da pele para ajudar a reduzir a mortalidade por melanoma. Disponível em ;(<https://www.isic-archive.com/>) ; acesso em 30/04/19

2. Comparar o desempenho do método proposto com outros trabalhos de detecção de melanoma presentes na literatura.

As próximas seções estão dispostas da seguinte forma: a seção 18.2 contém, de forma breve, descrições dos principais trabalhos relacionados encontrados na literatura. Já na seção 18.3 são apresentados conceitos importantes ao desenvolvimento e compreensão do trabalho bem como a base de dados e técnicas utilizadas. Logo em seguida, a seção 18.4 aborda as principais características deste trabalho, tais como ambiente e método de desenvolvimento, resultados alcançados e outras informações úteis a respeito da metodologia aqui implementada. Por fim, na seção 18.5 são apresentadas as conclusões deste trabalho, bem como possíveis trabalhos futuros.

18.2 Trabalhos Relacionados

No que concerne aos trabalhos desta linha de pesquisa, pode-se observar uma tendência de crescimento no uso de técnicas de aprendizado profundo. A Tabela 122 mostra as principais informações dos trabalhos relacionados e em seguida há uma breve descrição de cada um deles.

Tabela 122 – Tabela Comparativa de Trabalhos Relacionados.

Referências	Base de Imagens	Arquiteturas ou Descritores	Resultados
Mahbod et al. (2019)	ISIC 2016 ISIC 2017 Total: 2.037 imagens	AlexNet VGG16 ResNet	AUC de 83,83%
Menegola et al. (2017)	Atlas ISIC 2016 Total: 900 imagens	VGG16	ACC de 83,5%
Chang (2017)	ISIC Total: 2000 imagens	InceptionV3	ACC de 80%
Yu et al. (2017)	ISIC 2016 Total: 900 imagens	VGG16 GoogleNet FCRN	ACC de 94%
Lopez et al. (2017)	ISIC <i>Dermofit Image Library</i> Dermnet Total: 26.000 imagens	VGG	SEN de 79%
Esteva et al. (2017)	ISIC EDL <i>Stanford Hospital</i> Total: 900 imagens	InceptionV3	ACC de 72%.
Majtner, Yildirim-Yayilgan e Hardeberg (2016)	ISIC 2016 Total: 900 imagens	RSurf LBP	ACC de 82%.

[Mahbod et al. \(2019\)](#) abordaram um método computacional totalmente automático para classificação de lesões de pele, que emprega características profundas otimizadas. Eles utilizaram três modelos profundos pré-treinados, o AlexNet, o VGG16 e o ResNet, como geradores de características profundas que são usadas para treinar o classificador *Support Vector Machine (SVM)*. Eles usaram as imagens do desafio ISIC 2016 e ISIC 2017, totalizando 2037 imagens dermatoscópicas, incluindo 411 melanomas malignos, 254 ceratose seborreica e 1372 nevos benignos. As imagens são de vários tamanhos, ângulos fotográficos e condições de iluminação. O resultado obtido foi de uma área sob a curva ROC de 83,83% para classificação de melanoma.

[Menegola et al. \(2017\)](#) investigam o uso de transferência de aprendizado em redes de aprendizagem profunda. Os conjuntos de dados empregados nesse trabalho foram o *Atlas of Dermoscopy (Atlas)* e o *ISBI Challenge 2016 / ISIC* que contém 900 imagens para treinamento (sendo 273 melanomas) e 379 para testes

(sendo 115 melanomas). Os autores aplicaram a transferência de aprendizado na arquitetura VGG16 com *fine-tuning* pré-treinada com pesos da ImageNet e também o SVM para classificar melanoma e lesões benignas, alcançando uma acurácia de 83.5%.

Chang (2017) relata uma técnica de aprendizagem profunda utilizando a transferência de aprendizado. Como base de dados foram utilizadas 2000 imagens dermatoscópicas coletadas no arquivo da ISIC sendo 374 imagens de melanoma, 1372 imagens de nevos e 254 imagens de verruga seborreica. Ele aplicou a transferência de aprendizado em uma rede neural Inceptionv3 atingindo uma acurácia de 80%.

Yu et al. (2017) propuseram um novo método para o reconhecimento do melanoma, com redes neurais convolucionais profundas. Os autores utilizaram o conjunto de dados do desafio ISIC 2016 que possui imagens da base de dados da ISIC e é dividido em 900 imagens para treino e 350 imagens para teste. Eles aplicaram diferentes arquiteturas com diferentes profundidades, dentre elas a VGG 16, a GoogleNet com 22 camadas de profundidade e *Fully Convolutional Residual Network - FCRN*. A que atingiu os melhores resultados foi a FCRN de 50 camadas de profundidade obtendo uma acurácia de 94%, uma sensibilidade de 91% e uma especificidade de 95%.

Lopez et al. (2017) propuseram uma abordagem baseada em transferência de aprendizado para treinamento com a arquitetura VGG usando pesos pré-treinados da ImageNet. As imagens utilizadas nesse artigo são do conjunto de imagens do desafio ISIC 2016, da *Dermofit Image Library* e Dermnet, onde os autores obtiveram um valor de sensibilidade de 78% e uma precisão de 79% para classificação de lesões benignas ou malignas.

Esteva et al. (2017) propuseram um método para classificação de lesões cutâneas usando uma única CNN (arquitetura InceptionV3) treinada no banco de dados ImageNet, removendo a camada de classificação da rede e atualizando com sua própria base de dados. Nesse trabalho foi utilizado um conjunto de dados que vem de uma combinação de repositórios de dermatologia de acesso aberto, do ISIC, da *Edinburgh Dermofit Library - EDL* e dos dados do *Stanford Hospital*. O método proposto alcançou acurácia de 72%.

Majtner, Yildirim-Yayilan e Hardeberg (2016) abordaram um sistema automatizado de reconhecimento de melanoma. Os autores coletaram as imagens na base de dados da ISIC separando 900 imagens (727 imagens de lesões benignas e 173 imagens de melanoma) para treinamento e 379 imagens (304 imagens de lesões benignas e 75 imagens de melanoma) para teste. Todo o conjunto de dados consiste em imagens de várias fontes, simulando o uso real das soluções propostas. Eles adotam um método de aprendizagem profunda combinado com os descritores RSurf e *Local Binary Patterns - LBP* onde conseguiram atingir uma acurácia de 82%, uma sensibilidade de 53%, uma especificidade de 89% e uma área sobre curva de 78%.

A análise dos trabalhos encontrados contribuiu para identificar que as técnicas de aprendizagem profunda estão sendo bastante utilizadas em aplicações de imagens médicas. Também observou-se o uso de hardwares com grande capacidade de processamento para atingir os objetivos propostos por cada trabalho, por isso resolvemos investigar o uso dessas técnicas com um hardware de baixo poder de processamento. Para contornar o problema da baixa quantidade de imagens e da limitação do hardware utilizado neste trabalho, resolvemos usar a transferência de aprendizado que será abordada mais a frente. Também decidimos testar outros descritores clássicos para confirmar se as CNNs realmente apresentavam um melhor desempenho com uma quantidade de imagens reduzidas. Com as pesquisas realizadas, também vimos que seria necessário utilizar classificadores externos para a validação, por isso utilizamos dois: o SVM e o RandomForest. Em seguida, veio a etapa de desenvolvimento e testes computacionais.

18.3 Materiais e Métodos

Nesta seção são apresentados a base de imagens e as principais arquiteturas de CNNs e descritores clássicos comparados. Por fim são apresentadas as métricas de avaliação dos resultados obtidos.

18.3.1 Base de Dados

A base de imagens utilizada nesse trabalho foi coletada da International Skin Imaging Collaboration - ISIC) que contém atualmente uma grande coleção de imagens dermatoscópicas de livre acesso ao público, que foram coletadas dos principais centros clínicos do mundo todo e adquiridas a partir de uma variedade de

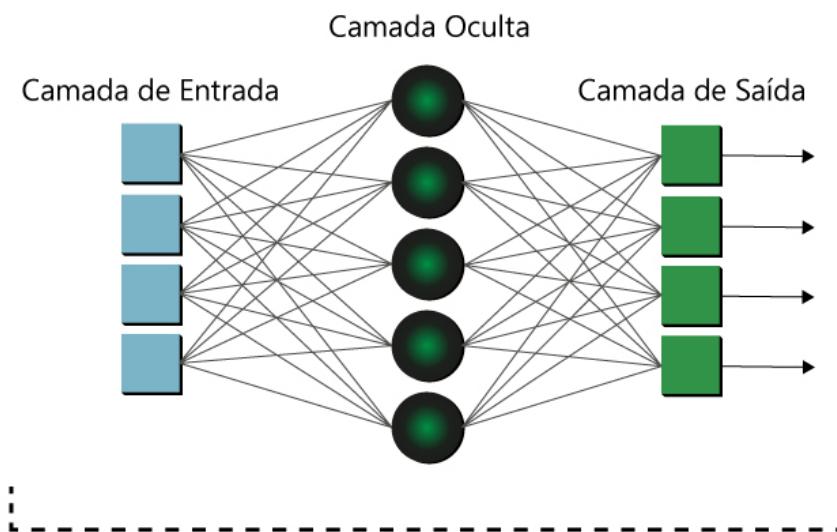
dispositivos dentro de cada centro. Todas as imagens recebidas no arquivo da ISIC são rastreadas quanto à privacidade e garantia de qualidade. A maioria das imagens tem metadados clínicos associados, avaliados por especialistas reconhecidos em melanoma.

Foram escolhidas duas categorias (classes) para compor a base de imagens que, de acordo com [Codella et al. \(2015\)](#), são as mais difíceis de serem distinguidas pelos especialistas. Ela é composta por 800 imagens sendo, 400 de melanoma e 400 de nevos atípicos, que não são melanoma mas, possuem características bem semelhantes como a cor, o formato e tamanho. Elas possuem dimensões variadas que vão desde 722x542, até 6708x4419 pixels e foram escolhidas individualmente, descartando aquelas que possuíam a presença de artefatos, naturais (cabelos, veias) ou artificiais (bolhas de ar, marcas de régua, gráficos de calibração de cores) pois, segundo [Yu et al. \(2017\)](#), tais artefatos podem embaçar ou obstruir as lesões da pele dificultando ainda mais o processamento dessas imagens. A seguir serão apresentados os conceitos das técnicas utilizadas nesse trabalho, começando com Redes Neurais Artificiais que é a base dos estudos desta área.

18.3.2 Artificial Neural Network

As Redes Neurais artificiais (*Artificial Neural Network - ANN*) são redes computacionais que tentam, a grosso modo, simular o processo de decisão de uma rede de célula nervosa do sistema nervoso central biológico humano ou animal ([DANIEL, 2013](#)). São capazes de aprender à partir de um conjunto de dados, trabalhando de forma especializada com reconhecimento de padrões. Elas, semelhante ao sistema nervoso, possuem componentes lógicos (Neurônios Artificiais), organizados em um espécie de rede, que recebem entradas (características) e passam por uma camada oculta que decide quais as saídas mais adequadas a cada uma dessas entradas. A Figura 131 traz a ilustração de uma ANN.

Figura 131 – Ilustração de uma ANN. Fonte: Adaptação de [LeCun et al. \(1998\)](#).



18.3.3 Deep Learning

A aprendizagem profunda (*Deep Learning* em inglês) é um subcampo do aprendizado de máquina que tem como foco principal desenvolver meios para resolver problemas de forma semelhante ao processo de tomada de decisão humano. O termo profunda não se refere a qualquer tipo de compreensão alcançada pelo método, ele se refere à ideia de múltiplas camadas de representação sucessivas ([CHOLLET, 2018](#)).

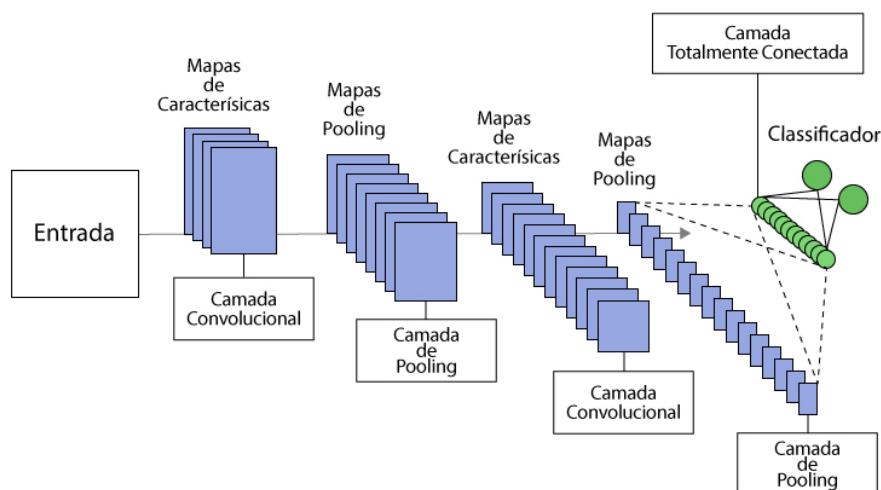
A profundidade do modelo é determinada pela quantidade de camadas ocultas. Os modelos de aprendizagem profunda atuais podem conter dezenas ou até centenas de camadas ocultas e todas são aprendidas automaticamente necessitando apenas da inserção dos dados de treinamento.

Ao passar dos anos surgiu uma categoria de redes neurais artificiais baseada em aprendizagem profunda que inovaram por utilizar convolução em algumas de suas camadas. Essa categoria que passou a ser o novo padrão de visão computacional, é chamada de rede neural convolucional que será descrita a seguir.

18.3.4 Convolutional Neural Network

As redes neurais convolucionais (CNNs) se tornaram o novo padrão em visão computacional e são fáceis de treinar quando existe grande quantidade de amostras rotuladas que representam as diferentes classes-alvo. Algumas das vantagens consistem em: (a) capacidade de extrair características relevantes através de aprendizado de transformações (kernels) e (b) depender de menor número de parâmetros de ajustes do que redes totalmente conectadas com o mesmo número de camadas ocultas. Como cada unidade de uma camada não é conectada com todas as unidades da camada seguinte, há menos pesos para serem atualizados, facilitando assim o treinamento (ARAÚJO et al., 2017a). A Figura 132 nos mostra uma ilustração de uma CNN com duas camadas convolucionais, duas de *pooling* e uma totalmente conectada.

Figura 132 – Ilustração de uma CNN formada por camadas convolucionais, de *pooling* e uma camada totalmente conectada. Fonte: Adaptação de Saha (2018).



Como observado na Figura 132 a rede neural convolucional possui três principais camadas: convolucionais, de *pooling* e totalmente conectadas. De acordo com Araújo et al. (2017a) as camadas convolucionais são responsáveis por extrair atributos dos volumes de entradas. As camadas de *pooling* são responsáveis por reduzir a dimensionalidade do volume resultante após as camadas convolucionais e ajudam a tornar a representação invariante a pequenas translações na entrada. As camadas totalmente conectadas são responsáveis pela propagação do sinal por meio da multiplicação ponto a ponto e o uso de uma função de ativação.

Quanto maior a quantidade de amostras (base de dados) que representam as diferentes classes, mais fácil será o treinamento de uma CNN. Mas, nem sempre é possível utilizar grandes quantidades de imagens devido a empecilhos como barreiras de acesso a essas bases ou até mesmo limitação do hardware, o que atrapalha o uso dessa ferramenta. Para contornar esse problema, foi desenvolvido uma técnica chamada de transferência de aprendizado, que será descrita a seguir.

18.3.5 Transferência de Aprendizado

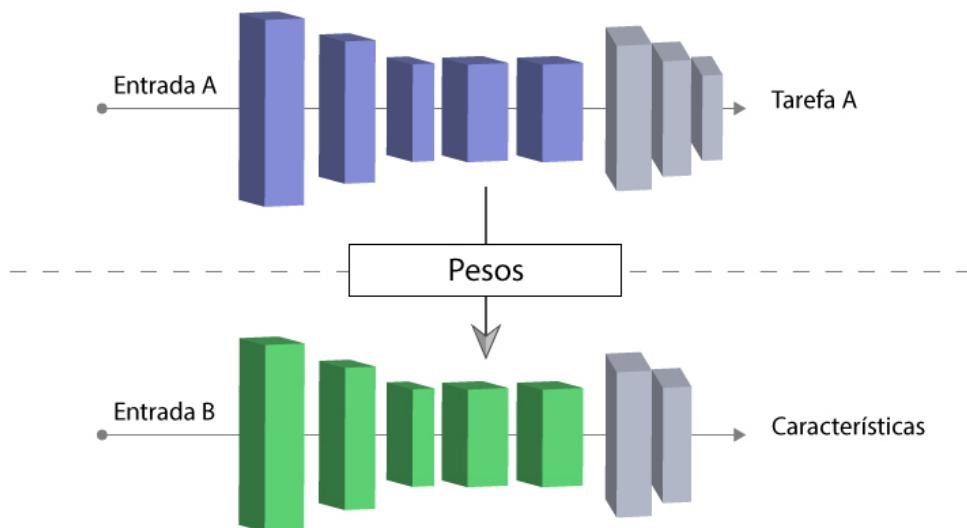
O termo transferência de aprendizado se refere ao processo de transferir o conhecimento da tarefa de origem para uma tarefa de destino permitindo um progresso rápido ou duradouro ao modelar a segunda tarefa. Na área de aprendizagem de máquina essa técnica é utilizada quando não é possível obter uma base de dados grande o suficiente para resolver uma tarefa. Transferência de aprendizado e adaptação de domínio referem-se à situação em que o que foi aprendido em um ambiente... é explorado para melhorar a generalização em outro ambiente (GOODFELLOW; BENGIO; COURVILLE, 2016).

Existem três medidas comuns pelas quais a transferência pode melhorar o aprendizado. O primeiro é o desempenho inicial alcançável na tarefa alvo usando apenas o conhecimento transferido, antes que qualquer outra aprendizagem seja feita, comparada ao desempenho inicial de um agente ignorante. A segunda é a quantidade de tempo que leva para aprender completamente a tarefa-alvo, dado o conhecimento transferido em comparação com a quantidade de tempo para aprendê-la do zero. O terceiro é o nível de desempenho final alcançável na tarefa alvo, comparado ao nível final sem transferência (TORREY; SHAVLIK, 2010).

Segundo Karpathy et al. (2016), basicamente há três formas de se utilizar a transferência de aprendizado com CNNs. A primeira consiste em usar a CNN como extrator de características. Este é o caso das CNNs pré-treinadas da ImageNet, que remove a última camada totalmente conectada, transformando a CNN em extrator de características. A segunda forma é o *fine-tuning*, onde os pesos da rede pré-treinada são ajustados continuando a retropropagação. E a terceira forma são os modelos pré-treinados. Como as CNNs modernas levam de 2 a 3 semanas para treinar em várias GPUs no ImageNet, é comum ver pessoas liberando seus últimos pontos de verificação da CNN para o benefício de outras pessoas que podem usar as redes para o *fine-tuning*.

A Figura 133 nos mostra um esquema de transferência de aprendizado aplicada neste trabalho onde utilizamos os pesos de CNNs treinadas para a ImageNet e removemos a camada de classificação, assim o resultado da rede são as características que descrevem a imagem de entrada. Depois disso utilizamos um outro classificador para a classificação.

Figura 133 – Exemplo da transferência de aprendizado utilizada neste trabalho. Fonte: Elaborada pelo autor.



18.3.6 Modelo das Arquiteturas Comparadas

Após analisar a bibliografia que resultou da pesquisa a respeito do tema, foram encontrados diversos modelos de arquitetura compatíveis com a proposta desse trabalho. Foram realizadas comparações e avaliações para determinar qual a melhor performance dentro do objetivo do trabalho. Com isso, foi possível selecionar

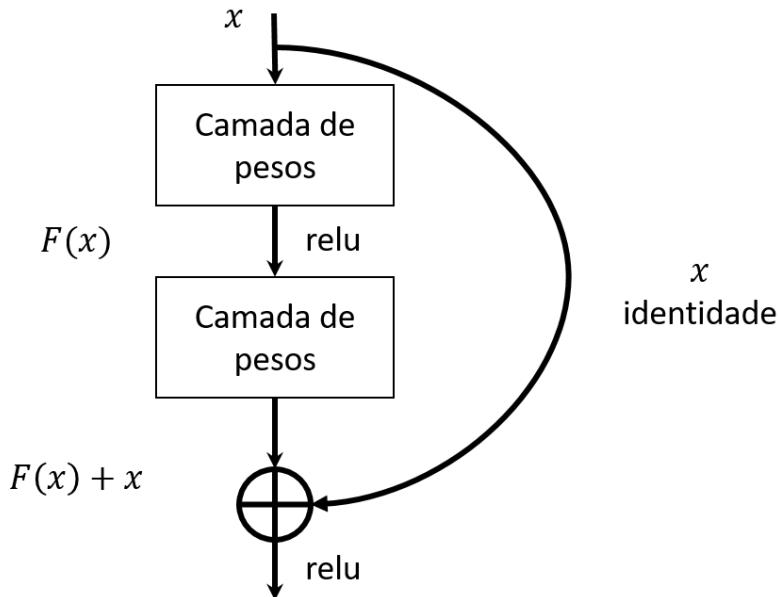
nove principais modelos para estudo mais profundo e consequente avaliação da melhor performance. Tais modelos são descritos à seguir.

18.3.6.1 ResNet

A ResNet é uma arquitetura proposta por [He et al. \(2016\)](#) que tem como principal característica suas conexões residuais. Na aprendizagem residual, o foco não é aprender características, mas sim aprender alguns resíduos. Resíduo pode ser entendido como a subtração da característica aprendida na camada de entrada.

Um dos problemas que as ResNets resolvem é o famoso desaparecimento de gradiente. Isso ocorre porque, quando a rede é muito profunda, os gradientes de onde a função de perda é calculada diminuem facilmente para zero após várias aplicações da regra de cadeia. Com tal arquitetura, os gradientes podem fluir diretamente através das conexões de salto. A Resnet fornece uma estrutura de treinamento para facilitar formação de redes substancialmente mais profundas. A Figura 134 mostra uma ilustração de um bloco residual, característico da ResNet.

Figura 134 – Ilustração de um bloco residual na arquitetura da rede ResNet. Fonte: Adaptação de [He et al. \(2016\)](#)



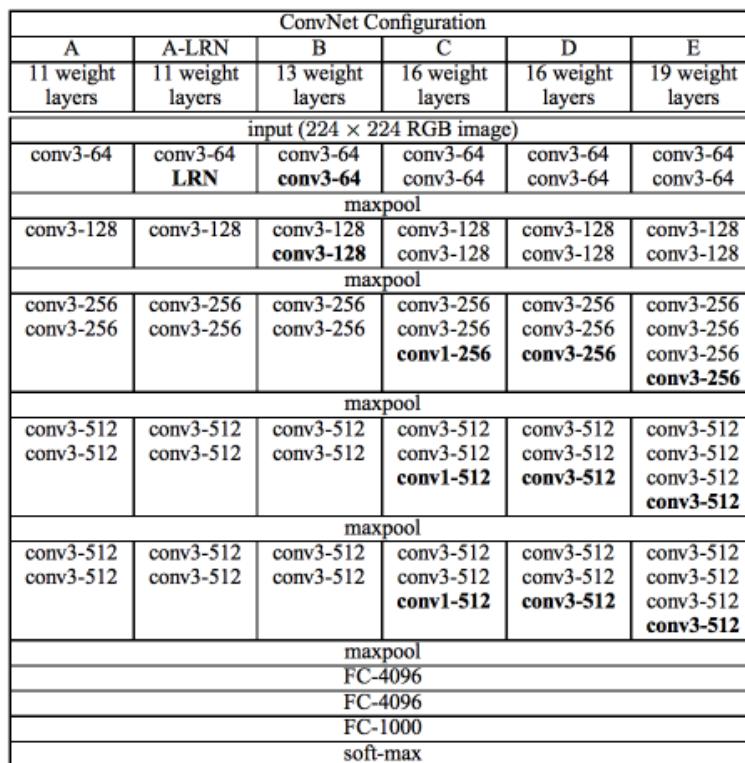
18.3.6.2 VGG

A rede VGG proposta por [Simonyan e Zisserman \(2014a\)](#) foi a primeira a utilizar filtros pequenos (3x3) em cada camada convolucionial. Ele faz a melhoria sobre o AlexNet, construída por [Krizhevsky, Sutskever e Hinton \(2012\)](#), onde filtros grandes (9x9 e 11x11) eram usados para capturar características similares na imagem. A grande contribuição da VGG foi a ideia de que múltiplas convoluções 3x3 em sequência podiam substituir efeitos de filtros de máscaras maiores (5x5 e 7x7), e que resultavam em maior custo computacional ([ARAÚJO et al., 2017a](#)). A proposta original da rede VGG era muito mais profunda que a do AlexNet. Nela as profundidades mais comuns são as de 16 e 19 camadas. A Figura 135 nos mostra uma ilustração da arquitetura utilizada na VGG.

18.3.6.3 InceptionV3

Baseado no artigo de [Szegedy et al. \(2016\)](#), o InceptionV3 é uma versão melhorada das anteriores (GoogleNet e InceptionV2), que teve como uma de suas principais mudanças a alteração da camada de convolução tradicional de 7x7 para 3 camadas de 3x3. Na sua parte *inception* ela possui 3 módulos de criação 35 x 35

Figura 135 – Ilustração da arquitetura da VGG. Fonte: [Simonyan e Zisserman \(2014a\)](#).



com 288 filtros cada (A). Isso é reduzido para 17 x 17 com 768 filtros (B) e logo em seguida há 5 instâncias de módulos fatoriais de criação onde ocorre uma nova redução para 8 x 8 x 1280 (C). Ao todo ela possui 42 camadas. A Figura 136 nos mostra uma ilustração da arquitetura InceptionV3.

18.3.6.4 Inception-ResNet

É uma combinação proposta por [Szegedy et al. \(2017\)](#) que une as conexões residuais (ResNet) e a arquitetura Inception. De acordo com [He et al. \(2016\)](#), as conexões residuais são essencialmente importantes para o treinamento de arquiteturas muito profundas. Como as redes do Inception tendem a ser muito profundas, foi proposto a substituição do estágio da concatenação do filtro da arquitetura Inception por conexões residuais. Isso permitiria o Inception colher os benefícios da abordagem residual, mantendo sua eficiência computacional. A Figura 137 mostra um exemplo de uma camada InceptionResnetV2.

18.3.6.5 Xception

Também conhecida como *Extreme Inception*, é uma variação mais forte da *Inception* que possui como uma de suas características principais a substituição dos módulos de iniciação padrões por convoluções separadas em profundidade. A arquitetura Xception possui 36 camadas convolucionais formando a base de extração de características da rede ([CHOLLET, 2017](#)). A Figura 138 nos mostra uma comparação de um módulo da InceptionV3 e um da Xception.

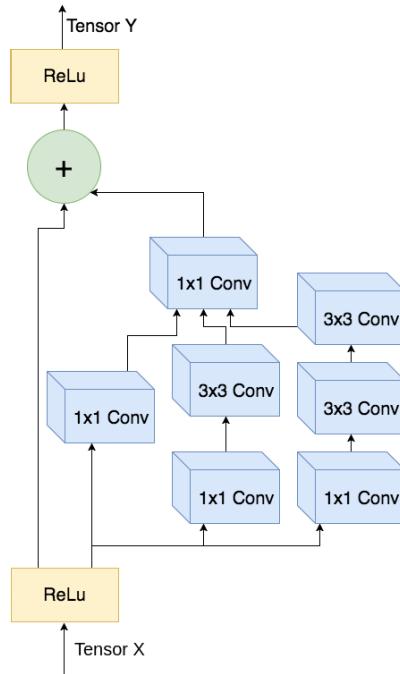
18.3.7 Descritores Clássicos Comparados

Com a pesquisa realizada também foi possível selecionar 3 descritores clássicos para comparar sua performance com as arquiteturas citadas acima. Esses descritores trabalham com imagens não segmentadas, por isso foi decidido utilizá-los, pois as imagens não foram segmentadas.

Figura 136 – Ilustração de uma arquitetura InceptionV3. Fonte: Adaptação de Szegedy et al. (2016).

Tipo	Tamanho do Fragmento	Tamanho da Entrada
conv	$3 \times 3 / 2$	$299 \times 299 \times 3$
conv	$3 \times 3 / 1$	$149 \times 149 \times 32$
conv padded	$3 \times 3 / 1$	$147 \times 147 \times 32$
pool	$3 \times 3 / 2$	$147 \times 147 \times 64$
conv	$3 \times 3 / 1$	$73 \times 73 \times 64$
conv	$3 \times 3 / 2$	$71 \times 71 \times 80$
conv	$3 \times 3 / 1$	$35 \times 35 \times 192$
$3 \times$ Inception	(A)	$35 \times 35 \times 288$
$5 \times$ Inception	(B)	$17 \times 17 \times 768$
$2 \times$ Inception	(C)	$8 \times 8 \times 1280$
pool	8×8	$8 \times 8 \times 2048$
linear	logits	$1 \times 1 \times 2048$
softmax	classifier	$1 \times 1 \times 1000$

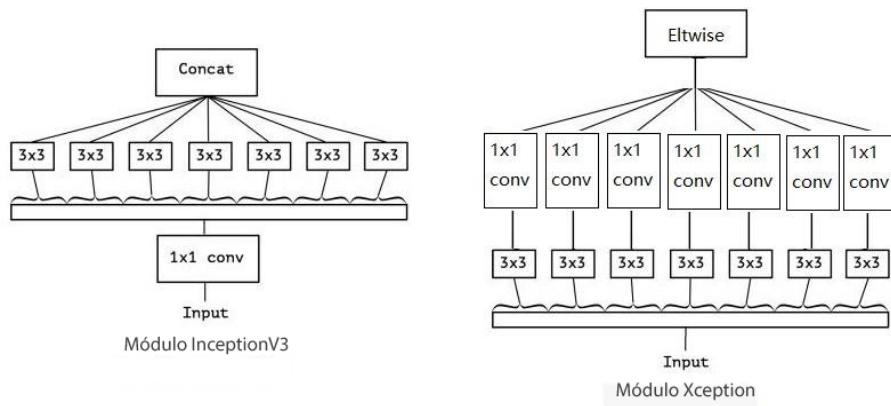
Figura 137 – Ilustração de uma camada InceptionResNetV2. Fonte: Nguyen-Meidine et al. (2017).



18.3.7.1 Histogram of Oriented Gradients - HOG

Ao utilizar esse descritor é necessário realizar a divisão de uma determinada imagem em pequenas regiões conectadas, chamadas de células, e, em seguida, para cada uma dessas partes, o cálculo de um histograma de gradientes orientados para os pixels. É possível então acumular esses histogramas através de múltiplas células para obter como resultado o vetor de características.

Figura 138 – Arquiteturas InceptionV3 (esquerda) e Xception (direita). Fonte: Adaptação de [Programmer-Sought \(2017\)](#).



No descriptor de característica HOG, a distribuição (histogramas) de direções de gradientes (gradientes orientados) é usada como características. Os gradientes (derivadas x e y) de uma imagem são úteis devido ao fato da magnitude dos gradientes ser grande em torno das bordas e cantos (regiões de mudanças bruscas de intensidade) e sabemos que bordas e cantos contêm muito mais informações sobre a forma do objeto do que as regiões planas.

O descriptor HOG consiste então em um vetor concatenado dos componentes dos histogramas de células normalizadas de todas as regiões do bloco. Esses blocos normalmente se sobrepõem, o que significa que cada célula contribui mais de uma vez para o descriptor final. Existem duas geometrias de blocos principais: blocos R-HOG retangulares e blocos C-HOG circulares.

18.3.7.2 Local Binary Patterns - LBP

O LBP é um tipo de descriptor visual usado para classificação em visão computacional. O LBP calcula uma representação local da textura e essa representação local é construída comparando cada pixel com sua vizinhança de pixels. [Topi et al. \(2000\)](#) descreve que o LBP contém a densidade de cada recurso em uma região e que, para cada pixel, apenas a saída de melhor correspondência de detector de recurso é utilizada.

O uso de uma vizinhança circular e a interpolação bilinear de valores em coordenadas de pixel não inteiro permitem qualquer raio e número de pixels na vizinhança. Outra extensão importante é a definição dos chamados padrões uniformes, que pode ser usado para reduzir o comprimento do vetor de recursos e implementar um descriptor simples de rotação invariante. Um padrão binário local é chamado uniforme se o padrão binário contiver no máximo duas transições bit a bit de 0 a 1 ou vice-versa

18.3.7.3 Dense Daisy

O Daisy é um descriptor de imagem baseado em histogramas de orientação de gradiente semelhantes ao descriptor SIFT (*Scale Invariant Feature Transform*). Ele é formulado de uma maneira que permite a extração rápida e densa.

[Zhu, Bichot e Chen \(2011\)](#) afirma que o Daisy usa uma configuração circular de vizinhança em vez da uma retangular, além do mais a parte do descriptor é feito pela concatenação de todos os vetores dos círculos, depois que eles são normalizados para a norma da unidade. O DAISY usa o histograma de gradiente filtrado pelo filtro de Gauss como uma descrição de recurso, para que o desfoque da imagem tenha um efeito mais proeminente ([LI; ZIC, 2014](#)). Além disso, eles são invariantes a rotação sem depender de uma orientação de referência, melhorando ainda mais sua robustez.

18.3.8 Métricas Utilizadas

Para avaliar as arquiteturas e descritores apresentados, foram utilizadas cinco métricas consideravelmente difundidas, que serão descritas a seguir. As quatro primeiras fazem uso da matriz de confusão, que indica um ranking correto ou incorreto das classes em uso, agrupando os resultados em quatro categorias, que são: Verdadeiro Positivo (VP), Verdadeiro Negativo, (VN) Falso Negativo (FN) e Falso Positivo (FP).

18.3.8.1 Acurácia

Consiste no número de acertos (positivos) dividido pelo número total de exemplos. É recomendado utilizar esse tipo de métrica em bases com a mesma proporção de exemplos para cada classe, caso contrário haverá uma falsa impressão de bom desempenho. A acurácia pode ser calculada com a Equação 31.16.

$$Acuracia = \frac{VP + VN}{VP + VN + FP + FN} \quad (18.1)$$

18.3.8.2 Sensibilidade

A sensibilidade é uma medida da proporção de casos reais que foram previstos como positivos (positivos verdadeiros). Em outras palavras, um teste altamente sensível é aquele que identifica corretamente pacientes com uma doença. A sensibilidade pode ser calculada com a Equação 18.2.

$$Sensibilidade = \frac{VP}{VP + FN} \quad (18.2)$$

18.3.8.3 Especificidade

A especificidade de um teste (também chamado de Taxa Negativa Verdadeira) é a proporção de pessoas sem a doença que terão um resultado negativo. Ela mostra o quanto bem o teste identifica pacientes que não têm uma doença. A especificidade pode ser calculada com a Equação 18.3.

$$Especificidade = \frac{VN}{VN + FP} \quad (18.3)$$

18.3.8.4 Kappa

O índice Kappa é um coeficiente de concordância para escalas nominais que mede o relacionamento entre a concordância, além da casualidade, e a discordância esperada (ROSENFIELD; FITZPATRICK-LINS, 1986). Este índice vem sendo recomendado como uma medida apropriada da exatidão por representar inteiramente a matriz de confusão, e não apenas os elementos que se situam na diagonal principal, o que ocorre quando se calcula a exatidão global da classificação (ROSENFIELD; FITZPATRICK-LINS, 1986). O Kappa pode ser encontrado com a Equação 32.6.

$$Kappa = \frac{\Theta_1 - \Theta_2}{1 - \Theta_2} \quad (18.4)$$

Onde,

$$\Theta_1 = \frac{VP + VN}{VP + VN + FP + FN} \quad (18.5)$$

e

$$\Theta_2 = \frac{[(VP + FN)(VP + FP)] + [(VN + FN)(VN + FP)]}{(VP + VN + FP + FN)^2} \quad (18.6)$$

Sobre a avaliação dos classificadores, Landis e Koch (1977a) partitionaram o nível de exatidão do índice Kappa conforme a Tabela 123.

Fonte: Araujo (2014)

Tabela 123 – Nível de exatidão de uma classificação, conforme o valor de índice Kappa.

Índice Kappa (K)	Qualidade
$K < 0.2$	Ruim
$0.2 \leq K < 0.4$	Razoável
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito Bom
$K \geq 0.8$	Excelente

18.3.8.5 Área sob a curva ROC - AUC

Segundo Braga (2001), a área sob a curva ROC (AUC) é um método gráfico simples e robusto que permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. Este método é usado na análise de classificação para determinar qual dos modelos utilizados prediz melhor as classes. Quanto mais próximo o AUC de um modelo chegar a 1, melhor ele será.

O próximo capítulo aborda a metodologia utilizada neste trabalho bem como os resultados alcançados na classificação do melanoma.

18.4 Discussões e Resultados

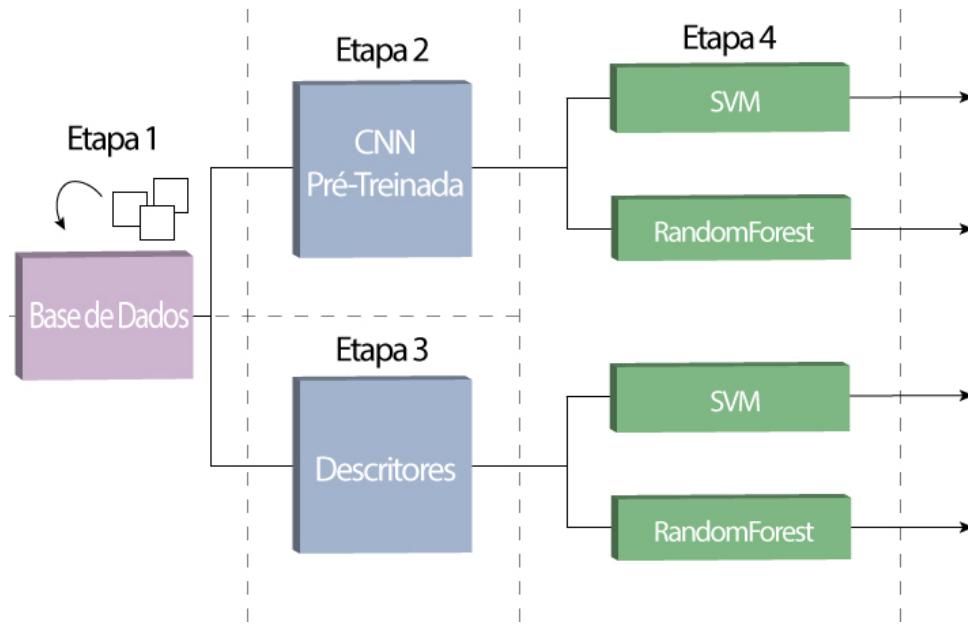
A primeira etapa (1) foi a coleta das imagens de duas classes de lesões cutâneas, o melanoma e nevos atípicos, pois o nevos é um tipo de lesão de pele que apresenta características muito semelhantes ao melanoma, como a cor, o formato e a textura. Para compor a base de dados utilizada nesse trabalho, foram selecionadas imagens do arquivo da ISIC, por ser de fácil acesso e aberta ao público. A segunda etapa (2) consistiu em extrair características dessas imagens utilizando as redes pré-treinadas com pesos da ImageNet, disponíveis na biblioteca Keras. Houve um processamento da base para alterar o tamanho de entrada dos dados para o formato exigido por cada uma das redes e, devido a transferência de aprendizado, as suas camadas de classificação foram removidas. Com isso as arquiteturas e descritores produziram a quantidade de características mostradas na Tabela 124.

Tabela 124 – Tabela com a quantidade de características extraídas pelas redes pré-treinadas e descritores.

Algoritmos	Quantidade
Xception	204800
InceptionV3	131072
ResNet50	100352
VGG16	25088
VGG19	25088
InceptionResNetV2	1536
HOG	576
Daisy	104
LBP	18

Na terceira etapa (3) utilizamos os descritores clássicos para a extração de características da base. A quarta e última etapa (4) foi o uso dos classificadores, *Support Vector Machine* (SVM) e o *RandomForest*, com as características extraídas das arquiteturas e dos descritores para a validação. Nesta última etapa, utilizamos o método de divisão de base de dados onde foi dividido 75% para treino e 25% para teste e depois usamos o método *cross validation*. Nos dois, o número de árvores estimadas do RandomForest foi alterado (entre 10 e 300 árvores) e foram testados ao todo quatro kernels diferentes do SVM: Linear, Polynomial, RBF e Sigmoid, numa tentativa de alcançar maiores resultados. A Figura 139 mostra um esquema da metodologia aqui descrita.

Figura 139 – Esquema da metodologia utilizada neste trabalho. Fonte: Elaborada pelo autor.



18.4.1 Testes Avaliativos

Para a realização dos testes, utilizou-se um computador com acesso à internet, cujas especificações são descritas na Tabela 125.

Tabela 125 – Tabela de especificações da máquina utilizada neste trabalho.

Característica	Descrição
Modelo	X450LC
RAM	6GB
Memoria de Vídeo	2GB
Placa de Vídeo	Nvidia GeForce GT 720M
Processador	Intel I5 4200 @ 1.60GHz
Sistema Operacional	Windows 10

A seguir, os resultados obtidos com a metodologia proposta serão apresentados e discutidos. As métricas nas tabelas estão simbolizadas por suas letras iniciais: Acurácia (A), Sensibilidade (S), Especificidade (E), Kappa (K), Área Sob a Curva ROC (AUC), Precisão (P), Recall (R). As Tabelas 126 e 127 mostram os resultados obtidos com as arquiteturas mencionadas na sessão 18.3 utilizando o método de divisão da base de dados (75% para treino e 25% para teste) e *cross validation*, respectivamente. Já as Tabelas 128 e 129 mostram os resultados obtidos com os descritores utilizando novamente o método de divisão da base de dados (75% para treino e 25% para teste) e *cross validation*, respectivamente.

Como pode ser observado, ResNet50 obteve o melhor resultado nos dois métodos, *cross validation* e divisão da base, sendo o último o melhor, atingindo uma acurácia de 88,5%, sensibilidade de 89,2%, especificidade de 87,7% e uma área sob a curva ROC de 88 com o classificador SVM (*kernel Linear*). Já a VGG19 se saiu melhor dentre as redes e os descritores com o RandomForest (200 árvores estimadas), porém não ultrapassou o SVM. As Tabelas 128 e 129 nos mostram os resultados obtidos com os descritores.

O descritor que obteve o melhor resultado foi o HOG atingindo uma acurácia de 73% com o classificador RandomForest com 200 árvores estimadas, sendo inferior ao da ResNet50. A Tabela 130 apresenta uma breve comparação entre os resultados encontrados utilizando a metodologia proposta e alguns trabalhos citados no Capítulo 18.2. Porém, alguns dos trabalhos utilizaram metodologias e bases diferentes da usada neste trabalho. Portanto, os dados apresentados na Tabela 130, são apenas para ilustrar/comparar os resultados obtidos com os trabalhos da área, sem desmerecer nenhum dos métodos citados.

Tabela 126 – Tabela de resultados das arquiteturas com o método de divisão da base.

SVM					
Arquiteturas	A (%)	S (%)	E (%)	K	AUC
ResNet50	0.88	0.89	0.87	0.76	0.88
InceptionResNetV2	0.81	0.80	0.82	0.63	0.82
VGG19	0.81	0.82	0.80	0.63	0.82
InceptionV3	0.80	0.78	0.82	0.60	0.80
VGG16	0.76	0.76	0.77	0.53	0.77
Xception	0.76	0.76	0.76	0.52	0.76
RandomForest					
Arquiteturas	A (%)	S (%)	E (%)	K	AUC
VGG19	0.82	0.83	0.81	0.64	0.82
ResNet50	0.81	0.82	0.80	0.62	0.81
VGG16	0.81	0.81	0.80	0.62	0.81
InceptionV3	0.80	0.83	0.77	0.60	0.80
Xception	0.77	0.79	0.75	0.54	0.77
InceptionResNetV2	0.75	0.79	0.70	0.49	0.75

Tabela 127 – Tabela de resultados das arquiteturas com o método de *cross validation*.

SVM				
Arquiteturas	A (%)	AUC	P (%)	R
ResNet50	0.80	0.87	0.82	0.77
InceptionV3	0.76	0.83	0.74	0.79
VGG16	0.75	0.82	0.78	0.74
VGG19	0.74	0.81	0.76	0.73
InceptionResNetV2	0.73	0.81	0.72	0.77
Xception	0.73	0.81	0.75	0.70
RandomForest				
Arquiteturas	A (%)	AUC	P (%)	R
VGG19	0.78	0.86	0.83	0.74
Xception	0.78	0.86	0.83	0.74
ResNet50	0.77	0.85	0.78	0.75
VGG16	0.76	0.84	0.79	0.74
InceptionV3	0.75	0.82	0.75	0.74
InceptionResNetV3	0.73	0.81	0.73	0.69

Tabela 128 – Tabela de resultados dos descritores com o método de divisão da base.

SVM					
Descritores	A (%)	S (%)	E (%)	K	AUC
HOG	0.70	0.83	0.81	0.64	0.82
LBP	0.49	0.0	1.0	0.0	0.50
Daisy	0.49	0.0	1.0	0.0	0.50
RandomForest					
Descritores	A (%)	S (%)	E (%)	K	AUC
HOG	0.73	0.81	0.72	0.74	0.82
LBP	0.63	0.66	0.60	0.26	0.63
Daisy	0.61	0.64	0.62	0.54	0.50

18.5 Conclusão

Neste trabalho avaliamos e comparamos o desempenho de seis arquiteturas de redes profundas e três descritores clássicos utilizando transferência de aprendizado para a detecção automática de melanoma em

Tabela 129 – Tabela de resultados dos descritores com o método de *cross validation*.

SVM				
Descritores	A (%)	AUC	P (%)	R
HOG	0.70	0.77	0.72	0.68
Daisy	0.52	0.69	0.54	0.87
LBP	0.49	0.36	0.40	0.80
RandomForest				
Descritores	A (%)	AUC	P (%)	R
HOG	0.73	0.81	0.72	0.74
Daisy	0.67	0.74	0.70	0.66
LBP	0.61	0.64	0.62	0.54

Tabela 130 – Tabela comparativa dos resultados dos trabalhos da área com este trabalho.

Trabalhos	A (%)	S (%)	E (%)	AUC
Mahbod et al. (2019)	-	-	-	0.83
Menegola et al. (2017)	0.83	-	-	-
Chang (2017)	0.80	-	-	-
Yu et al. (2017)	0.94	0.91	0.95	-
Lopez et al. (2017)	-	0.78	-	-
Esteva et al. (2017)	0.72	-	-	-
Majtner, Yildirim-Yayigun e Hardeberg (2016)	0.82	0.53	0.89	0.78
Método Proposto	0.88	0.89	0.87	0.88

imagens médicas. Os resultados obtidos demonstraram que as redes de aprendizado profundo obtiveram um desempenho superior aos descritores em descrever as imagens de pele.

A arquitetura ResNet50 atingiu o melhor resultado dentre os algoritmos usados neste trabalho, com valores acima de 75% em todas as métricas de avaliação, dividindo a base e usando o classificador SVM. Os bons resultados na avaliação fornecem oportunidades para propor novas metas de pesquisa na área de lesões de pele, principalmente em casos de melanoma.

Desta forma, este trabalho apresenta contribuições em duas áreas de estudo. Na área da saúde, oferecendo uma metodologia como segunda opinião a profissionais da área médica para auxiliar no diagnóstico do melanoma. E na área da computação, fornecendo informações importantes para a área de processamento de imagens, como forma de auxiliar pesquisas futuras e também de terceiros.

Como trabalhos futuros, pretende-se: aumentar a quantidade de amostras utilizando mais bases de imagens e também operações de aumento de dados; desenvolver um método automático para a detecção do melanoma em meio a outros tipos de lesões de pele; testar outras técnicas de transferência de aprendizado como por exemplo, o *fine-tuning*;

Detecção e Segmentação do Disco Óptico em Imagens da Retina

Autores: Rafael Araújo Ferreira; Camila Catiely de Sá Almondes; Alcilene Dalilia de Sousa; Flávio Henrique Duarte de Araújo.

Resumo: É notável o crescimento do número de casos tardivamente diagnosticados de glaucoma que ocasionam cegueira em uma boa parcela dos indivíduos afetados, bem como os esforços da comunidade de tecnologia da informação em desenvolver métodos de diagnóstico automático desta patologia. Estes métodos requerem, na maior parte dos casos, ferramentas que auxiliem na preparação das imagens que serão utilizadas como base em seu desenvolvimento. Por esta razão se propôs a criação de um sistema de segmentação automática do Disco Óptico em imagens médicas de fundo de olho, capaz de preparar imagens para serem utilizadas em sistemas de diagnóstico automático. No decorrer deste trabalho se demonstra a viabilidade de adoção das Redes Totalmente Convolucionais como ferramenta para segmentação automática de imagens médicas de fundo de olho. Tal demonstração é feita através do treinamento e teste de quatro modelos de arquitetura de rede em condições semelhantes. A base de dados utilizada é constituída de 400 imagens de fundo de olho em alta qualidade, disponibilizadas publicamente com suas respectivas máscaras e informações que constituem o *ground truth* (verdade-terrestre). Foi demonstrado que dentro das condições utilizadas durante o estudo, as Redes Totalmente Convolucionais são uma solução viável para a segmentação automática do Disco Óptico, tendo seus melhores resultados quando utilizando a arquitetura VGG. Essa arquitetura alcançou uma acurácia de pixel de 88%, um bom resultado quando comparado a trabalhos semelhantes encontrados na literatura.

Keywords: Aprendizado Profundo, Glaucoma, Imagens Médicas, Processamento de Imagens, Redes Totalmente Convolucionais.

19.1 Introdução

O termo Glaucoma é atualmente associado à uma série de neuropatias ópticas progressivas que tem em comum uma lenta e crescente degeneração das células retinais ganglionares e seus axônios, o que resulta em uma aparência peculiar do Disco Óptico (DO) e um simultâneo padrão de perda visual (WEINREB; KHAW, 2004). Dada a viabilidade de diagnóstico através da avaliação da aparência do DO (BUSSEL; WOLLSTEN; SCHUMAN, 2013), abre-se um leque de possibilidades a respeito das formas plausíveis de auxílio tecnológico para um diagnóstico comumente realizado apenas por profissionais humanos com o devido gabarito.

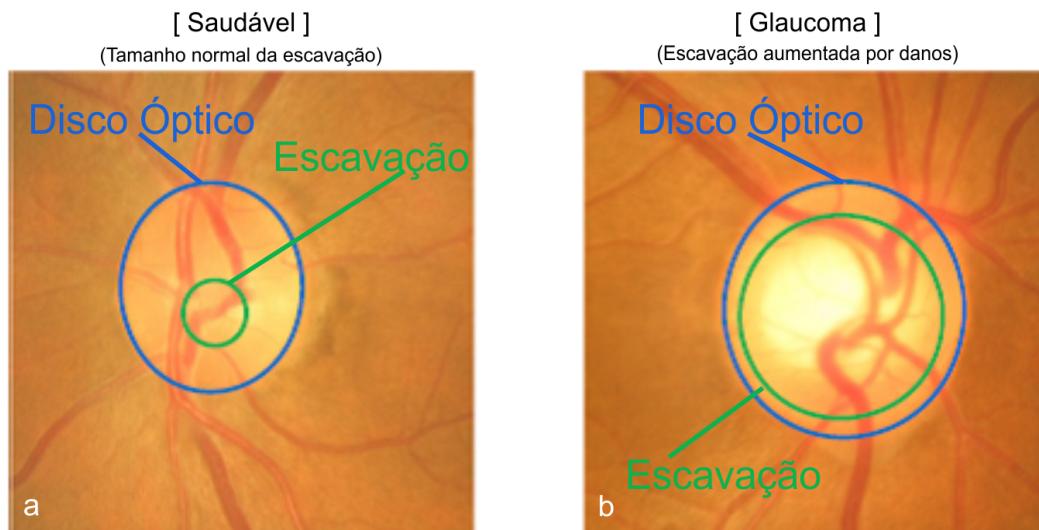
Com origens que remontam ao período da Grécia antiga, o termo glaucoma (derivado do grego *glaukos*) era utilizado para identificar um grupo de distúrbios oculares que levavam à cegueira, normalmente indicados por características como coloração (esverdeada ou esbranquiçada, como se observa na cegueira) ou o próprio tamanho do globo ocular, normalmente alterado pelo aumento da pressão intraocular (sintoma frequentemente observado em diversos quadros de disfunção ótica). Não havia, porém, pertinência a nenhuma patologia em especial (MONTEIRO, 2014).

Atualmente o Glaucoma é associado a um conjunto de neuropatias ópticas lentas e progressivas que resultam em um padrão de perda visual, cuja progressão é frequentemente observada concomitantemente com uma peculiar alteração da aparência do Disco Óptico (WEINREB; KHAW, 2004). Por se tratar de uma patologia crônica degenerativa, o diagnóstico durante os estágios iniciais é de evidente importância e pode ser realizado por meio de exames clínicos como a tonometria, que determina a pressão intraocular (LAURETTI; FILHO, 1997), ou ainda através de métodos de imagem, como por exemplo a Tomografia de Coerência Óptica (TCO), um exame não invasivo que permite a obtenção de imagens da Cabeça do Nervo Óptico (CNO) e da retina, comumente utilizado para a avaliação de danos estruturais glaucomatosos (BUSSEL; WOLLSTEN; SCHUMAN, 2013).

O DO é a principal área do olho humano afetada pela patologia em estudo, é uma denominação clínica criada para descrever as estruturas que envolvem o tecido neural posicionado internamente à cabeça do nervo óptico (REIS ALEXANDRE SOARES CASTRO; BURGOYNE; CHAUHAN, 2012). A escavação patológica do Disco Óptico está frequentemente associada ao glaucoma e outras condições neuro-oftálmicas (ZHANG YI-XIN; WEI, 2014).

Com isso é possível assegurar a existência de características físicas observáveis que podem indicar fortemente a presença do Glaucoma ou condições semelhantes. A Figura 140 ilustra a marcação das áreas de interesse em imagem de olho saudável (a) e de olho glaucomatoso (b) sendo que o círculo azul sinaliza o DO enquanto o verde a Escavação.

Figura 140 – Escavação em Olho Saudável (a) e Glaucomatoso (b).



Fonte: Fonte: Adaptação de ETH Zurich Department of Computer Science.

Uma vez observada a viabilidade de diagnóstico do Glaucoma através da avaliação da aparência do Disco Óptico, decidimos iniciar a criação de um sistema que segmentasse imagens médicas de fundo de olho. Isto deve resultar num aumento de praticidade na preparação de imagens para sistemas de diagnóstico automático.

O objetivo geral deste trabalho é desenvolver um método de segmentação do Disco Óptico em imagens de fundo de olho, que seja robusto e performe tão bem quanto, ou melhor que os disponíveis na literatura. Como objetivos específicos temos: Aplicar Redes Totalmente Convolucionais (RTC's) para segmentação do DO como área de interesse; Determinar a melhor arquitetura de RTC para cumprir este objetivo.

A investigação da literatura permitiu identificar as técnicas baseadas em Aprendizado Profundo (AP) mais utilizadas, tanto para fins gerais quanto especificamente para tratamento de imagens médicas. Consequentemente, após um levantamento das características de cada uma das técnicas utilizadas, foi possível determinar que as Redes Totalmente Convolucionais seriam primordialmente a técnica mais adequada para este trabalho, tendo sido utilizada na detecção ([ABDULLA, 2017](#)) e segmentação semântica de objetos ([SHELHAMER; LONG; DARRELL, 2016](#)), mas não para segmentação com foco na preparação de imagens para classificadores de imagens médicas e ferramentas de diagnóstico automatizado. Com isto, seguiram-se as etapas de desenvolvimento, testes laboratoriais, correções de código, e consequente treinamento dessa rede para adequação de pesos e posterior obtenção de resultados, que foram comparados às segmentações realizadas por profissionais da área médica, gerando os índices apresentados na Seção [19.5](#) deste trabalho.

A principal contribuição deste trabalho é a avaliação das Redes Neurais Totalmente Convolucionais na segmentação semântica (seguindo o contorno real do DO) de imagens médicas de fundo de olho.

Nas próximas seções o trabalho obedece à seguinte disposição. Na seção [19.2](#) são apresentados conceitos fundamentais ao desenvolvimento e compreensão do trabalho, bem como a base de dados e técnicas utilizadas. Em sequência, a seção [19.3](#) contém brevemente descritos os principais trabalhos relacionados encontrados na literatura, bem como suas principais semelhanças e diferenças em relação a este trabalho. Logo em seguida, na seção [19.4](#), são abordadas as principais características deste trabalho, tais como ambiente de desenvolvimento, método desenvolvido, resultados alcançados e outras informações úteis sobre a solução aqui implementada. Por fim, na seção [19.6](#) são apresentadas as principais conclusões deste trabalho, bem como possíveis trabalhos futuros.

19.2 Referencial Teórico

Nesta seção são apresentados os principais conceitos que fundamentaram a construção do método desenvolvido.

19.2.1 Redes Neurais Articiais

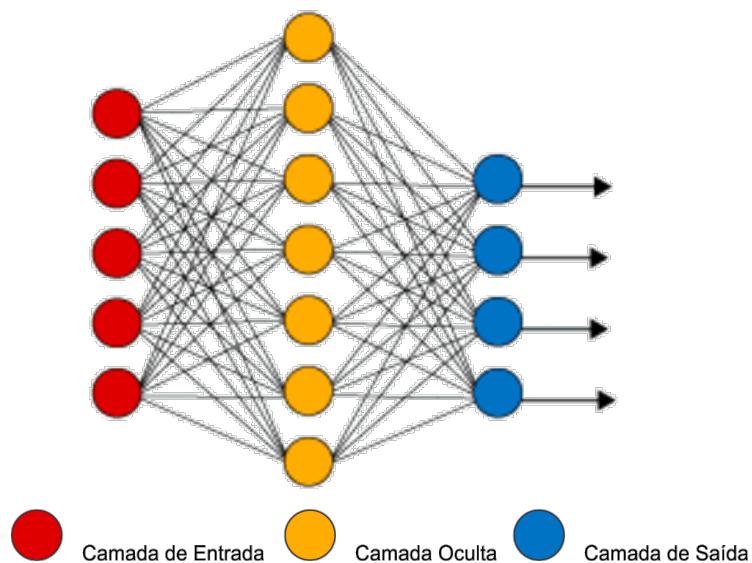
Redes Neurais Articiais (RNA) são uma classe de algoritmos de aprendizagem de máquina que são capazes de aprender à partir de coleções de dados e se especializar no reconhecimento de padrões, inspirados na estrutura e funcionamento do cérebro, onde células especializadas em tratar estímulos (neurônios) recebidos como impulsos elétricos se organizam em rede com o propósito de interpretar os estímulos recebidos e determinar quais atitudes devem ser tomadas. Semelhantemente, as redes neurais articiais são constituídas de componentes lógicos (Neurônios Articiais) que recebem entradas adaptadas (metadados) e “descobrem” quais as saídas adequadas para cada uma destas entradas ([ROSEBROCK, 2017a](#)). A Figura [141](#) traz a ilustração de uma RNA.

19.2.2 Aprendizado Profundo

Deep Learning, em português, Aprendizado Profundo é um subcampo da aprendizagem de máquina que, por sua vez, é um subcampo da inteligência artificial, e assim como seu campo geral tem como objetivo central desenvolver técnicas e algoritmos que possam ser usados para resolver problemas com os quais os humanos estão habituados a lidar de forma intuitiva e praticamente automática ([ROSEBROCK, 2017a](#)). A Figura [142](#) demonstra uma Rede de Aprendizado Profundo.

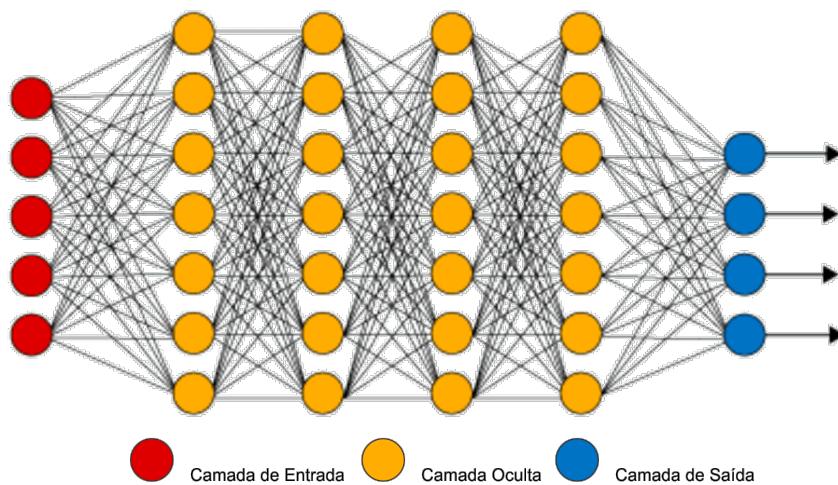
”Métodos de Aprendizado Profundo são métodos de aprendizagem representativa com múltiplos níveis de representação, obtidos ao se compor módulos simples, mas não lineares capazes de transformar a representação em um nível baixo (iniciando com a entrada bruta)

Figura 141 – Ilustração de uma Rede Neural Artificial.

Fonte: Adaptação de *Deep Learning Book*.

2

Figura 142 – Ilustração de uma Rede de Aprendizado Profundo.

Fonte: Adaptação de *Deep Learning Book*.

3

em uma representação em nível mais alto e sutilmente mais abstrato” - ([LECUN; BENGIO; HINTON, 2015](#), p. 436)

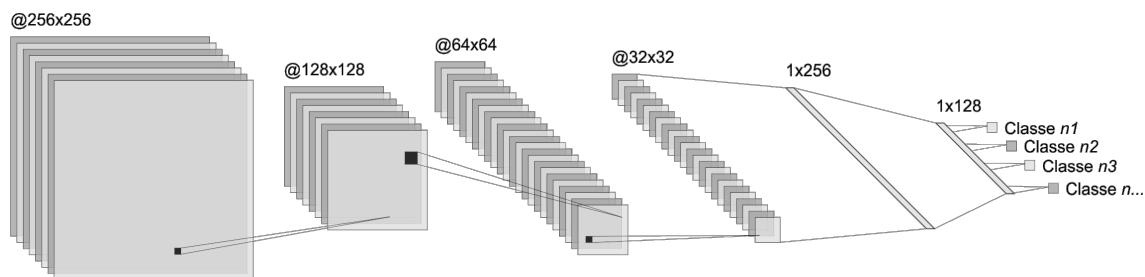
Existem inúmeras abordagens de Aprendizado Profundo derivadas de modificações das Redes Neurais Artificiais, criadas e aplicadas para os mais devidos fins. Porém, no que tange ao PDI, algumas abordagens se destacam; entre elas as Redes Neurais Convolucionais (RNC) e as Redes Totalmente Convolucionais (RTC). Nas seções à seguir serão apresentados os fundamentos necessários para a compreensão destas técnicas.

19.2.3 Redes Neurais Convolucionais

Análogas às Redes Neurais Artificiais tradicionais (RNA), Redes Neurais Convolucionais são compostas de neurônios artificiais que se auto otimizam através da aprendizagem. Cada neurônio recebe uma entrada e faz uso de uma função de ativação, comportamento fundamental de dezenas de Redes Neurais Artificiais. A única diferença notável entre RNC e RNA é o fato de as RNC serem primordialmente utilizadas no campo de reconhecimento de padrões em sinais, o que permite a codificação de características específicas das imagens na arquitetura, tornando-as mais recomendadas e utilizáveis para tarefas com foco em PDI (THEANO, 2013).

A grande vantagem da aplicação de RNC's para o Processamento Digital de Imagens é sua grande capacidade de extrair características, por trabalhar com convoluções e *pooling*, ou seja, criando “sub-quadros de bits” de forma que seja possível realçar contornos e áreas de interesse de cada imagem (ROSEBROCK, 2017a), como ilustrado na Figura 143.

Figura 143 – Ilustração do Funcionamento de uma Rede Neural Convolucional.



Fonte: Elaborado pelo autor.

O processo de subdivisão da imagem em subgrupos de *pixels* cada vez menores é denominado *pooling*, enquanto Convolução caracteriza pela extração de características destes subgrupos. Vale lembrar que o processo realizado em camadas de *pooling* está frequentemente associado a camadas de *padding* que auxiliam na diminuição do volume de dados, camadas de *dropout* que ignoram unidades aleatórias para evitar falso aprendizado, bem como camadas *upsampling*, que auxiliam na deconvolução.

Como visto na Figura 143, em Redes Neurais Convolucionais as imagens de entrada são divididas em “janelas” que serão processadas e cujas características serão somadas e propagadas para as próximas camadas da rede neural. Contudo, considerando a necessidade de gerar imagens propriamente ditas como resultado do processamento, surgem as Redes Totalmente Convolucionais, apresentadas na seção 19.2.4.

19.2.4 Redes Totalmente Convolucionais

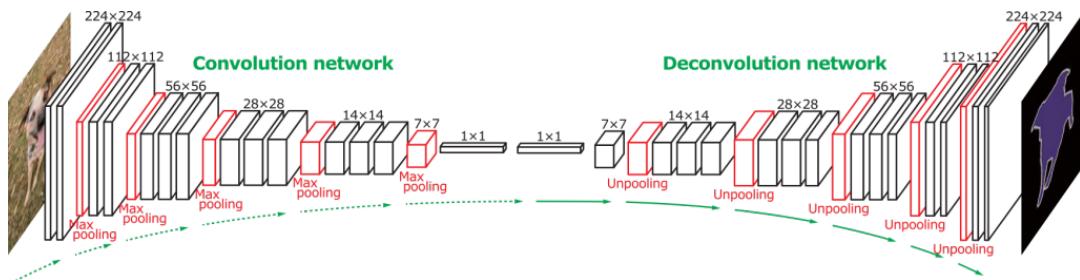
Derivadas de uma modificação das RNC's convencionais, as Redes Totalmente Convolucionais (RTC's) são de grande valia para sistemas de segmentação automática e detecção de objetos, tendo em vista que sua arquitetura substitui as últimas camadas de uma RNC comum por camadas deconvolucionais, criando assim um processo reverso e tornando janelas de bits maiores a cada camada, o que permite que a saída da rede seja uma imagem semelhante à entrada, porém, segmentada. Após o devido treinamento as RTC's são capazes de detectar objetos no contexto das imagens, como ilustrado na Figura 144. Por observar esta capacidade e a inexistência de trabalhos semelhantes, as RTC's foram escolhidas como objeto de estudo desse trabalho.

A próxima seção apresenta a base de imagens utilizada para o treinamento e avaliação dos algoritmos no presente trabalho.

19.2.5 Base de Imagens

Para o desenvolvimento de um sistema de Processamento Digital de Imagens é preciso constituir uma base de imagens confiável para que seja possível realizar o processo de treinamento, bem como avaliar

Figura 144 – Ilustração do Funcionamento de uma Rede Totalmente Convolucional.



Fonte: Fonte: Azoft Digital Products.

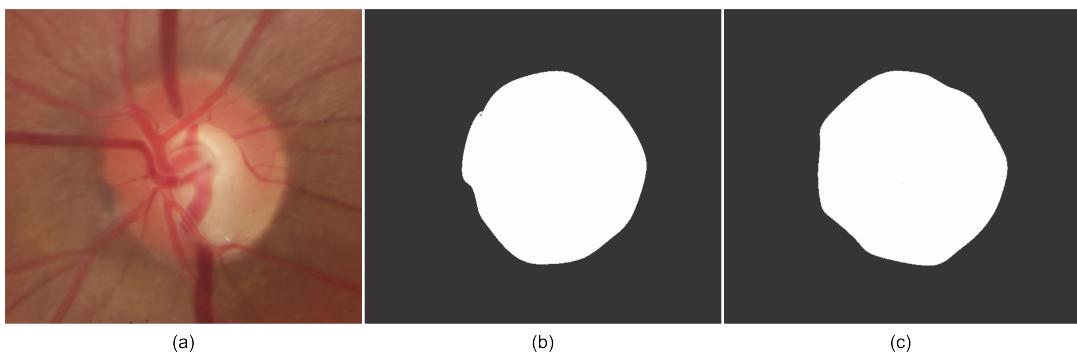
4

corretamente os resultados obtidos pelo mesmo. Com isso em mente, apresenta-se durante esta seção e suas subseções, a base de imagens utilizada para treinamento e avaliação do método desenvolvido.

19.2.6 RIM-ONE

A biblioteca RIM-ONE é uma coleção pública de imagens de fundo de olho com elevados padrões de qualidade no que diz respeito à definição das fotografias e sua documentação. A mesma foi elaborada com o objetivo de prover segmentações acuradas de imagens da cabeça do Nervo Óptico que servissem de base para a concepção de novos sistemas de detecção automática do Glaucoma (FUMERO et al., 2011b). O alto nível de qualidade da base é certificado pela presença em todas as suas versões de um arquivo de texto referente a cada uma das imagens. Este arquivo contém informações importantes, como o centroide médio de todos os contornos feitos por profissionais, bem como eixos em 8 direções partindo do centroide até o contorno do Disco Óptico. A Figura 145 demonstra imagens desta base.

Figura 145 – Demonstrativo das Imagens Contidas na Base de Dados Utilizada.



Fonte: Fonte: RIM-ONE R1.

Observa-se na Figura 145 em (a) uma imagem de fundo de olho contida na primeira de três versões disponibilizadas da RIM-ONE. Em (b) e (c) observam-se segmentações manuais realizadas por profissionais que auxiliaram na construção da base. É importante ressaltar que cada uma das versões da biblioteca tem características únicas. Observe:

- RIM-ONE R1: Primeira versão da biblioteca, conta com um total de 158 imagens de 478 pixels de altura e 436 de largura, divididas e organizadas como: Glaucoma Profundo (14), Glaucoma Mediano (14), Glaucoma Leve (12), e Normais (118). Contém também cinco segmentações de profissionais para cada uma das imagens e um arquivo de metadados referentes a cada imagem.

- RIM-ONE R2: Segunda versão da biblioteca, constituída de um total de 455 imagens de 400 *pixels* de altura e 469 de largura, divididas e organizadas como Glaucoma e Suspeitos (200) e Normais (255). Contém também um arquivo de metadados referentes a cada imagem. Não contém, entretanto, imagens referentes à segmentação de profissionais, sendo o arquivo de metadados a única referência para *ground truth*, por esse motivo essas imagens não foram utilizadas neste trabalho.
- RIM-ONE R3: Terceira e mais recente versão da biblioteca, constituída de 159 imagens estéreo (olhos direito e esquerdo) de 2144 *pixels* de altura e 1424 de largura, divididas e organizadas como Glaucoma e Suspeitos (74) e Normais (85). Contém também duas imagens referentes à segmentações de profissionais para cada uma das imagens, bem como arquivos de metadados referentes a cada imagem.

As três versões da biblioteca são disponibilizadas gratuitamente no site do MIAG⁵, somando 1072 imagens de fundo de olho em alta qualidade.

19.3 Trabalhos Relacionados

No que concerne aos trabalhos desta linha de pesquisa, pode-se observar uma tendência de crescimento no uso de técnicas de Aprendizado Profundo, tanto para segmentação de objetos em imagens comuns, quanto para segmentação de imagens médicas. A seguir são descritos os trabalhos relacionados com objetivos semelhantes a este.

1. STAAL et al. (2010) adotam a decomposição de alterações na aparência dos vasos sanguíneos baseado nas características impostas por cada condição clínica para detectar patologias óticas. Diferencia-se deste trabalho tanto por sua abordagem quanto pelas características observadas, não baseando o diagnóstico no aumento patológico da escavação do DO, mas na aparência dos vasos. Foi igualmente importante para este trabalho por trazer informações úteis a respeito da fisionomia do olho humano e características relevantes.
2. RAJAPUT; RESHMI; SIDRAMAPPA (2011) fazem uso de uma abordagem baseada em transformação de Hough para segmentação do DO e detecção da região da fóvea e seu centro. Assemelha-se a este trabalho tanto por seu objetivo, quanto pela base de imagens utilizada (RIM-ONE R1). Diferencia-se por utilizar uma abordagem diferente.
3. ALMAZROA et al. (2015) propõem um estudo a respeito das metodologias de segmentação do DO e escavação para detecção do glaucoma em imagens. Diferencia-se deste trabalho por não focar em uma técnica em específico, nem tampouco tratar da técnica aqui estudada. Foi de fundamental importância para este trabalho, principalmente por direcionar às linhas de pesquisa e abordagens já utilizadas para este fim e presentes na literatura.
4. SRIVASTAVA et al. (2015) faz o uso de Aprendizado Profundo na implementação de um sistema de segmentação automática do Disco Óptico em imagens da retina, resistente à Atrofia Parapapilar. Semelhantemente a este trabalho, utiliza uma abordagem de Aprendizado Profundo para segmentação automática e tem como saída uma imagem segmentada em “estado de arte”. Diferencia-se porém por sua abordagem ser baseada em camadas estacadas de codificadores automáticos.
5. TAN et al. (2017) faz o uso de uma RNC para segmentar as regiões do DO, Fóvea e Vasculatura Retinal em imagens médicas de fundo de olho. Se assemelha a este trabalho por ter entre seus objetivos a segmentação do DO em imagens médicas de fundo de olho. Diferencia-se abordagem utilizada para atingir este fim.

Dando sequência, são apresentados também os trabalhos relacionados que fazem uso de técnicas semelhantes para objetivos diferentes deste trabalho.

⁵ Disponível em <http://medimrg.webs.ull.es/>. Acesso em 11/11/2018.

1. PINHEIRO; COLLORBERT (2014) utiliza Redes Neurais Convolucionais Recorrentes, uma técnica deveras semelhante à aplicada neste trabalho, para criar legendas de objetos em cenas (imagens). Assemelha-se com este trabalho por ter o objetivo de preparar imagens para uso posterior. Difere, porém, na técnica utilizada, bem como no espectro do objetivo, tratando de objetos em geral e não de áreas de interesse em imagens médicas.
2. SHELHAMER; LONG; DARRELL (2016) faz uso da abordagem RTC's para segmentação semântica de objetos sem utilização de maquinário posterior, ou seja, aplica a técnica para retirar diretamente dela uma saída em “estado de arte” (uma imagem propriamente dita). Diferencia-se deste trabalho pelo objetivo geral, bem como pela arquitetura da rede montada, sendo neste caso uma RTC pixel-a-pixel, e nesse uma arquitetura estacada de *pooling/unpooling* progressivo. Foi de extrema importância por ser o primeiro trabalho que considerava a saída em “estado de arte”, ou seja, a possibilidade de ter uma imagem propriamente dita como saída da rede.
3. PAKHOMOV et al. (2017) aplica as RTC's para segmentação e adição de máscaras com o intuito final de detecção e identificação de objetos. Difere deste trabalho no que diz respeito ao objetivo final, ao segmentar objetos e não regiões de interesse em imagens médicas. Entretanto, explica de forma elementar os modelos mais utilizáveis de arquitetura de rede, contribuindo amplamente para o desenvolvimento do presente trabalho.

Após analisar os trabalhos relacionados optou-se pela utilização de RTC's por perceber que elas obtiveram bons resultados na detecção e segmentação de imagens. Além disso, não foram encontrados trabalhos que lhes utilizassem para o propósito deste trabalho. O desenvolvimento do método proposto é descrito no Capítulo 19.4.

19.4 Método Proposto

No decorrer desta seção são abordados ambiente e metodologia utilizada durante o desenvolvimento do presente trabalho.

19.4.1 Modelos de Rede

Após avaliar a bibliografia encontrada a respeito do tema, foram encontrados diversos modelos de arquitetura compatíveis com a proposta desse trabalho, os quais foram comparadas e avaliadas para determinar qual a melhor performance dentro do objetivo do trabalho. Com isso, foi possível selecionar quatro principais modelos para estudo mais profundo e consequente avaliação da melhor performance. Estes modelos foram escolhidos por apresentar arquiteturas diferentes, seja pela quantidade de camadas ou pelas diferentes formas de abordar o processo de convolução/deconvolução característicos de uma RTC. Tais modelos são descritos à seguir.

- **Modelo 1 - RTC ResNet**

A RTC ResNet é uma arquitetura deveras custosa. Com mais de 100 camadas se dividem entre funções de *pooling* (ver seção 19.2.3), convolução, *batch normalization* (normalização de lotes), *merge* (fusão), ativação e *upsampling* (sobre-amostragem) (HUNDT, 2017). Uma ilustração da arquitetura da RTC ResNet é mostrada na Figura 146

- **Modelo 2 - RTC VGG**

Construída com apenas 30 camadas e consideravelmente mais leve que o modelo ResNet, a RTC VGG é uma arquitetura menos custosa na qual a maior parte das camadas são de *pooling* e convolução (JU, 2016). Uma ilustração de sua arquitetura pode ser vista na Figura 147.

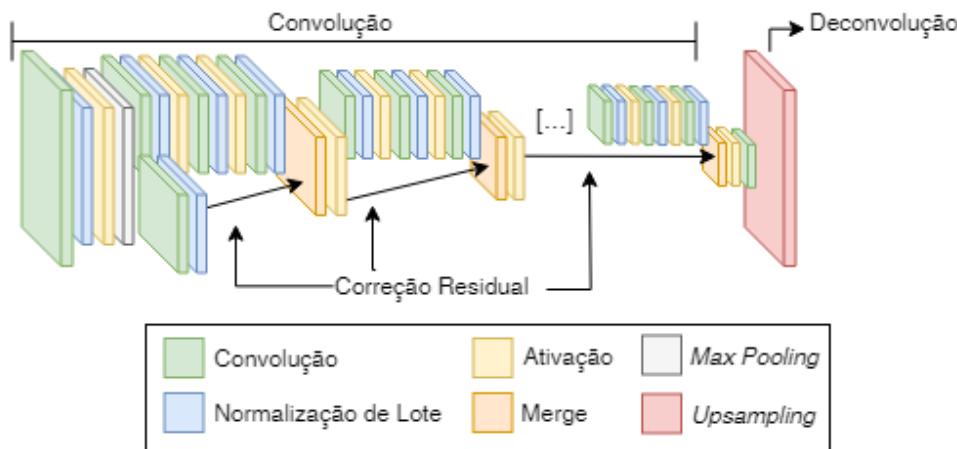
- **Modelo 3 - RTC Segnet**

Utilizando 38 camadas, a arquitetura RTC Segnet é principalmente composta de camadas convolucionais, aliadas a camadas de *pooling* e *padding* (ver seção 19.2.3). É possível observar uma ilustração desta arquitetura na Figura 148.

- **Modelo 4 - RTC Unet**

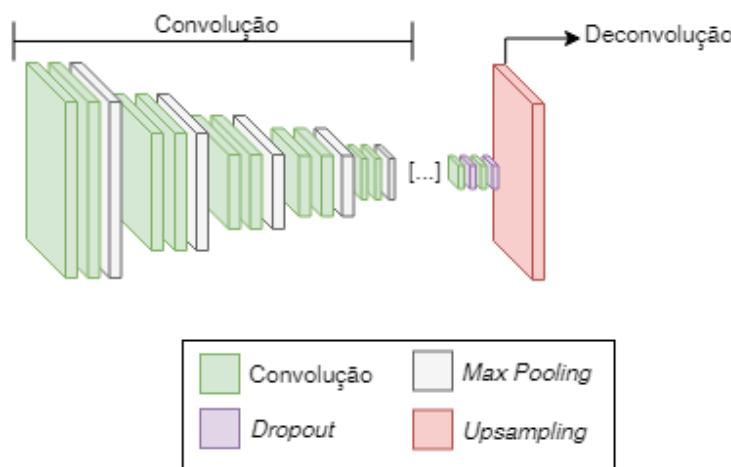
Constituída de mínimas 24 camadas, a RTC Unet é com certeza a mais simples das arquiteturas utilizadas, se mantendo fiel ao conceito mais básico de RTC, já ilustrado na Figura 144, por utilizar camadas convolucionais de tamanhos decrescentes "sobrepostas" com camadas deconvolucionais.

Figura 146 – Ilustração da Arquitetura ResNet.



Fonte: Fonte: Elaborado pelo autor.

Figura 147 – Ilustração da Arquitetura do modelo VGG.

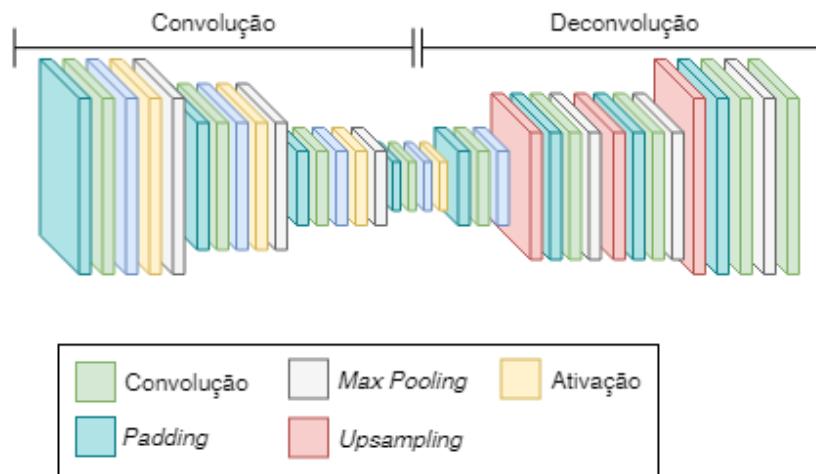


Fonte: Fonte: Elaborado pelo autor.

19.4.2 Treinamento e Avaliação

No processo de treinamento dos modelos avaliados foram utilizadas 400 imagens integrantes da primeira e terceira versão da biblioteca RIM-ONE. Isso porque a segunda versão não possui imagens da segmentação feita por profissionais (*ground truth*). Em contrapartida, a terceira versão da biblioteca é constituída de ima-

Figura 148 – Ilustração da Arquitetura do modelo Segnet.



Fonte: Elaborado pelo autor.

gens estéreo, o que nos garante duas unidades para cada imagem, bem como suas referentes segmentações. A quantidade de imagens utilizadas em cada etapa são descritas na Tabela 131.

Tabela 131 – Descrição do Experimento.

Etapa	Total de Imagens	Nº Glaucomatosas	Nº Normais	Épocas
Treinamento e Validação	300	138	162	100
Teste	100	50	50	-

Fonte: Elaborado pelo autor.

A avaliação de performance dos modelos foi realizada com base em quatro métricas consideravelmente difundidas. Tais técnicas são: Acurácia de Pixels, Acurácia Média, média da Intercessão sobre União (SHE-LHAMER; LONG; DARRELL, 2016) e Coeficiente Dice de Similaridade (DICE, 1945). Considerando que n_{ij} seja o número de pixels da classe i preditos como pertencendo à classe j , onde existem n_{cl} classes diferentes, que $t_i = \sum_j n_{ij}$ seja o número total de pixels da classe i , e que A e B são conjuntos de pixels pertencentes às áreas de interesse em duas imagens, calculamos:

- **Acurácia de Pixels:** $Ac_Pixel = \sum_i n_{ii} / \sum_i t_i$.
- **Acurácia Média:** $Ac_Média = (1/n_{cl}) \sum_i n_{ii} / t_i$.
- **Média da Intercessão sobre União:** $MIU = (1/n_{cl}) \sum_i n_{ii} / (t_i \sum_j n_{ji} - n_{ii})$.
- **Coeficiente Dice de Similaridade:** $Dice = 2(A \cap B) / (A + B)$.

Quando em se tratando do Coeficiente Dice de similaridade, temos comparações matemáticas entre a segmentação realizada pelos modelos sob avaliação e a segmentação disponibilizada pelos produtores da base de dados (*ground truth*), considerando boas segmentações os resultados que apresentem pontuação final superior a 0,70 (ZOU et al., 2004).

Os resultados obtidos ao fim das etapas de treinamento e testes avaliativos serão apresentados e discutidos na Seção 19.5.

19.5 Resultados e Discussões

Para a realização dos experimentos aqui descritos, utilizou-se um computador com acesso à internet, cujas especificações são descritas na Tabela 132.

Tabela 132 – Descrição do Ambiente de Desenvolvimento.

Característica	Descrição
Modelo	XL555LF
RAM	12GB
Memória de Vídeo	2GB
Placa de Vídeo	Nvidia GeForce 930M
Processador	Intel I5 5200 @ 2.20GHz
Sistema Operacional	Linux Ubuntu 14.0

Fonte: Fonte: Elaborado pelo autor.

Todos os modelos foram treinados nas condições descritas na Tabela 131 e tiveram seu desempenho observado e avaliado de acordo com as métricas descritas na Seção 19.4.2. O resultado dos testes executados são demonstrados na Tabela 133

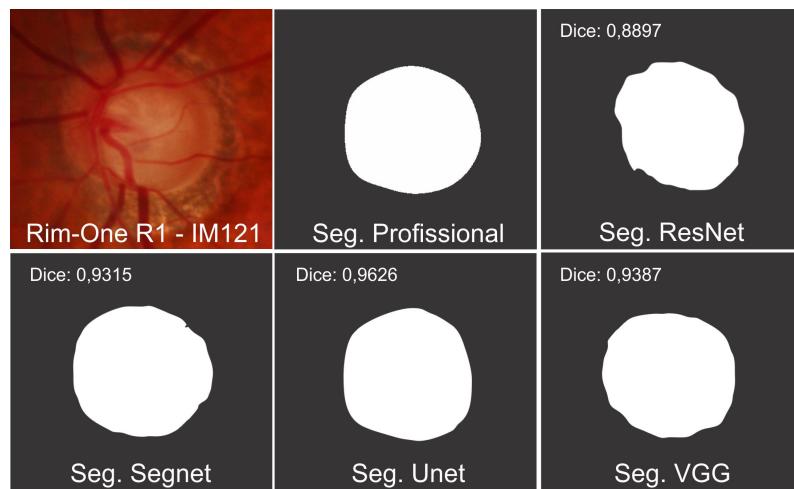
Tabela 133 – Resultados Obtidos.

Modelo	Ac. Pixel	Ac. Média	MIU	Dice
Resnet	0,7204	0,4795	0,3486	0,7062
Segnet	0,8327	0,6143	0,4829	0,8138
Unet	0,8725	0,6309	0,4987	0,8516
VGG	0,8891	0,6537	0,5330	0,8729

Fonte: Fonte: Elaborado pelo autor.

Ao observar os resultados obtidos pela RTC VGG, bem como a notável performance dos demais modelos em estudo, não apenas se prova a viabilidade de utilização de RTC's na segmentação semântica de imagens médicas, mas também se indica uma abordagem de Aprendizado Profundo ainda pouco explorada, mas de grande valia para uma linha de pesquisa inteira. Observe na Figura 149 exemplos de segmentações realizadas pelos modelos em estudo.

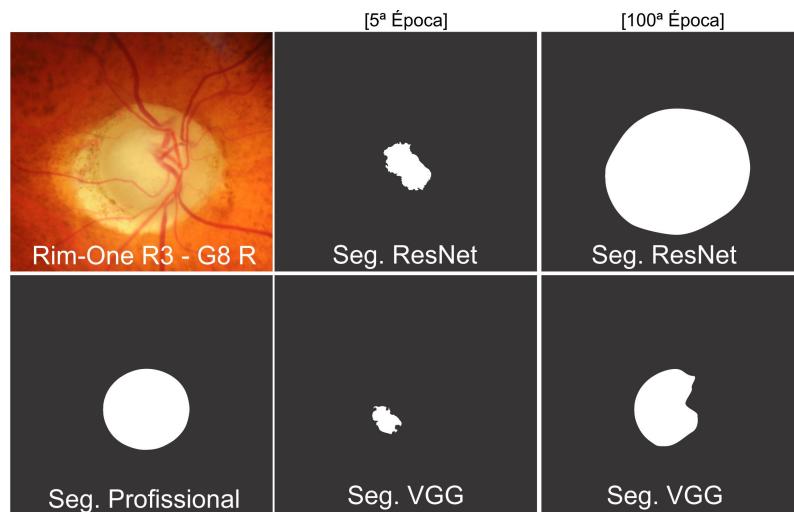
Figura 149 – Demonstração das Segmentações Feitas por Cada Modelo.



Fonte: Fonte: Adaptação de RIM-ONE RI.

Existiram também falhas de segmentação, tanto durante o treinamento como também ao fim, durante os testes. Observe na Figura 150 a demonstração da segmentação de uma imagem utilizando pesos da 5^a e 100^a épocas, ambas consideradas incorretas.

Figura 150 – Demonstração de Falhas na Segmentação.



Fonte: Fonte: Adaptação de RIM-ONE R3.

Realizando um rápido comparativo entre os trabalhos apresentados no Capítulo 19.3 que se utilizam de abordagens e métricas avaliativas semelhantes, apresentado na Tabela 134, percebe-se que o sistema proposto teve uma performance aceitável, podendo ser defendido como uma solução viável para a segmentação automática de imagens médicas. É possível notar que o trabalho que obtive índices de acurácia superior aos obtidos por este trabalho utilizou uma quantidade muito maior de amostras na sua base de treinamento, sendo estas amostras de bancos de imagens privados.

Tabela 134 – Comparativo Entre os Resultados Obtidos.

Trabalho	Ac_Pixel (%)	Nº de Amostras
SRIVASTAVA et al.	90,3	200.000
RAJAPUT; RESHMI; SIDRAMAPPA	88,7	169
RTC VGG	88,9	400

Fonte: Fonte: Elaborado pelo autor.

É importante também ressaltar que a utilização de maiores bases de treinamento implica na utilização de maior poder computacional, o que se pode observar quando ([TAN et al., 2017](#), p. 12) mencionam ao descrever seu ambiente, uma estação de trabalho equipada com dois processadores Intel Xeon de 2.20Ghz (E5-2650) e 512GB de RAM para processar suas mais de 75000 imagens. Embora [TAN et al.](#) possua objetivos semelhantes, não foi mencionado neste comparativo por fazer o uso de métricas avaliativas diferentes.

19.6 Conclusão

Neste trabalho avaliamos o desempenho de quatro arquiteturas de Redes Totalmente Convolucionais quando aplicadas para segmentação automática do Disco Óptico em imagens médicas de fundo de olho. Os resultados obtidos demonstram que nas condições do nosso experimento, a aplicação das RTC's em arquitetura VGG performou com maior acurácia e menores índices de perda.

De forma geral, considera-se cumprido o objetivo primário do trabalho. Os resultados obtidos serão de considerável valia para a continuação da pesquisa, bem como de futuros trabalhos de terceiros que poderão ter como base os dados aqui levantados.

São identificadas como principais contribuições deste trabalho o desenvolvimento de um sistema de segmentação automática de imagens médicas, a utilização de uma abordagem de Aprendizado Profundo ainda não explorada para este objetivo, como também a comparação de arquiteturas com diferentes número de camadas.

São muitas as possibilidades de continuação desta linha de pesquisa. Entre elas são destacadas a identificação de características e parâmetros que possam melhorar o desempenho do modelo implementado, implementação da ferramenta de detecção automática de danos glaucomatosos em imagens de fundo de olho, com a possibilidade de diagnose em tempo real com base em *stream* de vídeo, aproveitando do sistema criado para segmentação em tempo real; exploração do potencial das RTC's na segmentação de imagens; levantamento de dados e elaboração de artigos que sejam relevantes neste campo de pesquisa.

Parte II

Avaliação de Desempenho de Sistemas

Avaliação de Desempenho e Disponibilidade da Mobile Edge Computing usando Redes de Petri Estocásticas

Autores: Carlos Brito e Francisco Airton Silva

Resumo

A Mobile Edge Computing (MEC) é uma arquitetura de rede que surgiu como alternativa para reduzir a latência da rede, aproveitando a computação em nuvem e recursos de computação de borda. As soluções MEC precisam alocar dinamicamente as solicitações o mais próximo possível de seus usuários. No entanto, a capacidade da máquina servidora de recursos pode influenciar diretamente o desempenho e a disponibilidade da MEC. Os profissionais de sistemas devem compreender a relação entre desempenho e disponibilidade nos estágios de projeto de sistemas. Os modelos analíticos são adequados para tal objetivo. Portanto, este capítulo propõe modelos de redes de Petri estocásticas (SPN, do inglês *Stochastic Petri Net*) para avaliar o desempenho e a disponibilidade de ambientes MEC. Diferente de trabalhos anteriores, nossa proposta inclui métricas únicas, como probabilidade de descarte e uma análise de sensibilidade que orienta as decisões de avaliação. Os modelos são altamente flexíveis considerando quatorze transições no modelo básico e vinte e cinco transições no modelo estendido. O modelo de desempenho foi validado com um experimento real cujo resultado indicou igualdade entre experimento e modelo com valor-p igual a 0,684 pelo Teste-T. Em relação à disponibilidade, os resultados do modelo estendido, diferentemente do modelo base, ficam sempre acima de 99%, pois apresenta redundância nos componentes que estavam impactando a disponibilidade no modelo base. Também apresentamos uma análise numérica que serve como um guia prático para auxiliar os gerentes de infraestrutura de computadores no projeto de suas arquiteturas, tornando possível avaliar o trade-off entre MRT e utilização de recursos.

20.1 Introdução

Os dispositivos móveis mudaram a maneira como as pessoas vivem nos últimos anos. De acordo com o Statista (um portal alemão online de estatísticas), o número de dispositivos móveis será de cerca de 16,8

bilhões em 2023 (STATISTA...,). Infelizmente, os dispositivos móveis ainda apresentam recursos limitados em termos de vida útil da bateria, armazenamento e capacidade de processamento. As redes celulares devem suportar baixa capacidade de armazenamento, alto consumo de energia, baixa largura de banda e altas latências (ORSINI; BADE; LAMERSDORF, 2015). Além disso, o crescimento exponencial da Internet das Coisas (IoT, do inglês *Internet of Things*) promete tornar as redes sem fio ainda mais desafiadoras (BORGIA et al., 2016) (KERTESZ; PFLANZNER; GYIMOTHY, 2018).

Muitas pesquisas têm proposto arquiteturas otimizadas para este contexto, como a Mobile Cloud Computing (MCC) (MAROTTA et al., 2015) (OTHMAN et al., 2015) (VAQUERO et al., 2008). A MCC é a integração de computação em nuvem e computação móvel, que fornece recursos adicionais para dispositivos móveis centralizando seus recursos na nuvem (DINH et al., 2013). A técnica chamada de *Offload* pode transferir tarefas que usam muitos recursos para nuvens remotas. A MCC pode reduzir os problemas de latência ao fornecer um modelo seguro e eficiente. No entanto, os recursos da nuvem estão frequentemente longe dos usuários finais (JARARWEH et al., 2016a). Portanto, com vários dispositivos móveis, a MCC enfrenta desafios notáveis, como alta latência, vulnerabilidade de segurança e transmissão de dados limitada.

A Mobile Edge Computing (MEC) surgiu como uma alternativa mais eficiente à arquitetura MCC. O principal objetivo da MEC é enfrentar os desafios que a MCC tem enfrentado, implantando recursos ainda mais perto dos usuários — na borda da rede. Assim, a computação e o armazenamento são executados mais perto do dispositivo de origem (JARARWEH et al., 2016b). A MEC visa permitir que bilhões de dispositivos móveis conectados executem os aplicativos de computação intensiva em tempo real diretamente na borda da rede. As características distintivas do MEC são sua proximidade com os usuários finais, suporte à mobilidade e implantação geográfica densa dos servidores MEC (AHMED; AHMED, 2016). Como a MEC ainda é considerado um tema recente, algumas lacunas de pesquisa ainda precisam ser exploradas, como a avaliação de desempenho de arquiteturas da MEC.

A avaliação da disponibilidade é outro tema de interesse da área da MEC. A MEC foi projetada para prestar serviços em tempo real, portanto, robustez e disponibilidade são requisitos essenciais, uma vez que os recursos estão próximos aos clientes, prestando-lhes serviços críticos. Todos os componentes de um sistema estão sujeitos a falhas e as falhas devem ser reparadas o mais rápido possível. Avaliar a disponibilidade de uma arquitetura MEC pode ser caro, considerando experimentos práticos. Assim, é necessário avaliar as arquiteturas MEC antes mesmo de uma implantação real, usando modelos analíticos, por exemplo.

Portanto, faltam estudos que avaliem as arquiteturas do MEC por meio de modelos analíticos em termos de desempenho e disponibilidade. Redes de Petri estocásticas (SPN) (MARSAN, 1995) (TRIVEDI, 2002) são modelos analíticos capazes de representar concorrência, sincronização e paralelismo de sistemas complexos. Entre outras métricas, a SPN é adequado para avaliar o desempenho e a disponibilidade (SILVA et al., 2015; SANTOS et al., 2020; FERREIRA et al., 2019).

A utilização de SPNs já foi aplicada com sucesso no âmbito da MCC em trabalhos anteriores (SILVA et al., 2018; PINHEIRO et al., 2018b; PINHEIRO et al., 2018a). No entanto, as SPNs que representam as arquiteturas MEC são escassas até o momento. Este capítulo apresenta modelos SPN para avaliar a arquitetura MEC com diferença significativa com a literatura, incluindo análise de sensibilidade e métricas únicas, como probabilidade de descarte de solicitações e nível de utilização de recursos. Os modelos propostos permitem avaliar o trade-off entre o tempo médio de resposta (MRT) e a utilização de recursos, além da disponibilidade e do tempo de inatividade do sistema.

20.1.1 Objetivos

Este capítulo tem como objetivo geral desenvolver e expor resultados de modelos analíticos com redes de Petri que permitam avaliar o desempenho e a disponibilidade de um sistema de *offloading* de dados para a MEC. Os objetivos específicos deste trabalho são:

- Aplicar conhecimentos de modelagem de sistemas para gerar um modelo que permita avaliar a performance do sistema proposto com transições que permitam diferentes configurações conforme as necessidades do usuário;

- Apresentar uma validação estatisticamente comprovada a fim de garantir a fidelidade do modelo com o mundo real;
- Aplicar conhecimentos de modelagem de sistemas para gerar um modelo que permita avaliar a disponibilidade do sistema com diferentes tempos de recuperação e falha;
- Analisar a sensibilidade dos componentes usando tempos de recuperação e falha realísticos, e descobrir qual deles tem mais importância para o sistema.
- Melhorar a disponibilidade do modelo utilizando os componentes mais críticos detectados na análise de sensibilidade, através de redundância no dado componente.

20.2 Referencial Teórico

Nesta seção serão apresentados conceitos que serão utilizados nesse capítulo.

20.2.1 Redes de Petri Estocásticas

As Redes de Petri (do inglês *Petri Nets - PNs*) são ferramentas de modelagem gráfica e matemática usadas para representar diferentes tipos de sistemas, caracterizados por simultaneidade, assincronia, distribuição, paralelismo, não determinismo, e/ou processos estocásticos (CHEN; HA, 2018). As SPNs (Redes de Petri Estocásticas) são uma das classes de PNs mais comumente usadas. SPNs são PNs temporizadas, significando que a noção de tempo é incorporada na semântica da rede. No caso de SPNs, o atraso de disparo está associado às transições, e as transições são disparadas com atraso de tempo determinado a partir de uma distribuição aleatória definida, como a distribuição exponencial (XIAOLI et al., 2007).

As SPNs podem ser identificadas como um tipo de grafo direcionado dividido em duas partes, preenchido por três tipos de objetos. Esses objetos são lugares, transições e arcos direcionados que conectam os lugares para as transições e transições aos lugares. A Figura 151(a) ilustra dois tipos de transições (temporizadas e imediatas). A transição temporizada segue um comportamento estocástico, em conformidade com uma função de distribuição de probabilidade. A transição imediata dispara quando está ativado, sem esperar nenhum período de tempo. O círculo branco simboliza lugares. Os arcos são usados para conectar os lugares as transições. Os arcos inibidores bloqueiam ou permitem a passagem de *tokens* de um lugar para outro. E o *token* é atribuído a um determinado lugar.

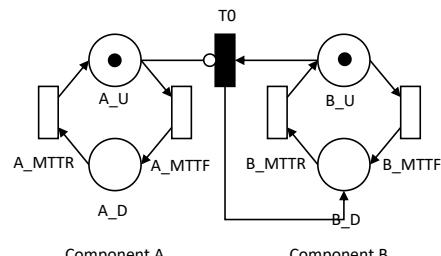
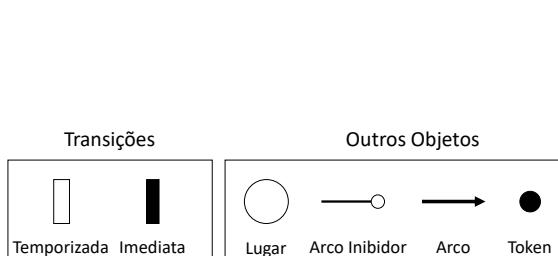


Figura 151 – Elementos básicos em modelos SPN

Em modelos SPN para avaliar disponibilidade, trabalha-se com o conceito de componente ativo ou inativo. A Figura 151(b) apresenta um pequeno modelo de disponibilidade com dois componentes (A e B). Ambos possuem tempos médios de falha (MTTF) e de reparo (MTTR). O componente A, por exemplo, está ativo quando possui *token* no lugar A_U e inativo quando possui *token* no lugar A_D. Nesse exemplo, para o componente B estar ativo, A também deve estar ativo. O arco inibidor garante que caso o componente A passe do estado *up* para o estado *down*, a transição T0 será disparada e o componente B também ficará em estado *down*.

20.2.2 Análise de Sensibilidade

A análise de sensibilidade é uma medida do efeito local de um determinado dado de entrada em relação ao dado de saída, tendo como objetivo esquematizar os elos fracos dos sistemas computacionais, e daí em diante, buscar adotar um conjunto de técnicas que visam melhorias para esses sistemas em diferentes cenários. (CAMPOLONGO; TARANTOLA; SALTELLI, 1999). De certo modo, a análise de sensibilidade pode trazer uma segurança necessária, e encaminhar os resultados dentro da perspectiva pré-estabelecida pelos administradores do sistema. Existem muitas maneiras de executar a análise de sensibilidade. Vários métodos estão à disposição, como: análise de regressão, análise de perturbação (PA), variação um por um, simulação de Monte Carlo, análise diferencial paramétrica, análise de correlação e diferença percentual. A distinção de qual método utilizar é uma etapa difícil, se fazendo necessário estabelecer quais recursos computacionais são disponíveis, e as características dos problemas abordados (CAMPOLONGO et al., 2004) (PIANOSI et al., 2016).

O método de diferença percentual foi escolhido para realizar a análise de sensibilidade deste trabalho, sendo uma abordagem que não utiliza um domínio contínuo para os valores de entrada de parâmetros. Este processo se baseia no cálculo da diferença percentual, ao variar um parâmetro de entrada de valor mínimo para o seu valor máximo. É necessário a utilização de todo o intervalo de valores possíveis de cada parâmetro para calcular as sensibilidades dos parâmetros (HOFFMAN; GARDNER, 1983).

A Equação 20.1 mostra como o índice de sensibilidade é calculado pelo método de diferença percentual. A expressão $\max \{Y(\theta)\}$ e $\min \{Y(\theta)\}$ são, respectivamente, os valores máximo e mínimo de saída, medidos ao variar o parâmetro θ sobre um intervalo de sua n possíveis valores de interesse. Se $Y(\theta)$ é conhecido por variar monotonicamente, somente os valores extremos de θ (i.e., θ_1 e θ_n) podem ser usados para calcular $\max \{Y(\theta)\}$, $\min \{Y(\theta)\}$ e como resultado $S_\theta\{Y\}$ (JÚNIOR, 2016).

$$S_\theta\{Y\} = \frac{\max \{Y(\theta)\} - \min \{Y(\theta)\}}{\max \{Y(\theta)\}} \quad (20.1)$$

20.3 Trabalhos Relacionados

Esta seção apresenta alguns trabalhos relacionados. A tabela 135 resume todos os artigos comparados. Quatorze artigos foram encontrados usando as palavras-chave que correspondem a esta proposta. Os artigos foram divididos levando-se em consideração cinco aspectos: Métricas, Variação da capacidade do servidor mestre-escravo, Análise de sensibilidade, Contexto e Uso de dependência de componente.

Métricas - As métricas ajudam a computar e, posteriormente, entender o comportamento de um sistema. Este capítulo usa métricas de desempenho e disponibilidade que não foram exploradas em conjunto por nenhum outro artigo anterior. Dentre os trabalhos que tratam de performance na borda, apenas este trabalho utilizou as métricas MRT, utilização de recursos e probabilidade de descarte. O trabalho de (COSTA et al., 2016), (MATOS et al., 2015), (DANTAS et al., 2012), (ARAUJO et al., 2014a), (OLIVEIRA et al., 2013), (SANTOS et al., 2018) e (YIN et al., 2020) focou exclusivamente na disponibilidade do sistema mas como veremos mais a frente, apenas o trabalho de (YIN et al., 2020) foca na borda, como este trabalho.

Este trabalho apresenta uma arquitetura de borda bastante única, e por isso, outra contribuição deste trabalho é a realização de experimentos com **Variação de capacidade do servidor mestre-escravo**. Dentre as obras relacionadas, nenhuma utilizou a arquitetura aqui utilizada, que fornece um maior número de parâmetros. Este trabalho também realiza uma **Análise de sensibilidade** que permite a identificação dos componentes chave para a arquitetura e, consequentemente, para o modelo. Dentre os trabalhos encontrados, apenas os trabalhos de (MATOS et al., 2015) e (SANTOS et al., 2018) realizaram uma análise de sensibilidade. No entanto, o trabalho de (MATOS et al., 2015) foi em um contexto Cloud, enquanto o trabalho de (SANTOS et al., 2018) apresenta um contexto de borda, mas focado especificamente na área de hospitais inteligentes.

Contexto - todos os artigos estão no contexto de computação remota, seja na nuvem, na borda ou na fog. A maioria dos trabalhos foca na arquitetura de borda, porém a maioria deles tem como objetivo medir o consumo de energia da borda. Diferentemente desses trabalhos, nosso trabalho usa métricas que não

Tabela 135 – Trabalhos relacionados

Trabalho Relacionado	Métricas	Variação de Capacidade do Servidor Mestre-Escravo	Análise de Sensibilidade	Contexto	Uso de Dependência de Componente
(KE; S; L, 2016)	Consumo de energia e transferência de dados	Não	Não	Edge	Não
(TRINH; YAO, 2017)	Consumo de energia	Não	Não	Edge	Não
(MAO; ZHANG; LETAIEF, 2016)	Consumo de energia	Não	Não	Edge	Não
(JARARWEH et al., 2016a)	Consumo de energia e Tempo de execução	Não	Não	Cloud/ Edge	Não
(BADRI et al., 2017)	Custo de comunicação e execução de realocação	Não	Não	Edge	Não
(ZAKARYA et al., 2020)	Consumo de energia, Tempo de execução, Custo de migração, Custo de usuário e Co-localização	Não	Não	Edge	Não
(GUILLÉN et al., 2021)	Consumo de energia e Tempo de execução	Não	Não	Edge	Não
(COSTA et al., 2016)	Disponibilidade	Não	Não	Cloud	Não
(MATOS et al., 2015)	Disponibilidade	Não	Sim	Cloud	Não
(DANTAS et al., 2012)	Disponibilidade	Não	Não	Cloud	Não
(ARAUJO et al., 2014a)	Disponibilidade	Não	Não	Cloud	Não
(OLIVEIRA et al., 2013)	Disponibilidade	Não	Não	Cloud	Não
(SANTOS et al., 2018)	Disponibilidade	Não	Sim	Cloud/ Edge/ Fog	Não
(YIN et al., 2020)	Disponibilidade	Não	Não	Edge	Não
Este trabalho	MRT, Utilização de recursos, Probabilidade de descarte e Disponibilidade	Sim	Sim	Edge	Sim

foram exploradas neste contexto até o momento. O trabalho de (YIN et al., 2020), é o único relacionado à disponibilidade que foca exclusivamente na borda.

Finalmente, **uso de dependência de componentes** é uma contribuição única. A técnica ainda é pouco explorada, mas é fundamental para garantir o correto funcionamento do sistema como um todo. A técnica é aplicada a modelos de rede de Petri para garantir que, se um componente que é a base de outros componentes cair, ocorra um efeito cascata para desligar os componentes que dependem dele. Assim, pode-se garantir que um modelo de disponibilidade será o mais próximo possível da realidade, e somente este trabalho fez uso desse aspecto das redes de Petri na modelagem da disponibilidade do sistema.

20.4 Visão Geral da Arquitetura

A Figura 152 ilustra a arquitetura MEC proposta que foi considerada para construir o modelo SPN. Essa arquitetura de base é amplamente adotada para descrever a infraestrutura MEC em vários trabalhos (TRINH; YAO, 2017; YUANZHE; SHANGGUANG, 2018; LI; WU, 2018) com um servidor MEC para processar fluxos de dados de entrada. Esses dados são gerados e enviados por aplicativos em execução em dispositivos móveis, por exemplo, aplicativos de monitoramento da saúde do usuário, renderização de jogos, etc. No contexto deste trabalho, nos concentraremos em aplicativos com alto nível de interatividade do usuário, incluindo periféricos como *smartphones* ou *tablets*.

Mais detalhadamente, na camada de computação de borda, temos dois servidores, um servidor mestre e um servidor com os nós escravos (que fará o processamento em si). Os escravos são microsserviços executados em contêineres. Neste trabalho, cada contêiner é configurado para ser executado em um núcleo de servidor. Portanto, se houver 16 núcleos, 16 contêineres serão executados. O uso de contêineres no contexto do MEC ainda não é muito comum. O uso de contêineres permite maior flexibilidade para dimensionar o poder computacional da arquitetura de acordo com o volume de tarefas e também restrições de tempos de resposta das aplicações.

O servidor mestre é responsável por receber solicitações de dispositivos móveis e distribuí-las entre

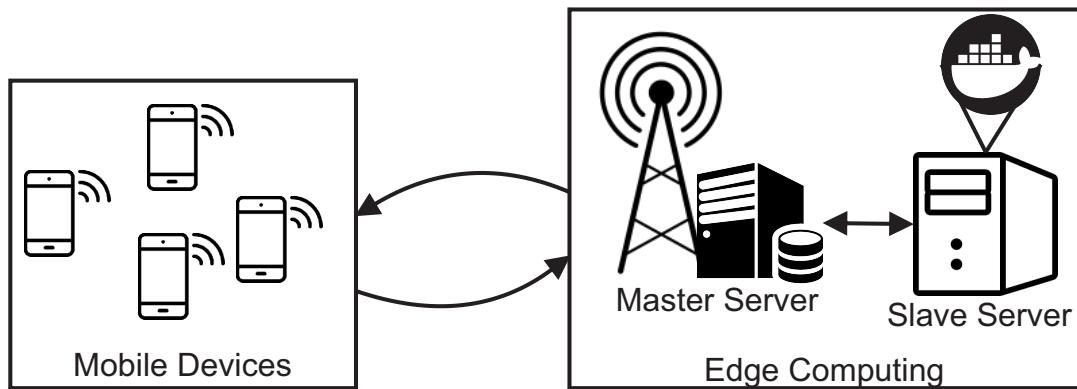


Figura 152 – Arquitetura MEC base adotada para Avaliação de Desempenho.

os servidores escravos. A princípio, o servidor mestre executa o serviço de gerenciamento com o serviço em modo *bare metal* (não virtualizado) e usando *threads*. No entanto, nada impede que o analista virtualize também o serviço do servidor mestre. Para gerenciar os contêineres, consideramos que um orquestrador de contêineres (por exemplo, Kubernetes¹ ou Swarm²) deve ser usado para aumentar a confiabilidade e elasticidade dos recursos.

Como ilustra a Figura 153, existem três blocos no fluxo de comunicação da arquitetura: (i) *Arrival Delay* (AD): intervalo de tempo entre a chegada das solicitações; (ii) *Distribution Delay* (DD): intervalo de tempo para distribuição de jobs; e (iii) *Processing Delay* (PD): tempo de processamento das solicitações pelos nós.

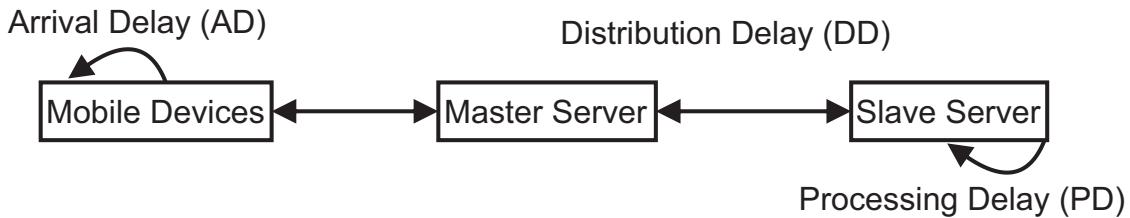


Figura 153 – Intervalos de tempo de comunicação para os componentes de arquitetura.

20.5 Avaliação de Desempenho da MEC

Esta seção apresenta dois modelos de avaliação de desempenho da arquitetura MEC. Além dos modelos, serão apresentados seus respectivos estudos de caso, métricas e validação.

Todas as avaliações, incluindo modelos de desempenho e disponibilidade, foram resolvidas por análise numérica. Normalmente, a análise numérica é preferencial ao invés da simulação porque oferece uma maior precisão nos resultados (REISIG, 2013). Portanto, o avaliador deve tentar primeiro a possibilidade de usar a análise numérica, mas às vezes isso não é possível. Redes de Petri e Cadeias de Markov podem apresentar o problema conhecido como “explosão de espaço de estado” se o modelo for muito grande. Em nosso caso, felizmente, o modelo pode ser resolvido por análise numérica.

¹ Kubernetes: <https://kubernetes.io/>

² Swarm: <https://docs.docker.com/engine/swarm/>

20.5.1 Modelo SPN básico para arquiteturas MEC

Nesta seção, descrevemos nosso modelo SPN para representar a arquitetura que integra módulos na borda da rede, apresentada na seção anterior. Ressaltamos que o objetivo do nosso modelo é possibilitar a avaliação do desempenho do sistema, antes mesmo de sua implantação. A Figura 154 apresenta nosso modelo SPN, composto de duas partes macro:

1. *Admission* que lida com a geração de solicitações;
2. *Edge* composto pelo servidor mestre e os servidores escravos; O servidor mestre recebe os dados e os distribui entre os escravos, que, em última instância, retornam os resultados aos clientes.

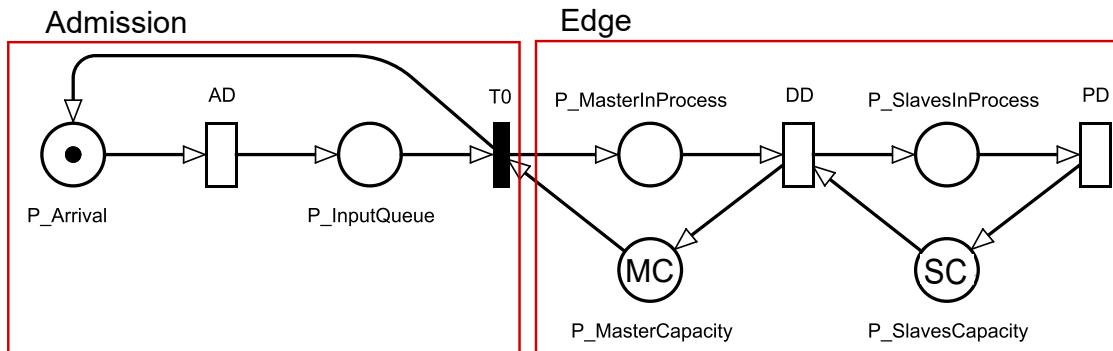


Figura 154 – Modelo SPN para uma arquitetura de Edge Computing.

Os componentes são representados por grafos como lugares (círculos), transições temporizadas (barras vazias) e marcações de lugares (círculos preenchidos). A Tabela 136 apresenta extensivamente todos os elementos do modelo. As transições temporizadas são parametrizadas com distribuições de probabilidade. O administrador do sistema deve relatar essas distribuições de acordo com a literatura ou realizar medições e caracterizações do sistema.

Tabela 136 – Descrição dos elementos do modelo.

Tipo	Elemento	Descrição
Lugar	P_Arrival	Esperando por novos pedidos
	P_InputQueue	Aguardando a disponibilidade da fila
	P_MasterInProcess P_MasterCapacity	Solicitações na fila do servidor mestre Capacidade do servidor mestre
Transições Temporizadas	P_SlavesInProcess P_SlavesCapacity	Pedidos na fila dos escravos Capacidade dos servidores escravos
	AD	Atraso de chegada entre requisições
	DD	Tempo para o servidor mestre distribuir os pedidos entre os nós escravos
Marcações	PD	Tempo gasto no processamento da solicitação por um nó escravo
	MC SC	Capacidade máxima do servidor mestre Capacidade máxima do servidor escravo

Dada a visão geral do modelo, agora descreveremos o fluxo de processamento de dados entre seus componentes. A sub-rede *Admission* consiste em dois locais P_Arrival e P_InputQueue, que representam a espera entre gerações de solicitações e a aceitação dessas solicitações na fila, respectivamente. Os tokens em P_Arrival e P_InputQueue representam qualquer tipo de solicitação que envolva entrada de dados. O tempo entre a chegada das solicitações é atribuído à transição AD. Consideraremos que os tempos

entre chegadas são distribuídos exponencialmente; esta suposição pode ser modificada alterando esta distribuição. A transição AD não leva em consideração as perdas de rede.

Quando T0 dispara, a sub-rede *Edge* é alcançada. O número de tokens em P_MasterInProcess representa o enfileiramento de solicitações na borda. A marcação MC em P_MasterCapacity indica a quantidade de espaço de armazenamento temporário do servidor mestre, enfileirando as solicitações. O enfileiramento ocorre quando não há capacidade disponível para atender à solicitação recém-chegada. Assim, se houver espaço de armazenamento suficiente, um token é puxado de P_InputQueue e P_-MasterCapacity, alocando um token para P_MasterInProcess. Quando isso acontece, o local P_-Arrival é então habilitado, permitindo uma nova chegada.

O disparo de DD representa o início da distribuição de requisições aos escravos. Estes disparos estão condicionados à quantidade de nós disponíveis para processamento em P_SlavesCapacity (com a marca SC). A marcação SC indica o número de nós disponíveis na borda da rede. Como as requisições estão sendo consumidas pelo local P_SlavesInProcess, os tokens saem de P_SlavesCapacity. Esse fluxo significa que cada solicitação será alocada em um recurso assim que chegar.

PD representa o tempo gasto pelo nó escravo para processar uma solicitação. Quando PD é disparado, um token é retirado de P_SlavesInProcess e um token é retornado para P_SlavesCapacity. A transição AD tem uma distribuição exponencial, uma vez que estamos considerando taxas de chegada distribuídas exponencialmente. Todas as outras transições têm a semântica *infinite server*, então cada trabalho é processado independentemente. É importante observar que o tempo de processamento depende da capacidade computacional de cada nó. Neste trabalho, consideramos que todos os nós de cada camada possuem a mesma capacidade computacional.

O modelo proposto permite avaliar um grande número de cenários, já que o avaliador deve configurar cinco parâmetros (ver Tabela 136). Os parâmetros incluem três transições cronometradas e os dois recursos relacionados ao local ou marcações de carga de trabalho. Qualquer mudança em um desses parâmetros pode impactar significativamente o tempo médio de resposta do sistema, o nível de descarte e o nível de utilização de recursos. A variação de cenários considerando um grande número de fatores é o que torna este modelo a principal contribuição deste trabalho.

20.5.1.1 Métricas de Desempenho

Nesta seção, definimos métricas para avaliar a arquitetura de borda com base no modelo proposto. O tempo médio de resposta (MRT) pode ser obtido considerando a lei de Little (LITTLE, 1961). A lei de Little considera o número médio de solicitações em andamento em um sistema (*RequestsInProgress*) — tamanho médio do sistema —, a taxa de chegada de novas solicitações (ARR) e o tempo médio de resposta (MRT). A taxa de chegada é o inverso do atraso de chegada — ou seja, $ARR = \frac{1}{AD}$. Um sistema estável é necessário para calcular métricas com base na lei de Little. Isso significa que a taxa de chegada deve ser menor ou igual à taxa de processamento do servidor. Presumimos que a taxa de chegada real não é necessariamente a taxa de chegada efetiva, uma vez que algumas solicitações podem ser perdidas ou descartadas devido à fila finita. Então, para obter a taxa de chegada efetiva, multiplicamos a taxa de chegada (ARR) pela probabilidade do sistema aceitar novas solicitações ($1 - Discard$) (JAIN, 1990). Portanto, a Equação 20.2 obtém MRT considerando a lei de Little e a taxa efetiva de chegada.

$$MRT = \frac{RequestsInProgress}{ARR \times (1 - Discard)} \quad (20.2)$$

A equação 20.3 obtém *RequestsInProgress*. Para calcular o número de solicitações em andamento no sistema, o analista deve somar o número esperado de tokens em cada local que representa as solicitações em andamento. Na Equação 20.3, *Esp(Place)* representa a esperança estatística de tokens no “Place”, onde $Esp(Place) = (\sum_{i=1}^n P(m(local) = i) \times i)$. Em outras palavras, *Esp(Place)* indica o número médio esperado de tokens naquele local.

$$RequestsInProgress = Esp(P_MasterInProcess) + Esp(P_SlavesInProcess) \quad (20.3)$$

A equação 20.4 define *Discard*. Deve haver um token na fila de entrada (*P_ArrivalQueue*) e não deve haver mais recursos disponíveis para processar novas solicitações nos nós mestre e escravo. $P(Place = n)$ calcula a probabilidade de n tokens naquele “Place”.

$$\text{Discard} = P(P_{\text{InputQueue}} = 1) \wedge (P_{\text{MasterCapacity}} = 0) \wedge (P_{\text{SlavesCapacity}} = 0) \quad (20.4)$$

Finalmente, além de *MRT*, também calculamos a utilização de recursos. A equação 20.5 nos dá a utilização do nó mestre. A equação 20.6 nos dá a utilização dos nós escravos. A utilização é obtida dividindo o número de tokens do local correspondente pela capacidade dos recursos totais.

$$U_{\text{Master}} = \frac{\text{Esp}(P_{\text{MasterInProcess}})}{MC} \quad (20.5)$$

$$U_{\text{Slaves}} = \frac{\text{Esp}(P_{\text{SlavesInProcess}})}{SC} \quad (20.6)$$

20.5.1.2 Análise Numérica

Esta seção apresenta duas análises numéricas para avaliações para o *MRT*, descarte e utilização. Em (PREMSANKAR; TALEB, 2018), os autores avaliaram uma arquitetura MEC com um único dispositivo móvel como cliente e contêineres executando os serviços. Os autores avaliaram um jogo 3D chamado Neverball, onde o jogador deve inclinar o chão para controlar a bola para coletar as moedas e chegar a um ponto de saída antes que o tempo acabe. Consideramos os parâmetros do sistema em (PREMSANKAR; TALEB, 2018) como parâmetros de entrada para nosso modelo. Portanto, nosso estudo evolui o trabalho em (PREMSANKAR; TALEB, 2018) realizando análises numéricas para avaliar os cenários considerando múltiplos parâmetros. Nós consideramos um de seus cenários com uma resolução de jogo de 800x600 pixels. O valor do parâmetro adotado correspondente ao *processing delay* (PD) de uma solicitação é de 24ms. Adotamos 5ms como tempo de distribuição das solicitações no sistema (transição DD). Estabelecemos que o servidor mestre possui uma restrição quanto ao número máximo de requisições que podem ser processadas simultaneamente. Esse número corresponde a 40 solicitações - ou seja, $MC = 40$.

O modelo permite uma ampla variedade de parametrizações. Na presente análise, variamos dois parâmetros: o intervalo de tempo entre as chegadas de requisições (AD) e a capacidade de recursos do servidor com nós escravos (SC). O valor de AD variou entre 1ms e 10ms, considerando um tamanho de passo de 0,5ms. A variável SC foi configurada com três possibilidades (8, 16 e 32), correspondendo ao número de núcleos em um servidor. Todos esses parâmetros podem ser variados de outras maneiras. Por exemplo, o número de escravos poderia não ser vinculado ao número de núcleos — dessa forma, o SC poderia armazenar milhares de tokens. Adotando os parâmetros mencionados acima, apresentamos os resultados considerando *MRT*, descarte e utilização do servidor mestre e servidores escravos.

A Figura 155 mostra os resultados para o tempo médio de resposta. A princípio, espera-se que quanto maior o intervalo de tempo entre chegadas (AD), menor será o *MRT*. O sistema será mais capaz de lidar com as solicitações de entrada com os recursos de processamento disponíveis. Também é esperado que quanto maior a capacidade do escravo (SC), menor o *MRT*, pois mais capacidade de processamento está disponível para atender às solicitações. Esses dois comportamentos são facilmente observados quando os valores dos descartes são mínimos no modelo (o que pode ser observado na Figura 158). Para descartes mínimos, o *MRT* diminui até o tempo mínimo para realizar solicitações sem espera na fila.

Porém, quando o sistema descarta as solicitações de entrada, observamos o aumento do *MRT* até um pico, diminuindo a partir daí. Este comportamento se deve ao valor limitador dos recursos do sistema, onde algumas das solicitações recebidas são descartadas quando não há mais recursos disponíveis para processá-las. Assim, limitando a variação do *MRT* ao tempo entre as chegadas. Como pode ser deduzido da lei de Little (JAIN, 1990), o tempo médio entre as saídas aumentará junto com o tempo médio entre as chegadas até atingir o pico. Em nossos resultados numéricos, os picos foram: para $SC = 8$ foi em $AD = 1,5$ ms e para $SC = 16$ foi $AD = 3,0$ ms. Nesses pontos, a quantidade de trabalho dentro do sistema começa a reduzir, reduzindo o *MRT* mesmo quando o AD aumenta drasticamente. É importante ressaltar que no *MRT* consideramos a taxa efetiva de chegada, ou seja, ajustamos seu valor considerando a probabilidade de descarte.

O MRT para $SC = 32$ é baixo, mesmo para um intervalo de tempo entre chegadas de 1 ms. Comparando $SC = 16$ e $SC = 32$, e considerando um atraso de chegada de 2,5 ms, os MRTs se equalizam. Considerando um atraso de chegada de 5,5ms, e o $SC = 8$, também apresenta o mesmo resultado médio. Portanto, se o contexto real tivesse um $AD = 5,5$ ms, um servidor de 8 núcleos teria o mesmo desempenho que servidores mais poderosos. Portanto, nosso trabalho pode auxiliar os gestores na tarefa de escolher os servidores e identificar o melhor desempenho e custos, considerando a carga de trabalho esperada.

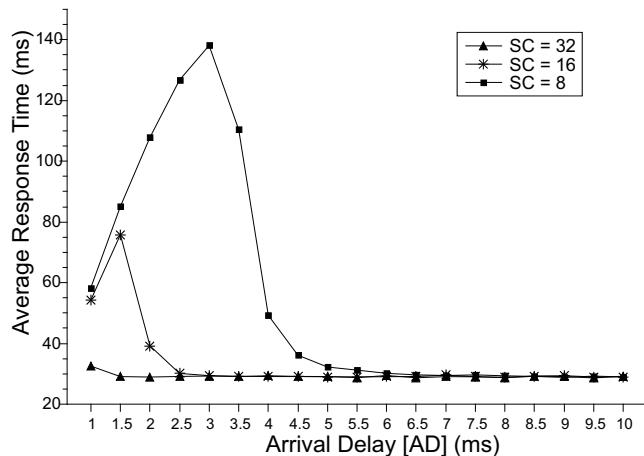


Figura 155 – Mean Response Time (MRT).

A Figura 156 mostra o nível de utilização do servidor mestre. O servidor mestre é o primeiro componente que a solicitação atinge ao entrar na camada MEC. Para as configurações $SC = 8$ e $SC = 16$, o nível de utilização é em torno de 100% nos menores valores de AD; depois disso, a utilização cai. Para $SC = 32$, mesmo com $AD = 1,0$ ms, o valor de utilização atinge apenas 20%. Com $AD = 5,5$ ms, as três configurações têm valores semelhantes e próximos a 0%. O administrador do sistema deve considerar o desejo de níveis altos ou baixos de servidor ocioso. A Figura 157 mostra o nível de utilização dos servidores escravos. Quanto maior o número de recursos, menor o nível de utilização dos escravos. À medida que AD aumenta, o nível de utilização diminui sutilmente em todos os três casos. No entanto, esta queda só começa em $AD = 3,0$ ms para $SC = 8$ e em $AD = 1,5$ ms para $SC = 16$. Até estes pontos, o nível de utilização está em torno de 82%, o que ocasiona o comportamento do MRT explicado acima.

A Figura 158 apresenta a probabilidade de descarte de novas solicitações. Para $SC = 32$, a probabilidade de descarte é igual a 0. Portanto, se for possível adquirir um servidor com 32 núcleos, não haverá descarte independente do intervalo entre as chegadas das solicitações. Para $SC = 8$ e $SC = 16$, apenas a partir de $AD = 2,0$ ms e $AD = 4,0$ as probabilidades de descarte tendem a 0. Esses intervalos iniciais de descarte estão diretamente relacionados ao alto nível de utilização apresentado por ambos os servidores, impactando diretamente no tempo médio de resposta. Portanto, qualquer análise estocástica realizada com o modelo proposto deve observar as quatro métricas para obter uma visão completa do comportamento do sistema. Também é possível identificar os limites operacionais do sistema. Em outras palavras, esses limites representam quantos trabalhos podem ser perdidos sem comprometer a utilidade do sistema.

20.5.2 Modelo Refinado com Estado Absorvente

Os administradores de sistema que desejam usar uma arquitetura MEC devem estar cientes de quando seus aplicativos têm maior probabilidade de concluir a execução. As Funções de Distribuição Cumulativa (CDFs, do inglês Cumulative Distribution Functions) podem indicar tal momento por meio da probabilidade máxima de absorção. As CDFs estão associados a uma distribuição de probabilidade específica. Neste trabalho, a distribuição de probabilidade está relacionada à probabilidade de finalizar a execução do aplicativo em um determinado tempo. A probabilidade é obtida por meio de avaliação transitória, gerando probabilidades

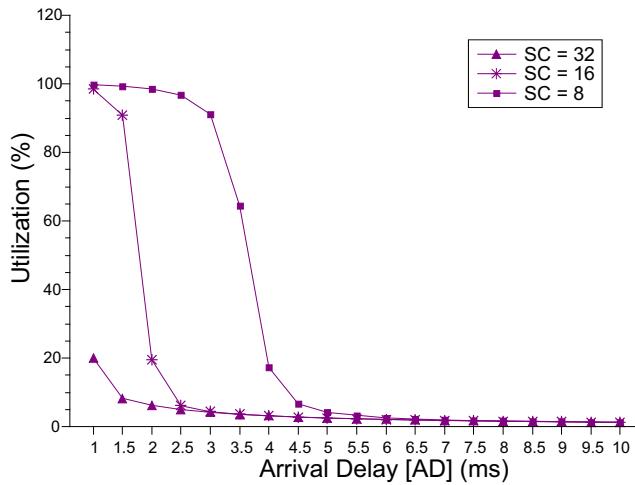


Figura 156 – Nível de utilização do servidor mestre.

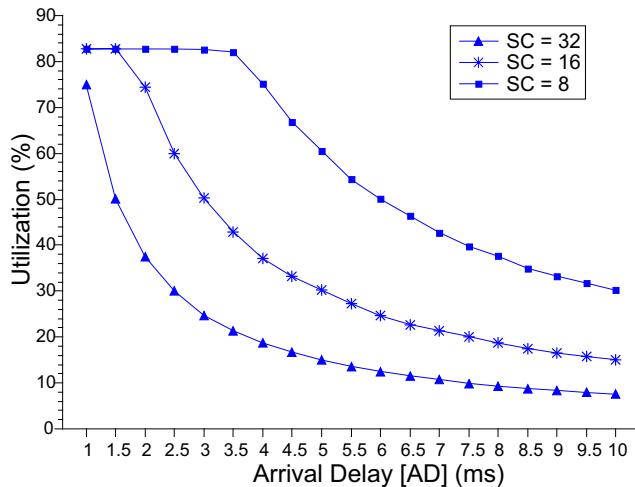


Figura 157 – Nível de utilização do servidor escravo.

com o tempo tendendo a um valor t . Em outras palavras, os desenvolvedores calculam a probabilidade de absorção em $[0, t]$, por meio da avaliação transitória, onde $F(t)$ se aproxima de 1.

As CDFs indicam a probabilidade máxima de o processamento de um aplicativo ser concluído dentro de um determinado intervalo de tempo. Neste trabalho, o estado absorvente é alcançado quando o modelo está no estado FINISH. Para uma melhor compreensão das métricas dependentes do tempo, é necessário definir a diferença entre o estado transitório e o absorvente. Os estados transitórios são definidos como estados temporários. Em outras palavras, quando o sistema sai de um estado transitório, existe a probabilidade de nunca mais voltar a ele. Por outro lado, um estado absorvente é um estado em que, quando o sistema o atinge, não há saída. A Figura 159 mostra a adaptação que fizemos em nosso modelo SPN apresentado anteriormente para calcular CDFs.

Três alterações foram feitas: (a) um local de estado absorvente (denominado *Finish*) foi adicionado na parte direita do modelo indicando que quando as solicitações chegam a este local, tais solicitações não mudarão de estado; (b) no bloco de Admissão, o loop de feedback ($T0 \rightarrow P_{Arrival}$) foi retirado, indicando que novas solicitações não serão geradas de forma inequívoca; e (c) há um novo parâmetro chamado BATCH

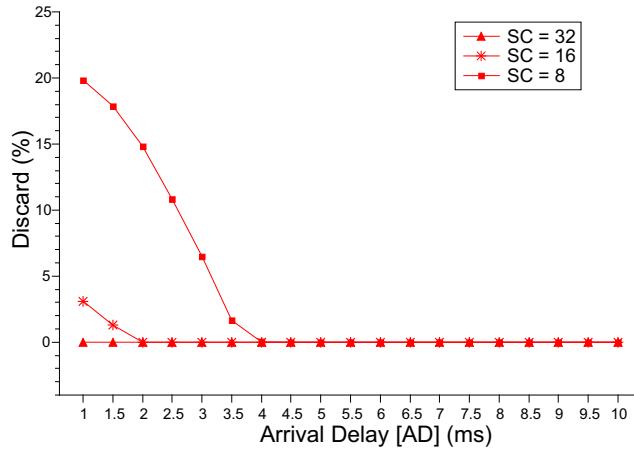


Figura 158 – Probabilidade de descarte.

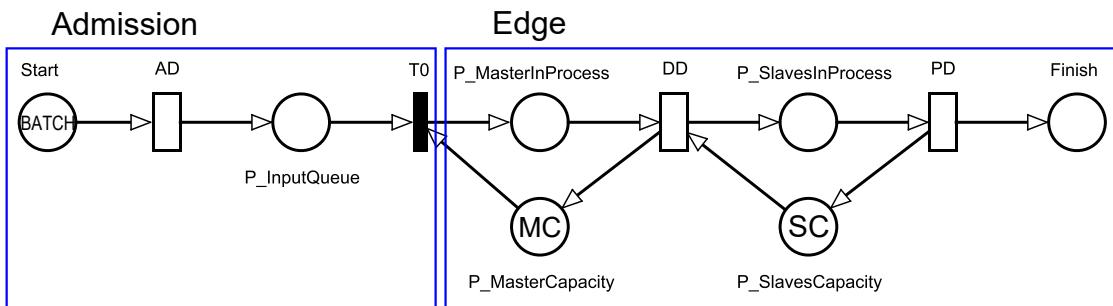


Figura 159 – Modelo SPN usando estado de absorção.

(no local P_Arrival, agora denominado *Start*) que representa o número de *jobs* (tokens) que serão processados. A CDF calcula a probabilidade desses *jobs* concluírem o processamento do aplicativo em um determinado momento e em um intervalo de tempo específico.

20.5.2.1 Estudo de caso 1

Para este estudo, definimos a capacidade do servidor mestre (MASTERC) em 40, o tempo entre chegadas (AD) em 5 ms, e criamos 3 cenários variando a capacidade do servidor com os nós escravos (SLAVEC) em 4, 8 e 16. Estes cenários foram definidos com o objetivo de verificar qual configuração do servidor escravo melhor atende aos requisitos de um administrador de infraestrutura, de acordo com o tempo total desejado para a execução da aplicação. A Tabela 137 permite uma melhor visualização dessas variáveis.

Tabela 137 – Cenários possíveis para o estudo de caso 1.

Cenário	MASTERC	AD	SLAVEC
#1	40	5	4
#2	40	5	8
#3	40	5	16

A Figura 160 mostra os resultados obtidos para a CDF. Em geral, o cenário #1 é o que leva mais tempo para executar o aplicativo. Observe que os cenários #2 e #3 são os melhores casos, e com níveis de desempenho

próximos um do outro. Apesar de algumas ocorrências em que o cenário #3 se sai melhor do que #2, quando a probabilidade de execução termina perto de 1, os tempos se equilibram. Também podemos ver que ambos os cenários possuem um tempo de execução em que a probabilidade aumenta intensamente. Supondo que um administrador de infraestrutura deseja que seu aplicativo seja concluído em 800 ms, isso pode ser feito usando os cenários #2 e #3, e uma escolha pode ser feita com base na disponibilidade de recursos.

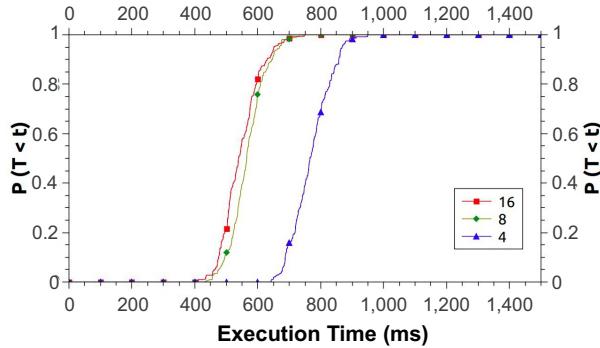


Figura 160 – Funções de distribuição cumulativa para o estudo 1.

20.5.2.2 Estudo de caso 2

Para este estudo, definimos a capacidade do servidor mestre (MASTERC) em 40, a capacidade do servidor escravo em 16 nós, e criamos alguns cenários variando de tempo de chegada (AD) com valores entre 1 ms a 10 ms, com incrementos de 1 ms. A Tabela 138 apresenta a combinação dessas variáveis.

Tabela 138 – Cenários possíveis para o estudo de caso 2.

Cenário	MASTERC	AD	SLAVEC
#1	40	1	16
#2	40	2	16
#3	40	3	16
#4	40	4	16
#5	40	5	16
#6	40	6	16
#7	40	7	16
#8	40	8	16
#9	40	9	16
#10	40	10	16

A Figura 161 apresenta os resultados obtidos para a métrica CDF. O tempo de execução do aplicativo aumenta com o aumento do AD. Neste estudo, temos um lote de 100 solicitações. Cada uma dessas solicitações entra no modelo de acordo com o intervalo definido no AD. Assim, para AD igual a 1ms, sabemos que o tempo mínimo de execução da aplicação é 100ms, enquanto para AD igual a 10ms, o tempo mínimo de execução é 1000ms. Também podemos ver que ambos os cenários possuem um tempo de execução onde a probabilidade aumenta intensamente; no entanto, este aspecto crescente diminui ligeiramente à medida que o AD cresce. Supondo que um administrador de infraestrutura deseja que seu aplicativo seja executado em 700 ms, o modelo garante que isso possa ser alcançado com os cenários #1, #2, #3 ou #4.

20.5.2.3 Validação do Modelo

Esta seção apresenta a validação do modelo SPN refinado proposto. O objetivo desta validação foi comparar o MRT calculado pelo modelo e o MRT coletado por experimentos em um cenário real. Um *test-bed*

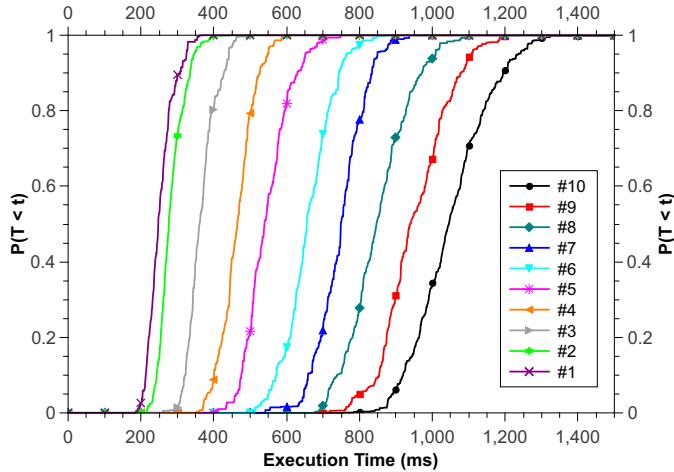


Figura 161 – Funções de distribuição cumulativa para o estudo 2.

foi implementado em um laboratório com Internet de 40 Mbps, conforme mostrado na Figura 162. Um computador — Intel Core i7 2.4 Ghz 8GB — agia como um gerador de solicitação sintética.

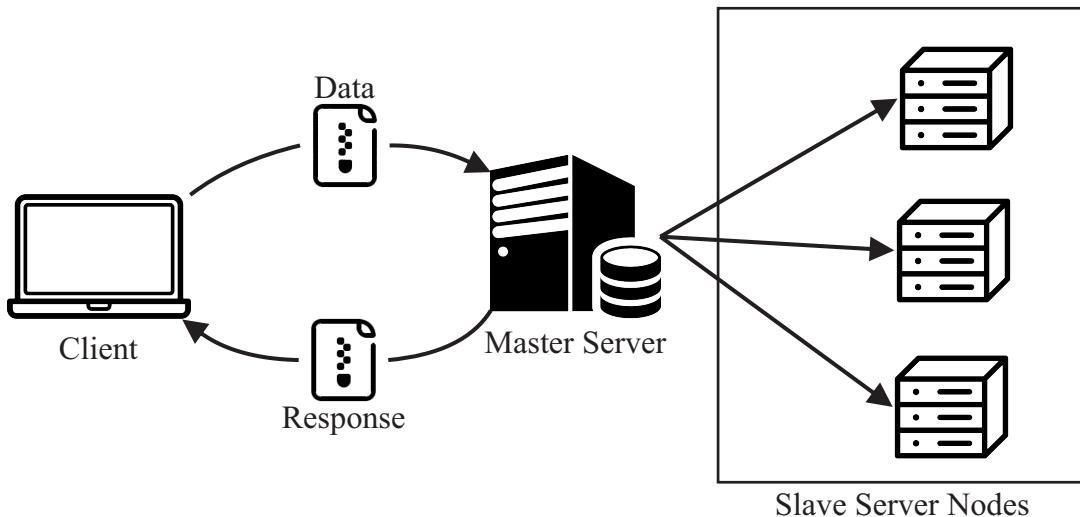


Figura 162 – Arquitetura *test-bed* de validação.

Para a validação, adotamos um algoritmo de processamento de texto. O algoritmo é o algoritmo clássico de contagem de palavras³, viral no contexto *Big Data Map-Reduce*. Este algoritmo contém o número de ocorrências exclusivas de palavras-chave em um arquivo de texto. O sistema organiza blocos de dados, visando dividir o texto. O número de etapas de mapeamento de *jobs* é determinado pelo número de divisões. Em seguida, o sistema distribui tarefas entre os nós escravos. Esta etapa gera um par de valores-chave, sendo a chave uma palavra e o valor a quantia 1. Observe que apenas o resultado do mapeamento ainda não é a ocorrência total das palavras. A etapa de classificação gera uma lista de todos os valores por chave e a etapa de redução processa essa lista para obter o resultado final. A etapa de redução gera um arquivo com uma frequência de cada palavra. Em nossa implementação, cada nó executa o algoritmo de mapeamento e

³ Word Count Algorithm <https://tinyurl.com/y8hof5x>

ordenação. Por fim, é necessário coletar o tempo de processamento de 15MB de texto em um único nó e utilizar esse resultado para parametrizar o modelo.

Implantamos a borda com quatro máquinas — uma como mestre e três como nós escravos. Uma nova solicitação foi gerada a uma taxa de entrada de 230 s. A solicitação consistia em três arquivos de texto de 40 MB cada. O objetivo de cada máquina processando apenas um arquivo era atingir o nível máximo de paralelismo sem sobrecarregar o sistema, obedecendo à Lei de Little. Foi necessário coletar o tempo médio de processamento de um arquivo no nó escravo (PD) para parametrizar o modelo. Os valores estão resumidos na Tabela 139. O MRT calculado pelo modelo SPN por meio da análise estacionária foi de 86,74s. A utilização total dos recursos foi de 22,4 %.

Tabela 139 – Parâmetros do modelo SPN para validação do modelo.

Processing Delay	Arrival Delay	Master Capacity	Slave Capacity
51.7 (s)	230 (s)	20	3

A próxima etapa foi realizar solicitações à borda repetidamente. O número de solicitações foi 100. A amostra apresentou distribuição normal e média de 86,91s. O Teste-T de uma amostra⁴ é usado para fazer inferências sobre a média de uma população, com base em dados de uma amostra aleatória. Adotamos o Teste T de uma amostra para comparar o MRT gerado pelo modelo com a média da amostra. A hipótese nula é que ambas as médias são iguais. A Tabela 140 apresenta o resultado do teste estatístico.

Tabela 140 – Resultados numéricos do teste-T de uma amostra.

Sample Size	Mean	StDev	SE Mean	95% CI	T	p-value
100	86.906	5.039	0.504	(85.906; 87.905)	0.41	0.684

O valor p é superior a 0,05 e, portanto, não podemos refutar a hipótese nula com 95% de confiança. O resultado gerado pelo modelo é estatisticamente equivalente ao experimento. Concluindo, o modelo proposto é totalmente preciso. O modelo reflete o ambiente real e é útil para o planejamento de arquiteturas MEC.

20.6 Avaliação da Disponibilidade do MEC

Esta seção apresenta dois modelos SPN com foco na avaliação da disponibilidade da arquitetura MEC apresentada anteriormente. Primeiramente, apresentamos uma proposta de base e, em seguida, uma versão estendida é detalhada.

20.6.1 Proposta Base - Arquitetura

A Figura 163 mostra a arquitetura em camadas atribuída aos dois servidores. O servidor mestre é responsável por receber solicitações de dispositivos móveis e distribuí-las aos nós escravos, portanto, o componente de software denominado balanceador de carga é o responsável por tal distribuição. A política de distribuição deve ser definida pelo avaliador. O servidor escravo executa uma plataforma de virtualização, que ilustramos como sendo o Docker⁵, acima do qual estão os N contêineres. O usuário do modelo pode adotar qualquer plataforma de virtualização. Neste caso, o usuário deve considerar o respectivo Tempo médio até a falha (MTTF, do inglês Mean Time to Failure)/Tempo médio até o reparo(MTTR, do inglês Mean Time to Repair) para alimentar o modelo.

⁴ Teste-T de uma amostra <https://tinyurl.com/yanthw4e>

⁵ Docker: <https://www.docker.com/products/>

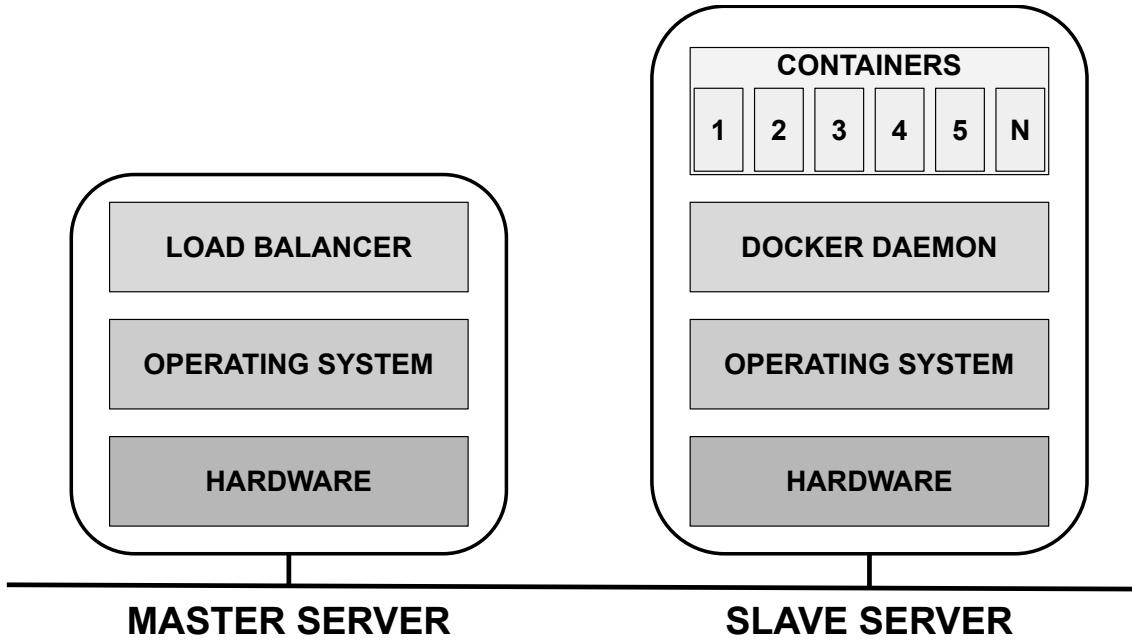


Figura 163 – Arquitetura em camadas composta por dois servidores.

20.6.1.1 Proposta Base - Modelo SPN

A Figura 164 apresenta um modelo SPN para a arquitetura MEC com as seguintes funções: (i) *Master Server* é responsável por receber solicitações de dispositivos móveis e distribuí-las entre os servidores escravos; (ii) *Slave Server* é responsável pelo processamento dos dados recebidos do servidor mestre. Os componentes do modelo correspondem às mesmas camadas apresentadas na Figura 163. Cada componente possui seus respectivos MTTF e MTTR. Ambos *master server* e *slave server* foram modelados, levando em consideração a dependência entre os componentes; ou seja, quando um componente falha, as transições imediatas farão com que os próximos componentes dependentes também falhem.

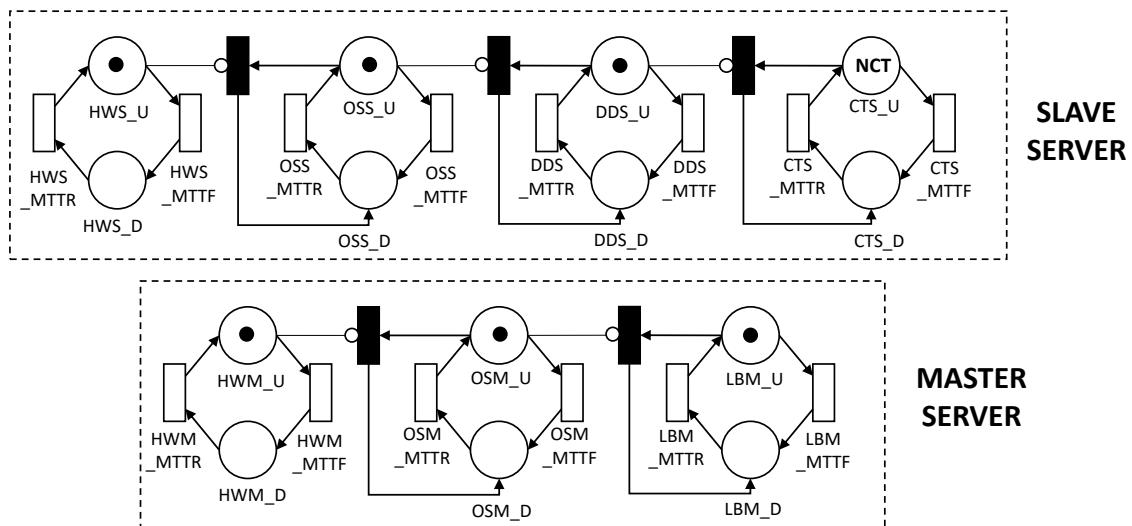


Figura 164 – Modelo básico SPN para uma arquitetura MEC com layout mestre-escravo.

A marcação NCT corresponde ao número de containers disponíveis. O *slave server* estará funcionando quando tiver NCT tokens no local CTS_U (contêiner ativo). O avaliador pode definir na métrica (com a marca NCT) quantos contêineres devem estar ativos para que o sistema funcione. Consideramos que *slave server* não está funcionando quando há um token em um dos seguintes lugares: HWS_D (hardware inativo), OSS_D (sistema operacional inativo), DDS_D (docker daemon inativo), CTS_D (contêiner inativo). A mudança entre o estado ativo e inativo é causada pelas seguintes transições: HWS_MTTF, OSS_MTTF, DDS_MTTF e CTS_MTTF — para tempo médio até a falha — e HWS_MTTR, OSS_MTTR, DDS_MTTR e CTS_MTTR — para tempo médio de reparo.

O *Master Server* estará ativo quando tiver tokens no local LBM_U (balanceador de carga ativo). Consideramos que o *master server* não está funcionando quando tem um token em um dos seguintes locais: HWM_D (hardware inativo), OSM_D (sistema operacional inativo), LBM_D (balanceador de carga inativo). A mudança entre o estado ativo e inativo é causada pelas transições: HWM_MTTF, OSM_MTTF e LBM_MTTF — para tempo médio de falha — e HWM_MTTR, OSM_MTTR e LBM_MTTR — para tempo médio de reparo.

Duas métricas foram aplicadas: disponibilidade e tempo de inatividade. A equação de disponibilidade representa a soma das probabilidades dos componentes em estado ativo. P representa probabilidade e # representa o número de tokens em um determinado lugar. O tempo de inatividade (D) pode ser obtido por $D = (1 - A) \times 8760$, onde A representa a disponibilidade do sistema e 8760 representa o número de horas no ano. Para a disponibilidade, o sistema está totalmente funcional quando todos os contêineres e o LB (balanceador de carga) estão ativos. Portanto, a disponibilidade é calculada por: $A = P\{(\#CTS_U = NCT) \text{AND} (\#LBM_U > 0)\}$.

As condições de guarda garantem que as transições sejam acionadas apenas quando uma condição específica for satisfeita. A condição de guarda garante que o modelo reflita os comportamentos do sistema no mundo real. Por exemplo, a transição OSS_MTTR tem a seguinte condição de guarda: $P\{\#HWS_U > 0\}$, o que significa que para acionar a transição de recuperação do sistema operacional, o hardware (HW) deve estar ativo. Seguindo este exemplo, as transições DDS_MTTR, CTS_MTTR, OSM_MTTR e LBM_MTTR seguem o mesmo padrão e só são ativadas se seus respectivos componentes de dependência estão ativos. Veja Tabela 141 para mais detalhes.

Tabela 141 – Condições de guarda usadas para garantir a dependência dos componentes.

Transição	Expressão
OSS_MTTR	$P\{\#HWS_U > 0\}$
DDS_MTTR	$P\{\#OSS_U > 0\}$
CTS_MTTR	$P\{\#DDS_U > 0\}$
OSM_MTTR	$P\{\#HWM_U > 0\}$
LBM_MTTR	$P\{\#OSM_U > 0\}$

20.6.2 Proposta Estendida - Arquitetura

Uma limitação significativa da proposta básica é que, se um dos dois servidores falhar, todo o sistema irá parar de funcionar. Por esse motivo, a Figura 165 apresenta uma segunda proposta de arquitetura redundante, que visa melhorar a taxa de disponibilidade do sistema em relação à arquitetura de base. Para esta arquitetura, consideramos redundância apenas no *slave server* para avaliar uma possível melhoria de disponibilidade onde ocorre o processamento dos dados. Tal decisão de redundância é baseada na análise de sensibilidade que será apresentada na seção 20.6.2.2. A análise de sensibilidade evidenciou os contêineres como os componentes mais críticos do sistema. Ambos os servidores escravos estão sempre funcionando; entretanto, o Docker e os contêineres no *slave server* 02 são instanciados apenas se o *slave server* 01 falhar. Portanto, há um mecanismo de redundância de *warm standby* (YUAN; MENG, 2011) no hardware e sistema operacional, e *cold standby* (BRIŠ, 2013) no Docker e contêineres.

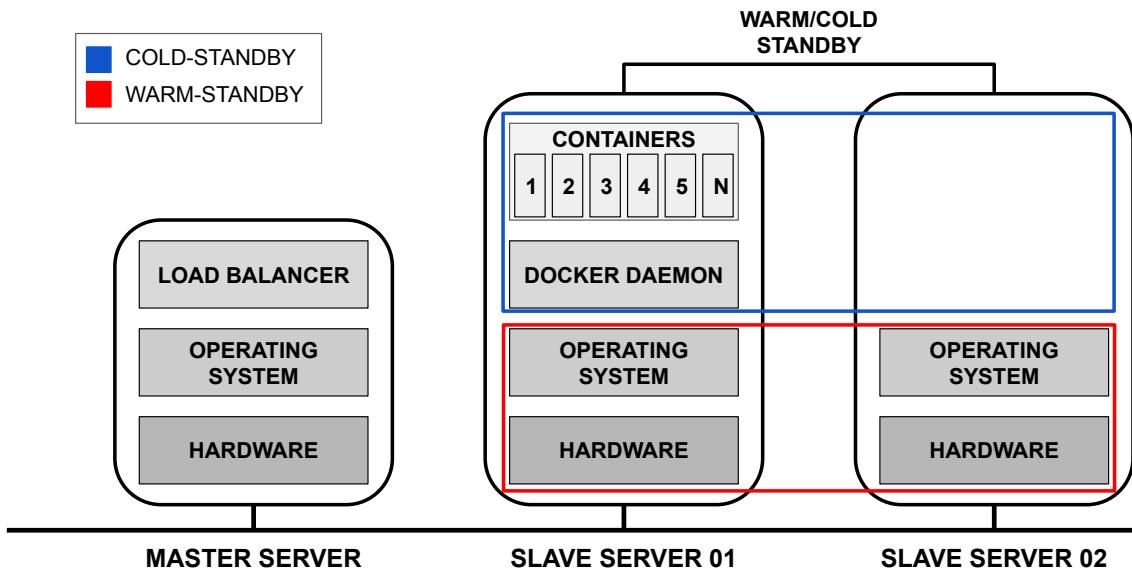


Figura 165 – Arquitetura em camadas com redundância no servidor escravo.

20.6.2.1 Proposta Estendida - Modelo SPN

A Figura 166 apresenta um modelo SPN estendido para a arquitetura MEC, com 2 *slave servers* e um *master server*. O modelo SPN estendido apresenta uma mudança no *slave server*, enquanto o *master server* mantém os mesmos componentes. Conforme mencionado anteriormente, este novo modelo apresenta um mecanismo híbrido entre *warm standby* (componentes HW e OS) e *cold standby* (componentes DD e CTS). Se algum dos componentes de software do *slave server 01* falhar, outros componentes serão iniciados no *slave server 02*. Duas novas transições foram adicionadas para satisfazer essas condições, HWSF_MTTF e OSSF_MTTF (no bloco *slave server 02*), que representam o MTTF para HW e OS quando Docker é desativado, ou seja, HW e OS estão ociosos.

Basta observar o estado das camadas superiores da arquitetura para calcular a disponibilidade devido à dependência dos componentes. Portanto, o sistema estará funcionando quando os contêineres e o balanceador de carga estiverem funcionando. O NCT é o parâmetro que representa o número máximo de contêineres que o sistema pode rodar. Em outras palavras, a soma do número de contêineres ativos nos dois servidores redundantes não pode exceder o valor de NCT. A transição SWITCH_TIME é disparada quando o *slave server 01* falha e este tempo corresponde ao tempo que leva para o Docker do *slave server 02* inicializar. Disparar SWITCH_TIME também coloca o número de contêineres correspondentes a NCT no lugar CTS2_D, ou seja, os contêineres são criados com status inativo no *slave server 02*, e levarão algum tempo para serem instanciados, correspondendo à transição CTS2_MTTR. Portanto, a disponibilidade do segundo modelo é dada por $A = P\{(\#CTS1_U + \#CTS2_U) = NCT) \text{ AND } (\#LBM_U > 0\}$. Vale ressaltar que as transições MTTF são do tipo *infinite server* (paralelo), e as transições MTTR são do tipo *single server* (simultâneo).

A Tabela 142 mostra as condições de guarda usadas para a operação do sistema no modelo estendido. Neste caso, o uso de condições de guarda evita poluição visual no modelo, uma vez que várias conexões devem ser feitas, o que tornaria o modelo difícil de entender.

20.6.2.2 Análise de Sensibilidade das Transições

Para a análise de sensibilidade, os parâmetros MTTF e MTTR foram variados em cinco valores dentro de uma faixa definida pelos valores máximo e mínimo (50% mais e menos o valor padrão). A Tabela 143

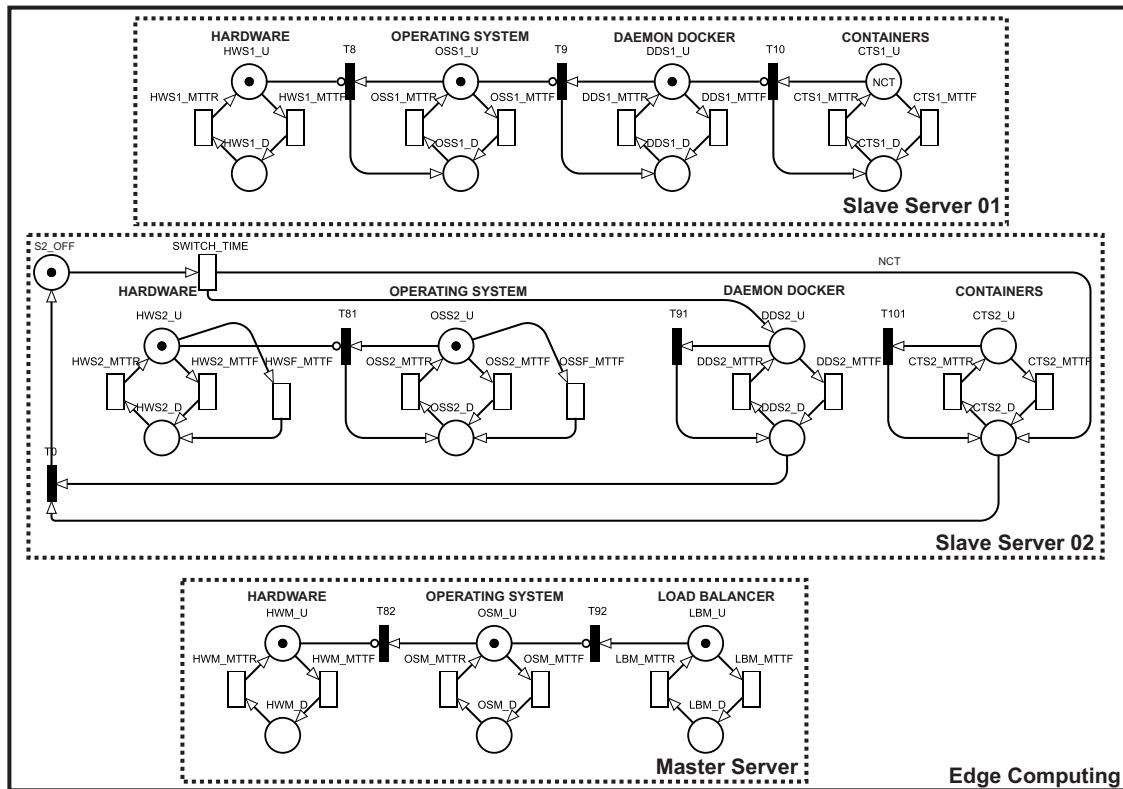


Figura 166 – Modelo SPN estendido para uma arquitetura MEC com servidor escravo redundante.

Tabela 142 – Condições de gurada para o modelo estendido.

Transição	Expressão	Descrição
OSS2_MTTR	HWS2_U >0	Ativado quando HW está funcionando.
DDS2_MTTR	OSS2_U >0	Ativado quando OS está funcionando.
CTS2_MTTR	(DDS2_U >0) AND (CTS2_U <CTS1_D)	Ativado quando Docker do slave server 02 está funcionando e o slave server 01 tem mais contêineres em falha do que contêineres funcionando no slave server 02.
HWSF_MTTF	S2_OFF=1	Ativado quando slave server 02 está ocioso.
OSSF_MTTF	S2_OFF=1	Ativado quando slave server 02 está ocioso.
HWS2_MTTF	S2_OFF=0	Ativado quando slave server 02 está funcionando.
OSS2_MTTF	S2_OFF=0	Ativado quando slave server 02 está funcionando.
T91	((#CTS1_U = NCT) OR (#OSS2_U = 0))	Ativado quando o slave server 01 está ativo ou quando o componente dependente (OS) está em falha.
T101	((#CTS1_U = NCT) OR (#DDS2_U = 0) OR (#CTS2_U >#CTS1_D))	Ativado quando slave server 01 está ativo ou o Docker falhou, ou o número de contêineres ativos no slave server 01 ficou maior que no slave server 02.
T0	#CTS1_U = NCT	Ativado quando slave server 01 está ativo.
SWITCH_TIME	#CTS1_D >0	Ativado quando slave server 01 falha.

apresenta os componentes e seus respectivos índices de sensibilidade que causam impactos significativos na disponibilidade do sistema. Os componentes mais significativos serão expostos e discutidos a seguir.

Os tempos de falha e recuperação dos contêineres foram os componentes que mais impactaram a disponibilidade do modelo básico; isso também pode ser visto na Figura 168(a), então pode-se considerar os contêineres como componentes essenciais do sistema. Os dados são processados nos contêineres. Quanto mais tempo ficarem fora de operação, mais prejudicada será a eficiência do sistema. Em seguida vêm os

Tabela 143 – Valores resultantes da análise de sensibilidade nas transições temporizadas dos modelos SPN.

Modelo Base		Modelo Estendido	
Variável	Índice	Variável	Índice
CTS_MTTR	6.330×10^{-3}	LBM_MTTF	1.898×10^{-3}
CTS_MTTF	4.047×10^{-3}	LBM_MTTR	1.894×10^{-3}
OSS_MTTF	2.409×10^{-3}	HWM_MTTF	1.519×10^{-3}
DDS_MTTF	2.152×10^{-3}	OSM_MTTF	9.506×10^{-4}
HWS_MTTF	1.985×10^{-3}	HWM_MTTR	9.120×10^{-4}
LBM_MTTF	1.898×10^{-3}	SWITCH_TIME	6.782×10^{-4}
LBM_MTTR	1.894×10^{-3}	CTS_MTTR	6.511×10^{-4}
HWM_MTTF	1.519×10^{-3}	CTS_MTTF	6.223×10^{-4}
OSM_MTTF	9.506×10^{-4}	OSM_MTTR	4.709×10^{-4}
HWM_MTTR	9.119×10^{-4}	OSS_MTTF	3.068×10^{-4}
HWS_MTTR	9.119×10^{-4}	DDS_MTTF	2.602×10^{-4}
OSS_MTTR	4.709×10^{-4}	HWS_MTTF	1.110×10^{-4}
OSM_MTTR	4.709×10^{-4}	DDS_MTTR	4.002×10^{-5}
DDS_MTTR	2.214×10^{-4}	OSS_MTTR	2.124×10^{-5}
		HWS_MTTR	4.193×10^{-6}
		HWSF_MTTF	1.790×10^{-6}
		OSSF_MTTF	3.577×10^{-7}

tempos de falha dos componentes do *slave server* (SO, DD e HW). Como esses componentes são partes que “suportam” os contêineres, seu tempo de atividade tem um impacto significativo no tempo de atividade dos contêineres. Os tempos de operação dos componentes do *master server* vêm posteriormente. Se o *master server* falhar, nenhum dado chegará para ser processado em qualquer *slave server*.

Conforme mencionado, os resultados da análise de sensibilidade no modelo base foram considerados para gerar um modelo estendido. No modelo estendido, houve uma inversão da ordem dos índices em relação à relevância do componente. Adicionando a estratégia de redundância híbrida, o índice de sensibilidade do *slave server* pode ser reduzido significativamente. Dada a redundância, quando qualquer componente do *slave server* 01 falhar, o *slave server* 02 será ativado. Os componentes do *master server* acabaram se tornando mais relevantes por não possuírem mecanismo de redundância. Os índices foram iguais à primeira análise no modelo base. Em seguida, temos SWITCH_TIME, que é essencial para definir o tempo que leva para inicializar o *slave server* 02, tornando-se uma variável relevante. As transições menos relevantes foram aquelas relacionadas à falha de hardware (HWSF_MTTF) e sistema operacional (OSSF_MTTF) em estado inativo. Essas transições têm valores muito altos. Outros componentes falharão muitas vezes antes desse tempo ser atingido.

A Figura 167 mostra com mais detalhes o impacto dos três componentes mais essenciais de cada modelo. Como pode ser visto nas Figuras 167(a), 167(b) e 167(c), a variação dos parâmetros influencia a disponibilidade até certo ponto. Todos os resultados do modelo base estão abaixo de 99% de disponibilidade, com exceção da transição CTS_MTTR (Figura 167(b)). Os resultados do modelo estendido apresentam valores muito superiores aos do modelo base (ver Figuras 167(d), 167(e) e 167(f)). Os resultados do modelo estendido, diferentemente do modelo base, ficam sempre acima de 99%, pois apresenta redundância nos componentes que estavam impactando a disponibilidade do modelo base.

20.6.3 Estudo de Caso

Esta seção apresenta um estudo de caso com uma avaliação de disponibilidade, considerando os dois modelos apresentados. Alguns parâmetros de entrada são necessários para realizar a avaliação. Os valores de MTTF e MTTR para cada componente foram extraídos de (ARAUJO et al., 2019) e (LISBOA et al., 2018). A Tabela 144 mostra os valores de entrada dos componentes do modelo. Por mais que o modelo proposto suporte configurações diferentes para cada servidor, em termos de simplificação, optamos por usar a mesma configuração entre os três servidores. O tempo para ativar o servidor redundante na transição

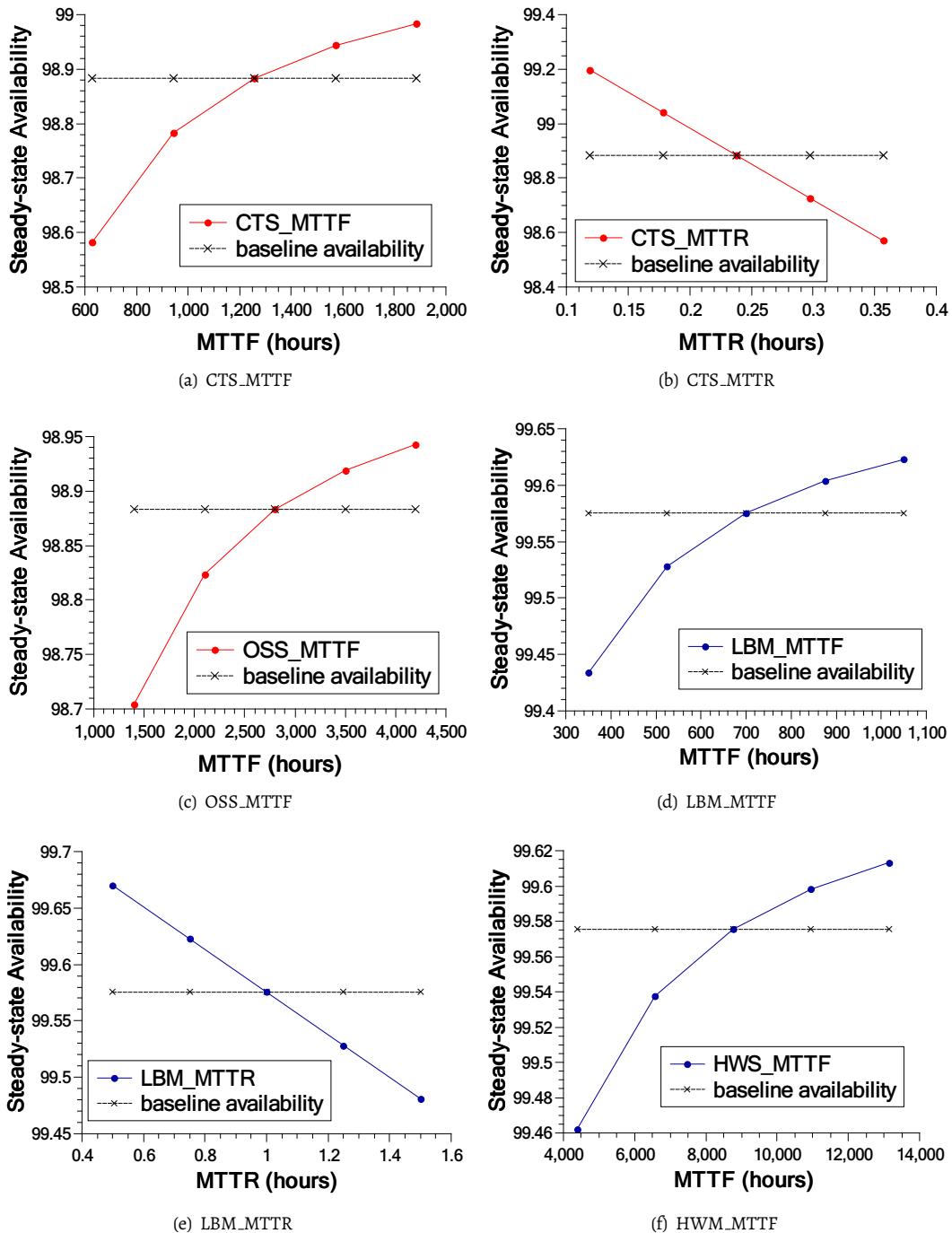


Figura 167 – Os resultados (a), (b) e (c) mostram as três transições mais importantes no modelo básico; Os resultados (d), (e) e (f) apresentam os resultados da análise de sensibilidade das transições mais importantes no modelo estendido.

SWITCH_TIME é 0,083333 horas, extraído de ([ARAUJO et al., 2019](#)).

20.6.3.1 Análise de Disponibilidade

Esta seção apresenta a análise de disponibilidade. Foram definidos cinco cenários, variando a quantidade de contêineres disponíveis (10, 20, 30, 40 e 50 contêineres). Esses cenários foram gerados com o objetivo de

Tabela 144 – Valores de entrada do modelo estendido.

Componente	MTTF (horas)	MTTR (horas)
Hardware	8760	8
Operating System	2800	1
Docker Daemon	2516	0.255
Container	1258	0.238
Load Balancer	700	1
Hardware (Idle)	17520	8
Operating System (Idle)	5600	1

comparar a disponibilidade dos dois modelos, bem como o impacto que o número de contêineres gera em cada arquitetura.

A Figura 168 mostra a disponibilidade e o tempo de inatividade calculados por análise estacionária com a ferramenta Mercury (SILVA et al., 2015). A variação de disponibilidade mostrada na Figura 168(a) mostra que a disponibilidade cai conforme novos contêineres são instanciados. Além disso, a disponibilidade do modelo estendido tende a cair mais lentamente do que no modelo básico. Acreditamos que este fato ocorra devido à redundância implementada no modelo estendido. Pois mesmo que adicionemos contêineres, haverá um componente redundante para suprir as eventuais falhas no *slave server 01*.

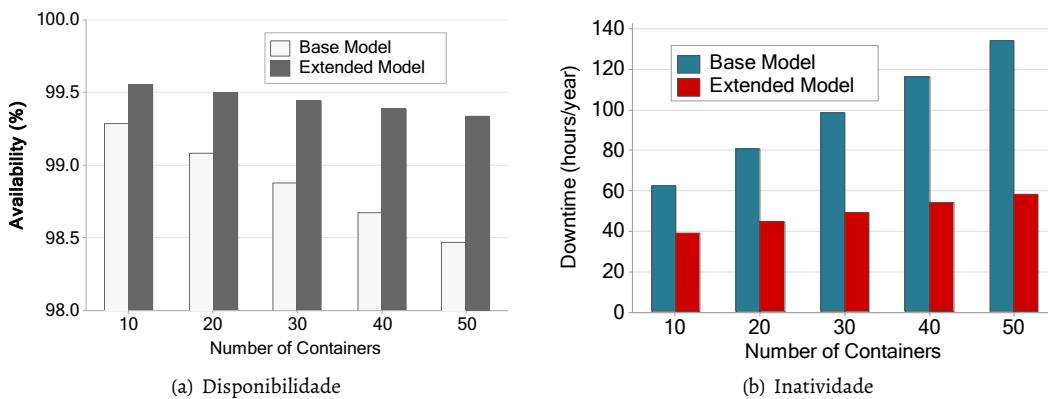


Figura 168 – Disponibilidade e tempo de inatividade das duas arquiteturas variando o número de contêineres.

A Figura 168(b) mostra o tempo de inatividade em horas por ano. O tempo de inatividade do modelo estendido tende a ser muito menor do que o do modelo básico. O modelo básico varia de 60 a 130 h/ano. O modelo estendido varia de 40 a 60 h/ano. O tempo de inatividade do modelo estendido permanece abaixo do modelo básico, mesmo com cinco vezes mais contêineres. Os valores individuais do modelo base crescem visivelmente mais e, a partir do terceiro cenário, tornam-se duas vezes mais longos que o modelo estendido. Após o quarto cenário, o tempo passa de 100 h/ano, enquanto o modelo estendido permaneceu abaixo de sessenta em todos os cenários.

Esses resultados são esperados, pois quanto mais componentes tivermos em um sistema, maiores serão as chances de um deles falhar, e quanto mais eles falharem, mais tempo leva para ser reparado. Nossa métrica considera o sistema funcionando quando todos os contêineres estão ativos; portanto, taxas de falha mais altas afetarão negativamente a disponibilidade. Esse comportamento é observado mesmo no modelo estendido. Porém, no modelo estendido, isso ocorre de forma muito mais sutil, pois existe um mecanismo de redundância no servidor.

20.7 Conclusão

Este capítulo propôs um modelo SPN para representar e avaliar o desempenho de uma arquitetura MEC subjacente. O modelo permite estimar o Tempo Médio de Resposta (MRT) e o nível de utilização de recursos na borda da rede. Um analista pode configurar até 5 parâmetros de entrada, o que permite um alto nível de flexibilidade de avaliação. Uma análise numérica foi realizada com base em dados reais de um artigo de referência.

A análise numérica permitiu observar o comportamento de quatro métricas (MRT, taxa de descarte, utilização do nó mestre e utilização dos nós escravos) em função do atraso na chegada. A análise numérica permitiu-nos concluir que o atraso na chegada é um parâmetro que tem um impacto significativo no desempenho do sistema. Para valores mínimos (até 3,0ms), foram observados níveis de descarte, o que pode comprometer o tempo médio de resposta.

O modelo refinado com estado absorvente permite estimar quando os aplicativos têm maior probabilidade de finalizar a execução, através do uso da CDF. Realizamos dois estudos de caso para demonstrar o uso desse modelo. No primeiro estudo, verificamos qual configuração de servidor escravo melhor atende aos requisitos de um administrador de infraestrutura, de acordo com o tempo total desejado para a execução da aplicação. O estudo constatou que o tempo total de execução aumenta à medida que os recursos diminuem. No segundo estudo, observamos a probabilidade de execução com base no tempo entre as chegadas. Observamos que, à medida que aumenta o tempo entre as chegadas, o tempo total necessário para concluir a execução do aplicativo também aumenta. O modelo de desempenho foi validado com um experimento real cujo resultado indicou igualdade entre o experimento e o modelo com valor p igual a 0,684 pelo Teste-T.

Este capítulo também propôs um modelo SPN para representar e avaliar a disponibilidade de uma arquitetura MEC subjacente. A partir desse modelo, foi possível analisar quais eram os componentes mais importantes do sistema, e foi possível gerar uma arquitetura estendida focada nesses componentes. Foram analisados cenários com diferentes quantidades de recipientes. O estudo concluiu que a arquitetura estendida mostrou uma melhoria considerável na disponibilidade em comparação com a arquitetura de base. Os contêineres se mostraram componentes de grande importância em ambos os modelos e que um servidor com responsabilidade de distribuição nem sempre é o gargalo do sistema. Os resultados do modelo estendido, diferentes do modelo base, sempre ficam acima de 99%, pois apresenta redundância nos componentes que estavam impactando a disponibilidade do modelo base.

Como trabalhos futuros, para o modelo de desempenho, pretendemos realizar outras análises numéricas, adicionando mais servidores ou considerando diferentes tipos de aplicações. Também pretendemos estender o modelo proposto para medir o gasto de energia e explorar a alocação entre várias torres do MEC. Para o modelo de disponibilidade, pretende-se adicionar redundância nos componentes do *servidor mestre* e também tentar explorar outros cenários para testar os modelos.

Uma Plataforma Distribuída para Avaliação de Desempenho em Aplicativos Móveis

Autores: João Marcos e Francisco Airton Silva

Resumo

O número de *Smartphones* em uso tem aumentado em níveis exponenciais mundialmente. Esse crescimento elevado tem trazido diversos desafios. Dentre eles, destaca-se a duração de bateria de dispositivos que não tem autonomia extensa. Diversas pesquisas tem focado no aprimoramento do hardware e do software para redução de consumo energético. As tecnologias de monitoramento de gasto energético por software não possibilitam uma comparação automática. Este capítulo propõe uma plataforma distribuída para monitoramento e comparação de consumo energético de aplicativos móveis de maneira automática, composta, em suma, por uma aplicação, denominada *APPCompare*. A aplicação desenvolvida realiza monitoramento, avaliação e comparação do consumo de energia de aplicativos instalados no dispositivo. Comparações realizadas utilizando métodos estatísticos precisos podem ser realizadas, onde podem ser escolhidos níveis de acurácia de 95 e 99%. Todas as informações que são obtidas pela aplicação podem sem enviadas para um servidor remoto, onde podem ser utilizadas para estudo posterior. Experimentos práticos foram realizados sobre aplicações de *streaming* de áudio, e avaliações de precisão e interferência foram executadas para investigar e quantificar os resultados obtidos pelo processo de monitoramento do gasto de energético de aplicações. Foi possível perceber que a ferramenta desenvolvida se mostrou muito eficaz na avaliação de consumo energético, com resultados precisos sobre as aplicações monitoradas, tornando a ferramenta aplicável em testes e ambientes reais.

Palavras-chaves: Consumo Energético, Avaliação de Desempenho, *Smartphones*, Monitoramento.

21.1 Introdução

O crescimento no uso de dispositivos móveis em geral tem sido exponencial nos últimos anos. De acordo com um relatório apresentado pela *Ericsson*¹ no ano de 2012, o número de *Smartphones* triplicará até 2018, chegando à marca de 3,3 bilhões de unidades no mundo. O crescimento do uso de *Smartphones* tem sido

exponencial considerando os relatórios anuais da Ericsson, como o de 2013, 2014, 2015 e 2016. relatórios afirmam que o número de usuários e dispositivos utilizados continuará a crescer rapidamente.

Os *Smartphones* são atualmente alimentados por baterias de Lítium-Ion. A duração de vida da bateria é de cerca de um ano a um ano e meio, o que representa entre 300 e 500 recargas. O que motiva o desenvolvimento de estratégias que minimizem o consumo energético de aplicativos sem necessidade de modificação no *hardware*, que representa maior custo (SILVA-FILHO et al., 2012).

Neste contexto, em decorrência da grande demanda por poder de processamento e recursos do dispositivo, desenvolvedores de aplicativos móveis também precisam se preocupar com a eficiência energética de seus *softwares*. Essa demanda cria uma grande necessidade por aplicações que permitam realizar estudos sobre consumo energético de dispositivos móveis.

A funcionalidade de comparação automatizada de consumo energético de aplicações móveis pode ser de grande valia. Quando executada a monitoração de mais de uma aplicação dentro do mesmo dispositivo, uma comparação que se utilize de métodos estatísticos de precisão, pode apresentar toda uma base estatística para comprovação de teorias de estudo, ou testes. Isso pode auxiliar diretamente no estudo e criação de aplicações móveis mais eficazes energeticamente.

Diversas pesquisas científicas foram realizadas a respeito do monitoramento de consumo energético de aplicativos móveis (CARROLL; HEISER, 2010; ZHANG et al., 2010; THIAGARAJAN et al., 2012; TRESTIAN et al., 2012; JUNG et al., 2012; CHOI, 2013; HAO et al., 2013; HAO et al., 2014). No entanto, nenhuma destas pesquisas oferece a possibilidade de comparação automática de consumo energético por aplicativo.

Existem aplicações que realizam o monitoramento do gasto de bateria, como por exemplo, o PowerTutor², que é amplamente utilizado na academia científica (XIAO et al., 2010; SAIPULLAH, 2012; LEE; JOE; KIM, 2012; BERNARDO et al., 2015). Porém, uma ferramenta como o PowerTutor somente realiza a monitoração de gasto energético de aplicações isoladas, não monitorando mais de uma aplicação. Também, com essa ferramenta, não é possível fazer o monitoramento de vários dispositivos, coletar seus resultados e enviar para servidores, para fins de estudos posteriores.

Apesar da heterogeneidade do mercado, a grande quantidade de smartphones, banda larga sem fio e computação em nuvem baseada em rede constitui uma vasta gama de oportunidades para desenvolvedores de aplicativos, chamando a atenção para as novas plataformas (CHOI, 2013).

21.1.1 Objetivos

21.1.1.1 Objetivo Geral

O objetivo geral deste trabalho é prover uma forma prática de avaliar o gasto energético de aplicativos em diversos modelos de dispositivos móveis de forma distribuída e com baixo nível de intrusão.

21.1.1.2 Objetivos Específicos

1. Implementar um aplicativo móvel que avalie o gasto energético de um aplicativo através de múltiplas observações.
2. Implementar um aplicativo móvel que compare estatisticamente o gasto energético de dois aplicativos.
3. Implementar um servidor na nuvem que capte a avaliação de desempenho do aplicativo sendo executado em múltiplos dispositivos móveis.

21.1.2 Organização do trabalho

O restante deste capítulo se organiza da seguinte forma: a Seção 2 lista os objetivos gerais e específicos deste pré-projeto; a Seção 3 oferece conteúdo teórico para entendimento do pré-projeto e sua proposta; a Seção 4 mostra os trabalho relacionados com esta pesquisa e apresenta algumas de suas características; a

¹ Relatório de 2012 da Ericsson: <https://www.ericsson.com/res/docs/2012/ericsson-mobility-report-november-2012.pdf>;

² PowerTutor: <http://ziyang.eecs.umich.edu/projects/powertutor/>;

Seção 5 apresenta a proposta do presente pré-projeto, além seus métodos de avaliação e estudo de caso e a Seção 6 apresenta o cronograma a ser seguido por este pré-projeto.

21.2 Referencial Teórico

Este projeto aborda tecnologias bastante difundidas atualmente, além da sua utilização para desenvolvimento de novas técnicas e conhecimento na área científica. Estas tecnologias podem ser estudos sobre consumo energético de dispositivos móveis e seus desafios, testes comparativos, como teste *T-Student* e ferramentas gráficas para representação da variação de dados observados de uma variável numérica, como gráficos *boxplot*.

Para que haja um melhor entendimento do conteúdo estatístico, o conhecimento do teste *T-Student* e gráficos *Boxplot* se fazem necessários por serem métodos comparativos e ferramentas para estudos e verificação de hipóteses sobre parâmetros amostrais.

Neste tópico serão abordados todas as seções que servem de apoio teórico para o presente trabalho.

21.2.1 Desafios da Eficiência do Consumo Energético de Dispositivos Móveis

O avanço da tecnologia com a redução no tamanho do transistor tem propiciado a incorporação de funcionalidades cada vez mais complexas em tais dispositivos e, na medida do possível, tem seguido as projeções de *Moore*. Além disso, no atual estado da arte, a maioria dos *smartphones* é alimentada por baterias de *Lítium-Ion*, na qual não tem evoluído na mesma velocidade dos dispositivos móveis, aspecto este que motiva o desenvolvimento de estratégias que minimizem o consumo de energia desses dispositivos sem a necessidade de alguma alteração na arquitetura do hardware. (SILVA-FILHO et al., 2012)

Durante muito tempo, várias pesquisas, que tinham por objetivo estudo do consumo energético de dispositivos móveis, foram desenvolvidas usando a técnicas como o DVFS - *Dynamic Voltage and Frequency Scaling* é um método que pode ser usado para reduzir consumo de energia através do ajuste da frequência e tensão de operação do dispositivo de acordo com o status e restrições de desempenho do sistema (LIANG; LAI, 2010) - em processador aplicadas a sistemas de tempo real com base em distintos modelos de tarefas para redução de consumo de energia (TAVARES et al., 2007; TAVARES; MACIEL; SILVA, 2008). No entanto, a fonte do consumo de energia inclui não somente o processador, mas também outros componentes tais como memória, dispositivos periféricos bem como recursos do sistema operacional.

Outros trabalhos, como o de (CHOI; SOMA; PEDRAM, 2004), consideram a carga de acesso à memória externa no estudo do impacto do DFVS ou ainda, como em (RAJAN; ZUCK; POELLABAUER, 2006), definem taxas de acesso a memória para assim propor um algoritmo para a economia de energia através de um *trade-off* ótimo frequência-tensão.

A crescente complexidade dos aplicativos móveis levou os usuários a atualizar regularmente seu hardware para acompanhar os crescentes requisitos de desempenho, mas continua causando o problema da curta duração da bateria (KOSTA et al., 2012) . Além de focar na identificação e refatoração de odores de código de energia dentro de aplicações móveis (DINH et al., 2013; VETRO; ARDITO; MORISIO, 2013; GOTTSCHALK et al., 2012), uma tendência de pesquisa para estender a vida útil da bateria é o descarregamento de tarefas (*offloading*) computacionalmente intensivas de dispositivos móveis com recursos restritos a máquinas substitutas (SATYANARAYANAN et al., 2001). As abordagens de descarregamento existentes podem ser classificadas em termos das seguintes categorias: Descarregamento de alvos, descarregamento de granularidade de tarefa, tempo de decisão de descarregamento e modelos de energia para descarregar a tomada de decisão.

Em relação às metas de descarregamento, vários estudos se concentraram no descarregamento de tarefas para servidores em nuvem (LIANG; LAI, 2010; CUERVO et al., 2010; XIA et al., 2014; BARBETTE; SOLDANI; MATHY, 2015; NANDHINI; TAMILSELVAN, 2014) enquanto outros investigaram tarefas de descarregamento para dispositivos móveis conectados intermitentemente sem hospedar e manter servidores (SHI et al., 2012; MTIBAA; HARRAS; FAHIM, 2013). Quanto à granularidade da tarefa de descarregamento (KUMAR et al., 2013), uma unidade de programa descarregada pode ser um método (LIANG; LAI, 2010; CUERVO et al., 2010; SHI et al., 2012; XIAN; LU; LI, 2007; LI; WANG; XU, 2001), um módulo (LIN et al., 2015; CHU

et al., 2004), um Pacote Open Services Gateway Initiative (OSGi) (ALI et al., 2016) ou uma aplicação móvel (XIA et al., 2014). Em relação ao tempo de decisão de descarregamento, as decisões de descarregamento podem ser tomadas estaticamente (XIAN; LU; LI, 2007; RIM et al., 2006; OU; YANG; LIOTTA, 2006) ou dinamicamente (LIANG; LAI, 2010; CUERVO et al., 2010; XIA et al., 2014) - (LIN et al., 2015; ALI et al., 2016; NANDHINI; TAMILSELVAN, 2014; SHI et al., 2012). Para abordagens estáticas, as tarefas a serem descarregadas são determinadas durante a fase de desenvolvimento para reduzir a sobrecarga durante a execução. Por outro lado, abordagens dinâmicas podem se adaptar a diferentes condições de tempo de execução, como flutuantes larguras de banda de rede.

21.2.2 Teste Paramétrico

Testes de hipóteses são largamente utilizados na academia como recursos para tomada de decisão. Através do seu uso, dados observados por um experimento podem ser submetidos à métodos estatísticos (Fisher, Neyman-Pearson, Bayes), tornando possível a validação de determinada teoria científica. Dentre os testes paramétricos mais utilizados atualmente, se encontram o Teste *T-Student* e ANOVA.

Um teste paramétrico é um teste estatístico que assume que os dados têm certa distribuição de probabilidade (PASTORE et al., 2010).

Segundo (SALOTTI, 2010), o teste paramétrico é aplicado no caso de atendimento aos pressupostos de distribuição normal das variáveis e de que as variâncias das duas amostras serem homogêneas.

Métodos paramétricos, em sua base, analisam a variabilidade dos resultados da variável dependente, de forma que seja permitida a aceitação ou refutação da hipótese nula, onde é postulado que os resultados investigados estão corretos.

21.2.2.1 Teste T

Sendo o teste de hipótese mais conhecido, o teste T de *Student* é de grande simplicidade e de fácil implementação (Podendo ser utilizado em *software*, como Minitab³, ou linguagens para cálculos estatísticos, como R⁴). O Teste T pode ser utilizado, principalmente, para avaliar a diferença entre médias de duas amostras.

Para (JUNIOR, 2017), o teste *T-Student*, ou simplesmente Teste T, é o método mais utilizado para se avaliar as diferenças entre as médias de dois grupos.

Para utilização correta do Teste T, alguns pressupostos devem ser verificados. Segundo (FRANCO, 2015), três suposições devem ser testadas, sendo elas:

1. As duas amostras devem ter distribuição normal;
2. As duas amostras devem ter a mesma variância;
3. As duas amostras devem ser independentes, ou seja, é preciso saber como os dados foram correlacionados.

(JUNIOR, 2017) afirma ainda que na comparação da média de uma amostra com a de uma população deve utilizar a seguinte fórmula:

$$t = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}}$$

Onde

\bar{x} é a média da amostra

μ é a média da população (ou referência)

s é o Desvio Padrão

³ Minitab: <http://www.minitab.com/pt-BR/>

⁴ Linguagem R: <https://www.r-project.org/>

n é o número de sujeitos

De acordo com (MINITAB, 2017c), uma propriedade importante do teste t é a sua robustez contra pressupostos de normalidade da população. Em outras palavras, com amostras grandes, os testes T são muitas vezes válidos mesmo quando o pressuposto de normalidade é violada. Esta propriedade os torna um dos procedimentos mais úteis para fazer inferências sobre médias da população.

21.2.2.2 ANOVA

Amplamente utilizado na academia, o método de análise de variância, ou ANOVA, é muito utilizado avaliação de afirmações sobre média de populações. Um dos principais objetivos do ANOVA é a verificação de existência de diferença entre médias, e se fatores conseguem exercer influência sobre alguma variável dependente.

A análise de variância (ANOVA), testa a hipótese de que as médias de duas ou mais populações são iguais. As ANOVAs avaliam a importância de um ou mais fatores, comparando as médias de variáveis de resposta nos diferentes níveis de fator. A hipótese nula afirma que todas as médias de população (médias de nível de fator) são iguais, enquanto a hipótese alternativa afirma que pelo menos uma é diferente. (MINITAB, 2017b)

Para que o teste ANOVA possa ser executado, são necessários uma variável de resposta contínua e pelo menos um fator categórico com dois ou mais níveis (MINITAB, 2017b). O teste ANOVA, um pouco diferente do teste T, funciona bem com valores onde pressuposição de normalidade é violada, com excessão de situações onde as distribuições são demasiadamente elevadas.

21.2.3 Gráficos BoxPlot

O gráfico Boxplot é utilizado regularmente em pesquisas científicas com o objetivo de agrupar e analisar dados quantitativos. Exibindo medidas de tendência central não paramétrica (como a mediana), forma de distribuição ou simetria de amostra, valores atípicos (*outliers*), é um recurso gráfico que pode substituir o uso de tabelas em casos específicos.

O Boxplot, também conhecido como gráfico de caixa, é um gráfico estatístico que possibilita representar a distribuição de um conjunto de dados com base em alguns parâmetros descritivos. (CAPELA; CAPELA, 2011)

Dentre as informações que o gráfico Boxplot oferece, estão informações sobre localização e dispersão. Porém, seu valor pode ser associado ao fato de que ele fornece sobre a cauda da distribuição.

Na utilização de gráficos Boxplot pode haver a ocorrência de pontos desgarrados, chamados *outliers*. Os *outliers* podem causar adversidades nas tomadas de decisões feitas a partir da análise dos dados, caso não sejam devidamente considerados.

De acordo com (FARIAS, 2013), o gráfico Boxplot é um gráfico construído com base no resumo dos cinco números, constituído por:

- Valor Mínimo
- Primeiro Quartil (Q1)
- Mediana (Segundo Quartil - Q2)
- Terceiro Quartil (Q3)
- Valor Máximo

Na Figura 169 é apresentado um exemplo de gráfico Boxplot, onde se fazem presentes todos os elementos listados anteriormente. Nas extremidades, os valores mínimos e máximos podem ser percebidos pelos componentes limite inferior e limite superior, respectivamente. Logo ao centro, o terceiro, segundo (mediana), e primeiro quartis. Vale ressaltar que a dispersão é representada através da amplitude do gráfico, podendo ser calculada como valor máximo - valor mínimo, e quanto maior a amplitude, maior será a variação dos dados.

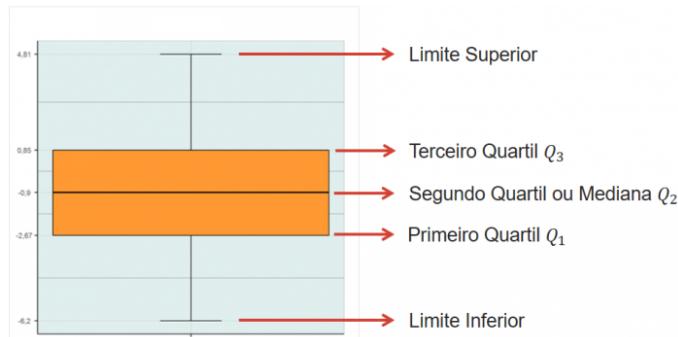


Figura 169 – Exemplo de gráfico Boxplot.

21.2.4 Design of Experiments (DoE)

O *Design of Experiments* (DoE - Planejamento de Experimentos) está bem estabelecido e é aplicado com sucesso em vários campos da indústria, como produto para desenvolvimento de projetos, (GUNST, 1996), química (LAZIC, 2006) e engenharia de software (KUHN; REILLY, 2002). Existe uma quantidade considerável de livros sobre a metodologia (SIEBERTZ; BEBBER; HOCHKIRCHEN, 2017; SELTMAN, 2012; ANTONY, 2006); Portanto, a metodologia geral aqui será descrita brevemente. Em cada parte da aplicação, será exemplificado cada caso metodológico para o estudo de caso.

O conceito geral de DoE foi criado por uma série de experimentos reais simulados em um sistema ou modelo de sistema sob observação. Em cada experimento, um ou vários parâmetros de projeto são alternados e o impacto no comportamento do sistema é avaliado. Quais parâmetro são alterados e como eles são alterados é definido usando um plano de experimento. O objetivo é obter o máximo de informações possíveis usando a menor quantidade de experimentos, e experimentos com o menor custo em termos de esforço computacional ou setup físico. O comportamento do sistema com base nas mudanças de parâmetros é observado usando conjuntos de saídas. No contexto do DoE, as saídas podem ser referenciadas como “indicadores de performance”, os parâmetros de projeto como “fatores” e os valores das configurações de “Níveis”.

Para analisar o impacto de cada fator no sistema e a interação com os outros fatores, várias combinações de fatores precisam ser testadas e, portanto, exige um experimento. Devido à grande quantidade de combinações possíveis, este é um esforço muitas vezes inviável em termos de tempo e custos. O DoE oferece uma coleção de métodos - referidos como planos experimentais ou tabelas de design - para reduzir a quantidade de experimentos necessários para encontrar informações precisas com o menor número de experimentos possível. O tipo de plano usado depende do objetivo do experimento. Encontrar a tabela de design ideal, por exemplo, a menor quantidade de experimentos necessários para descrever o comportamento dos sistemas corretamente, tem sido objeto de intensa pesquisa. Usando um conjunto de métodos estatísticos de avaliação nos dados resultantes do experimento, o impacto, efeitos e interações dos fatores em relação aos indicadores de performance escolhidos é avaliado. Os resultados do experimento podem ser usados para formular um modelo substituto matemático, também chamado de metamodelo em disciplinas de engenharia (MILLER et al., 2014; THOMAS et al., 2014). Neste trabalho foi aplicado a metodologia DoE para avaliação da arquitetura proposta. Três tipos de gráficos são adotados usualmente em estudos com DoE: Gráfico de Pareto, Gráfico de Efeitos Principais e Gráfico de Intereração (ESPEJO, 2006).

O Gráfico de Pareto permite detectar qual o efeito da interação de fatores é mais importante para o processo ou estudo de otimização de projeto com o qual se deve lidar. Exibe os valores absolutos dos efeitos e desenha uma linha de referência no gráfico. Qualquer efeito que ultrapasse essa linha de referência é potencialmente importante. Um gráfico de Pareto é construído como na Figura 170, por exemplo. O gráfico mostra que os fatores B (*tool geometry*) e C (*cutting angle*) e a interação AC são mais importantes.

O outro gráfico é chamado de “Gráfico de Efeitos Principais” e é o gráfico dos valores médios de resposta em cada nível de um parâmetro de projeto ou variável de processo. Pode-se usar esse gráfico para comparar

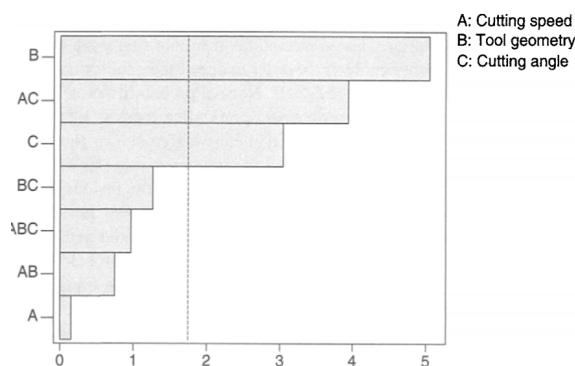


Figura 170 – Gráfico de Pareto com Efeitos Padronizados

a força relativa dos efeitos de vários fatores. O sinal e a magnitude de um efeito principal nos diriam o seguinte:

- O sinal de um efeito principal nos diz a direção do efeito, ou seja, se o valor médio da resposta aumenta ou diminui.
- A magnitude nos diz a força do efeito.

Se o efeito do parâmetro de projeto ou processo é positivo, isso implica que a resposta média é maior em nível alto do que em nível baixo da configuração do parâmetro. Em contraste, se o efeito for negativo, isso significa que a resposta média em nível baixo de configuração do parâmetro é maior que em nível alto. A Figura 171 ilustra o efeito principal da temperatura na resistência à tração de uma amostra de aço. Como você pode ver pela figura, a resistência aumenta quando as configurações da temperatura variam para o nível baixo (ou seja, 1 para -1). O efeito de um parâmetro de projeto ou processo (ou fator) pode ser calculado matematicamente usando a simples equação 21.1:

$$E_f = \bar{F}_{(+1)} - \bar{F}_{(-1)} \quad (21.1)$$

Quando $\bar{F}_{(+1)}$ significa a resposta média na configuração de alto nível de um fator, e $\bar{F}_{(-1)}$ significa a resposta média na configuração de baixo nível de um fator.

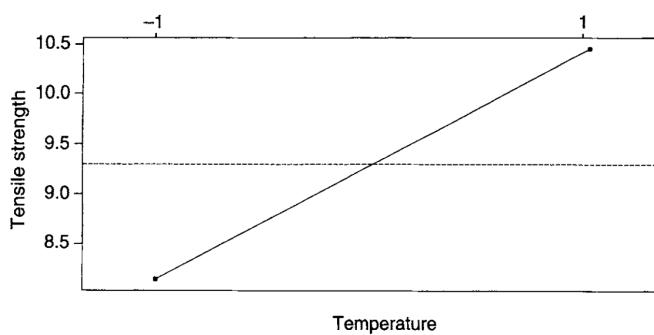


Figura 171 – Gráfico de Efeitos Principais da Temperatura na Resistência à Tração

O processo de "Interações", identifica efeitos importantes e determina sua magnitude, logo, as interações entre os efeitos são cruciais. As interações ocorrem quando o efeito de um fator depende do nível de outro fator. Uma medida de design sempre aborda vários fatores. Entender como esses fatores interagem em que magnitude permite para escolher a melhor combinação de medidas, revelando combinações de fatores

com efeito cumulativo ou degradante. A interação $I_{A,B}$ entre dois fatores A e B pode ser calculada usando a equação 21.2.

$$I_{A,B} = \frac{1}{2}(E_{A,B(+1)} - E_{A,B(-1)}) \quad (21.2)$$

O $E_{A,B(+1)}$ é o efeito do fator 'A' no nível alto do fator 'B' e $E_{A,B(-1)}$ é o efeito do fator 'A' no nível baixo do fator 'B'.

Para determinar se dois parâmetros de processo estão interagindo ou não, pode-se usar uma ferramenta gráfica simples, porém poderosa, chamada de gráficos de interação. Se as linhas no gráfico de interação forem paralelas, não haverá interação entre os parâmetros do processo. Isso implica que a mudança na resposta média do nível baixo para o alto de um fator não depende do nível do outro fator. Por outro lado, se as linhas não são paralelas, existe uma interação entre os fatores. Quanto maior o grau de afastamento de ser paralelo, mais forte o efeito de interação. Para interação sinérgica, as linhas no gráfico não se cruzam. Por exemplo, a Figura 172(a) é um exemplo de interação sinérgica.

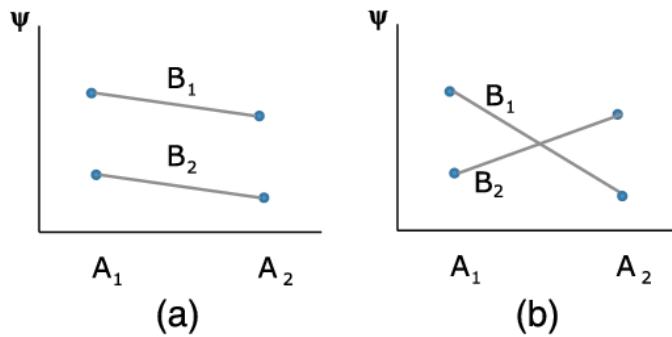


Figura 172 – Gráfico de Interações Sinérgica e Antagônica

Na interação antagônica, as linhas no gráfico se cruzam. Isso pode ser ilustrado na Figura 172(b). Neste caso, a mudança na resposta média para o fator A em nível baixo (representado por A_1) é visivelmente alta em comparação com o nível alto. Em outras palavras, o fator A é menos sensível à variação na resposta média em alto nível do fator B.

21.3 Trabalhos Relacionados

Esta Seção apresenta alguns trabalhos produzidos relacionados ao conteúdo de estudo do consumo energético de dispositivos móveis. Na apresentação dos trabalhos foram destacadas as correlações com o atual trabalho, bem como as diferenças encontradas entre os mesmos. A Tabela 145 elenca os fatores comparativos relevantes a este projeto, que são: Métricas de Monitoramento, Comparação Automática, Contexto, Envio para Servidor Remoto e Utiliza Metodologia para Planejamento de Experimentos. Os trabalhos estão ordenados por tema e ano de publicação (de 2010 a 2018).

O primeiro critério de avaliação dos trabalhos foi a métrica de monitoramento utilizada no estudo. Este critério visa identificar onde o estudo foi focado, e onde esperava encontrar seus resultados. Mesmo com o uso de diferentes ferramentas, os trabalhos aqui relacionados se utilizam da criação de modelos de monitoramento e estimação de consumo energético dentro dos contextos avaliados. Vários trabalhos se utilizaram da métrica de consumo energético do dispositivo, sendo este também o foco do presente trabalho. O segundo critério foi o contexto, que tinha por objetivo identificar qual/quais ambiente(s) e/ou dispositivo(s) utilizados nos trabalhos. Os trabalhos aqui relacionados se utilizaram de nichos específicos de *hardware*, como Arquitetura de computadores e Dispositivos Móveis, e *software*, como aplicações móveis. O monitoramento foi feito em diversos tipos de aparelhos ou ambientes. Os demais critérios foram a oferta da possibilidade de submissão dos resultados obtidos para um servidor remoto, realização de comparação automatizada dos resultados obtidos pela monitoração e utilização da metodologias voltadas para o plane-

Tabela 145 – Trabalhos Relacionados

Tema	Trabalho	Métricas de Monitoramento	Contexto	Comparação Automática	Envio para Servidor Remoto	Utiliza Metodologia para Planejamento de Experimentos
Avaliação de Desempenho de Performance de Sistema	(ADHIANTO et al., 2010)	CPU	Programas de Computador	Não	Não	Não
	(ZHANG et al., 2010)	Performance de Sistema e Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(HAO et al., 2014)	Performance de Interface e Uso de Rede	Aplicativos Móveis	Não	Não	Não
Avaliação de Desempenho de Hardware	(ZHANG et al., 2010)	Performance de Sistema e Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(HAO et al., 2013)	Consumo Energético, CPU, Memória RAM, Wi-Fi e GPS	Aplicativos Móveis	Não	Não	Não
Monitoração do Consumo Energético	(CARROLL; HEISER, 2010)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(THIAGARAJAN et al., 2012)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(JUNG et al., 2012)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(TRESTIAN et al., 2012)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(CHEN et al., 2013)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(CHOI, 2013)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	(MENDONÇA et al., 2015)	Consumo Energético e Performance	Dispositivos Móveis	Não	Não	Não
	(QIAN; ANDRESEN, 2015)	Consumo Energético e Performance de Sistema	Dispositivos Móveis	Não	Não	Não
	(BREITBACH; SANDERS; SCHULTES, 2018)	Consumo Energético	Dispositivos Móveis	Não	Não	Não
	Este Trabalho	Consumo Energético	Dispositivos Móveis	Sim	Sim	Sim

jamento de sus experimentos, para planejamento dos experimentos. Nenhum dos trabalhos utilizou estes três últimos critérios, sendo esse um diferencial deste projeto.

Os trabalhos relacionados foram agrupados em temas para uma melhor explanação de seus desafios e contribuições. Os trabalhos de (ADHIANTO et al., 2010; ZHANG et al., 2010; HAO et al., 2014) se focaram no estudo da avaliação de performance do sistema. O ambiente empregado por (ADHIANTO et al., 2010) foi computadores físicos, enquanto que (ZHANG et al., 2010; HAO et al., 2014) se focaram em dispositivos e aplicativos móveis.

(ADHIANTO et al., 2010), em seu trabalho, traz uma visão geral sobre a ferramenta *HPCTToolkit* e suas funcionalidades. A ferramenta em questão consiste em um conjunto de funcionalidades para coleta de medição de performance de executáveis totalmente otimizados sem adicionar instrumentação. É permitido analisar binários para entender a estrutura do código otimizado, correlacionando medições com a estrutura do programa, e apresentando os dados de performance resultantes em um organização que facilita a análise.

Em seu trabalho, (HAO et al., 2014) propõe PUMA, um *framework* dinâmico de análise que pode ser instanciado para um grande número de tarefas de análise dinâmica. A ferramenta proposta permite que seus usuários especifiquem estados do aplicativo para explorar. Além disso, fornece acesso programático ao estado da máquina, tempo de execução do aplicativo para análise e suporta o ambiente de tempo de execução dinâmica modificação. Ele encapsula os componentes comuns dos sistemas de análise dinâmica existentes e expõe uma série de ganchos que podem ser programados com um evento de alto nível linguagem de *script*.

Os trabalhos de (ZHANG et al., 2010; HAO et al., 2013) se focaram no estudo da avaliação de desempenho de *hardware* dos dispositivos. Em seu trabalho (ZHANG et al., 2010), cria um modelo de cálculo de consumo energético para dispositivos móveis. Esse modelo criado é baseado no monitoramento dos componentes de *hardware* do dispositivo monitorado. Já em (HAO et al., 2013), é feita uma avaliação de desempenho dos dispositivos através do uso de aplicações externas que verificam fatores como uso de componentes de *hardware* e consumo energético do dispositivo monitorado. São verificadas combinações de componentes do dispositivo enquanto ligados ou desligados (modos de economia de bateria, modo avião). Este trabalho também foca no monitoramento dos componentes de *hardware*, porém com objetivo de obtenção do consumo energético dos dispositivos.

Os trabalhos de (CARROLL; HEISER, 2010; THIAGARAJAN et al., 2012; JUNG et al., 2012; TRESTIAN et

al., 2012; CHEN et al., 2013; CHOI, 2013; MENDONÇA et al., 2015; QIAN; ANDRESEN, 2015) e (BREITBACH; SANDERS; SCHULTES, 2018) tratam-se de estudos focados, principalmente, na avaliação do consumo energético de dispositivos móveis.

O trabalho de (CARROLL; HEISER, 2010) fornece uma base para a compreensão e gestão consumo de energia de dispositivos móveis. Foi utilizada com abordagem a medição de consumo energético de um dispositivo móvel (Openmoko Neo Freerunner), quebrando os subsistemas maiores do dispositivo, sob uma visão em diferentes cenários. Foram tomadas conclusões sobre a distribuição de energia em CPU, memória, tela, hardware de processamento gráfico (GPU), áudio, armazenamento e variadas interfaces de rede.

(THIAGARAJAN et al., 2012) analiza o consumo energético do navegador nativo de dispositivos com Sistema Operacional *Android* em sites populares como Facebook. O autor utilizou um multímetro localizado na bateria do telefone medindo o consumo energético do dispositivo enquanto ele carrega e renderiza elementos individuais de páginas web: como *Javascript*, *CSS*.

(JUNG et al., 2012) propõe um modelo energético para cada componente de *hardware* do *smartphone*. Uma abordagem onde é criada um modelo para cada componente utilizando BMU (*Battery Monitor Unit*), que é um componente de *hardware* equipado na maioria dos dispositivos, que trás informações como voltagem, tensão e temperatura da bateria.

No trabalho de (TRESTIAN et al., 2012), é investigado o relacionamento entre o ambiente sem fio e o consumo energético do dispositivo móvel no contexto de entrega de vídeo (*streaming* de vídeo). O autor verificou os impactos de fatores como qualidade do *link* sem fio e carregamento da rede no consumo energético em um dispositivo *Android* enquanto ele executa *streaming* de vídeo sobre uma rede IEEE 802.11g.

No trabalho de (QIAN; ANDRESEN, 2015), é apresentado Jade, um sistema de computação inteligente para *offloading* para dispositivos móveis. O objetivo do autor é minimizar o consumo energético de aplicações móveis através de descarregamento de computação de granulação final. Como uma das principais contribuições, está um algoritmo de agendamento que, automaticamente, transporta cargas de trabalho para o servidor baseado na performance e energia necessários para a execução da tarefa.

O trabalho de (CHEN et al., 2013) avalia o consumo do *display* de vários modelos *smartphones* Samsung. É discutido ainda a modelagem de diferentes gerações de telas com a tecnologia AMOLED, além de se conduzida uma análise detalhada de várias aplicações de multimídia populares.

(CHOI, 2013) trás uma análise de dissipação energética sob dois diferentes pontos de vista, usando cenários e componentes de *smartphones*. O autor desenvolve um modelo de enfileiramento para consumo energético em diferentes componentes, e analisa o tempo de vida da bateria com diferentes comportamentos de uso.

Para seleção dos trabalhos, foi utilizado o critério de trabalhos mais relevantes para a academia, sendo escolhidos os trabalhos que mais foram citados por outros estudos. Também foi utilizado o critério de autores mais relevantes, onde os autores mais citados, e com mais publicações dentro da área de estudo sobre consumo energético foram escolhidos.

Todos os trabalhos comentados acima propuseram soluções para, principalmente, a eficiência energética dos dispositivos, ou ambientes, que estavam sendo utilizados em suas respectivas pesquisas. Porém, nenhum trabalho se utilizou de técnicas mais avançadas para planejamento de seus experimentos, como o DoE. Este é um dos principais diferenciais do presente trabalho.

Pode ser comentado, também, que todos os trabalhos aqui relacionados não se utilizaram de metodologias estatísticas para execução de seus comparativos. Além disso, nenhum dos trabalhos realiza um比较 totalmente automatizada de seus resultados obtidos, fazendo tudo manualmente, ou mesmo com a utilização de softwares, acabam sendo feitas totalmente por seus autores. Este também é um diferencial deste trabalho em relação aos demais.

21.4 Uma Plataforma Distribuída para Avaliação de Desempenho em Aplicativos Móveis

21.4.1 Software para monitoração, avaliação e comparação de consumo energético de Aplicações

Esta Seção tem por objetivo apresentar a aplicação desenvolvida, além de explanar sobre sua arquitetura e funcionamento.

21.4.1.1 Arquitetura Proposta

A plataforma que foi proposta para comparação de aplicativos é formada pela ideia de um sistema distribuído que coleta informações de monitoramento de diversos dispositivos.

Em cada dispositivo, o aplicativo desenvolvido, denominado *APPCompare*, executa a comparação e envia as informações para o servidor. O número de dispositivos (*Smartphones* e *Tablets*) que podem participar de um experimento como esse não é limitado pela plataforma. A única restrição que pode ocorrer é por parte do servidor configurado para recebimento dos dados. O usuário tem a liberdade de definir seu servidor na Nuvem para armazenamento.

O *APPCompare*, para realizar a monitoração de gasto energético das aplicações, se utiliza do mesmo modelo de cálculo de consumo da ferramente *PowerTutor* (ver Seção 21.4.2 para mais informações). Esta ferramenta é bastante utilizada na academia, principalmente por seu reconhecido modelo de cálculo de consumo, que apresenta bastante eficiência e acurácia nos resultados obtidos.

Fazendo parte da plataforma de avaliação de desempenho distribuída, foi desenvolvido um aplicativo que realiza comparação automática entre aplicativos móveis, considerando-se o critério de consumo energético. A aplicação funciona em seis fases:

1. **Seleção de Aplicativos:** Cada usuário pode selecionar um ou mais aplicativos que estão instalados no sistema para realizar a investigação. A Figura 175 (A) apresenta a tela para seleção de aplicativos.
2. **Configuração do Experimento:** Nesta etapa o usuário deve escolher o número de observações que serão realizadas pelo *APPCompare*. O número de observações definido é o mesmo para todos os aplicativos avaliados. A Figura 175 (B) mostra a tela para configuração do experimento.
3. **Execução do Experimento:** Nesta etapa são abertos, em sequência (não simultaneamente, para uma aplicação não interferir no consumo energético da outra), os aplicativos avaliados. Após a abertura do aplicativo em avaliação, o usuário pode usar funcionalidades cujo gasto energético deseja conhecer e comparar com os outros selecionados. Por exemplo, a comparação entre o consumo energético do serviço de mensagens do *Whatsapp* e *Facebook Messenger* seria um experimento possível. Para execução do experimento foi adotado o conceito de observações por segundo, onde são capturados os consumos energéticos das aplicações avaliadas no intervalo de um segundo, formando uma observação. O tempo de execução do experimento é variável conforme o número de observações, se forem 30 observações, será, assim, 30 segundos para monitoramento do aplicativo.
4. **Geração de Relatório em Gráficos:** Nesta etapa o *APPCompare* executa uma análise estatística e gera gráficos de fácil interpretação do usuário. Um gráfico de barras apresenta os consumos energéticos durante cada observação para cada aplicativo (ver Figura 176). Assim, pode ser observado o crescimento do consumo energético com o passar das observações. O gráfico *Boxplot* é utilizado para permitir a avaliação da distribuição e amplitude dos dados de gasto energético. O *BoxPlot* fornece informação sobre dispersão, *outliers* de dados e identificar o aplicativo com maior eficiência (menor consumo energético); Na Figura 176 (A) pode ser observado o aumento de consumo energético com o passar das observações. O eixo X apresentada cada uma das observações coletadas. O eixo Y apresenta o consumo energético. A 176 (B) apresenta gráficos *Boxplot*, onde é mostrada uma comparação do gasto energético das aplicações monitoradas.
5. **Geração de Relatório Conclusivo:** Posteriormente é apresentado uma conclusão estatística da comparação. O resultado do teste com T-Student é mostrado quando a comparação envolve somente dois aplicativos. O *APPCompare* informa se as amostras são estatisticamente iguais ou diferentes. Neste ponto o usuário pode escolher enviar o resultado para o servidor na nuvem ou simplesmente salvar os valores no próprio dispositivo.
6. **Envio para Servidor Remoto:** Após todas as tarefas anteriores realizadas, o *APPCompare* permite o envio de todas as informações e dados estatísticos obtidos para um servidor remoto. O servidor utilizado para envio das informações é o *Firebase*.

A Figura 173 apresenta um fluxograma simplificado com a sequência de passos realizada pela aplicação APPCompare.

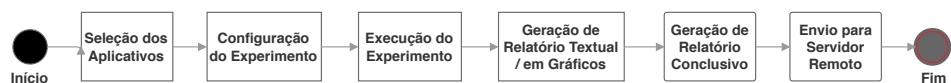


Figura 173 – Fluxograma com as fases da aplicação APPCompare

A Figura 174 mostra como se organiza o funcionamento e arquitetura do APPCompare.

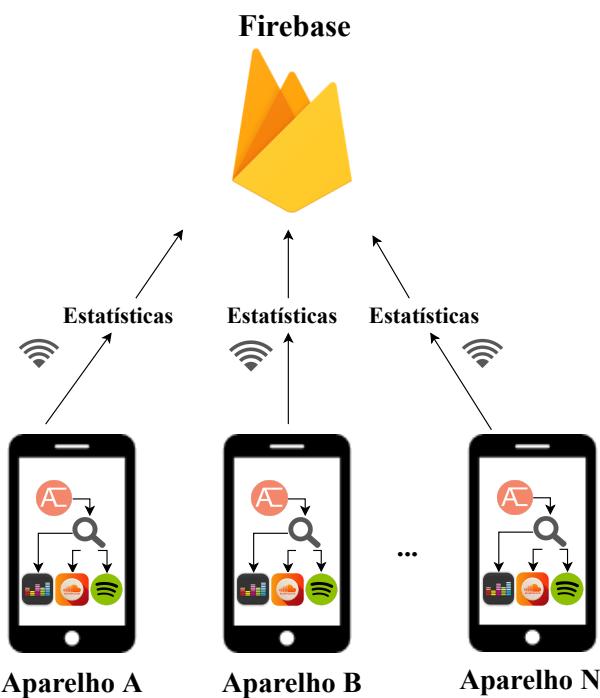


Figura 174 – Esquematização da Arquitetura

21.4.2 Funcionamento da Plataforma

Para criação de uma aplicação móvel capaz de executar monitoramento e mensuração do consumo energético do dispositivo foi necessário se utilizar de um modelo matemático complexo de cálculo de consumo. Para isto foi utilizado modelo criado por (ZHANG et al., 2010).

O modelo utilizado depende apenas do conhecimento da curva de descarga de tensão da bateria e acesso a um sensor de tensão da bateria, que está disponível na maioria dos *smartphones*. O Trabalho de (ZHANG et al., 2010), procurou pelos componentes do dispositivo que mais apresentavam relevância no consumo energético. Excluindo componentes com menor impacto, como cartão SD (memórias externas), os componentes modelados são: CPU e Display LCD, bem como GPS, Wi-Fi, interfaces de celular e de áudio.

A seguir, uma descrição sobre o modelo criado por (ZHANG et al., 2010), descreve como o autor separou os componentes mais relevantes para a obtenção do gasto energético do dispositivo. O autor desenvolveu a aplicação PowerTutor, já citada neste trabalho, que se utilizada do modelo de mensuração de gasto energético criado pelo mesmo no mesmo trabalho. O aplicação desenvolvida APPCompare se utiliza do modelo de medição de consumo energético da proposta.

- **CPU:** O consumo de energia da CPU é fortemente influenciado por utilização e frequência de CPU. Variação dinâmica, vazamento, e consumos de energia do circuito periférico invalida simples aproximações

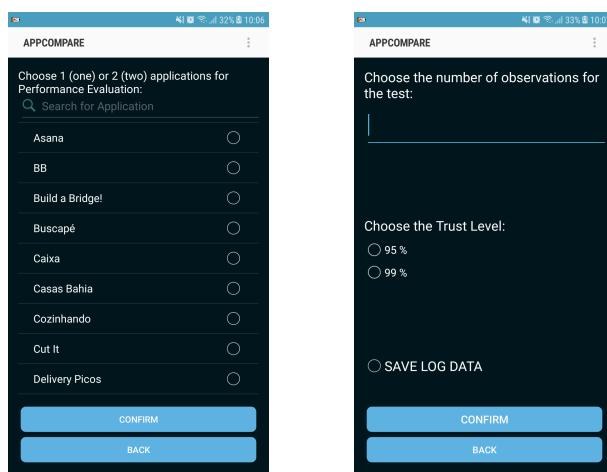


Figura 175 – (A) Seleção de Aplicativos (B) Configuração do Experimento

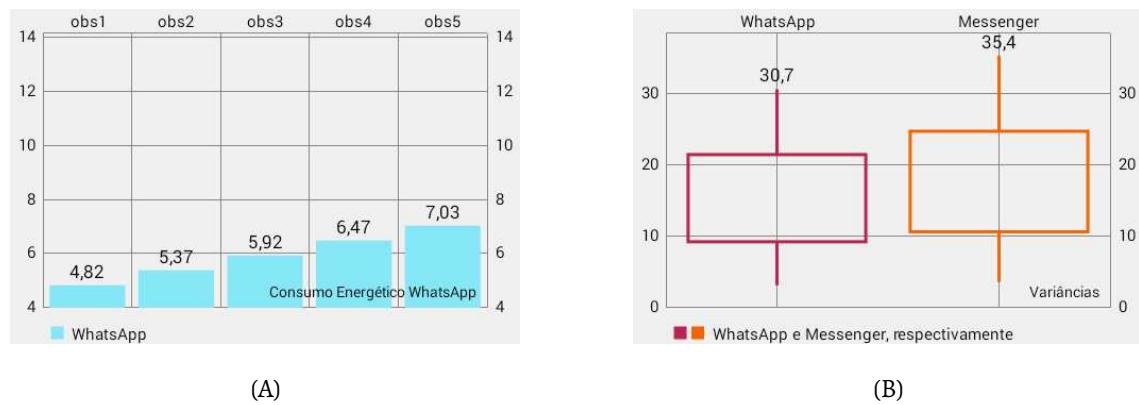


Figura 176 – (A) Gráficos de Barras e (B) Boxplot

de relações de potência cúbica de potência. Neste modelo, é medida a dependência do consumo de energia da CPU nas configurações de utilização e frequência-tensão.

- **LCD:** O modelo de energia do monitor LCD é derivado usando um programa de treinamento que liga e desliga o LCD e altera seu brilho. O autor simplificou em 10 níveis de brilho uniformemente distribuídos.
- **GPS:** Foi considerada a influência das seguintes variáveis relacionadas ao GPS: modo (por exemplo, ativo, dormir, ou desligado), o número de satélites detectados e a força do sinal de cada satélite. Todas essas variáveis são acessadas usando a API do *Android Software Development Kit*.
- **Wi-Fi:** Para obter o modelo Wi-Fi, foram considerados dois parâmetros de rede: taxa de dados e taxa de canal. O modelo de energia do Wi-Fi é derivado pela troca de tamanho fixo (1 KB) de pacotes TCP entre o smartphone e um servidor local. O modelo de energia Wi-Fi depende de quatro variáveis do sistema: número de pacotes transmitidos e recebidos por segundo (npackets), taxa de canal de *uplink* (*Rchannel*) e taxa de dados de uplink (*Rdata*). O Interface Wi-Fi tem quatro estados de energia: baixa potência, alta potência, *ltransmit* e *htransmit*. *Ltransmit* e *htransmit* são estados em que a placa de rede entra brevemente ao transmitir dados. Depois de enviar os dados, a placa retorna ao seu estado de energia. Ao transmitir em altas taxas de dados, a placa se altera apenas brevemente no estado de transmissão. O tempo para o estado de transmissão de baixa potência é ainda mais curto. O consumo de energia do componente Wi-Fi em qualquer estado de transmissão é de aproximadamente 1.000

mW. O estado de baixa energia é inserido quando a interface Wi-Fi não é enviar nem receber dados a uma taxa alta.

- **Celular:** O modelo de interface celular é derivado pelo envio Pacotes UDP entre um smartphone e um servidor local através da rede 3G da T-Mobile UMTS. Os tamanhos dos pacotes variam de 10 B a 1 KB. Para cada tamanho de pacote, variamos o atraso entre transmissões de 0 s a 12 s em intervalos de 0,1 s. Os resultados são semelhante para pacotes TCP.
- **Áudio:** A interface de áudio foi modelada medindo o consumo de energia quando não é usado, e quando um arquivo de áudio é reproduzido em volumes diferentes. (ZHANG et al., 2010) percebeu que o uso da interface de áudio influencia o consumo energético, mas o volume do alto-falante não.

A tensão de uma bateria de íons de lítio muda durante a descarga, permitindo a taxa de esgotamento de energia (consumo de energia) a ser estimado com base nas alterações da tensão observada. Um *logger* de relatório de informações roda em segundo plano para gravar a voltagem de saída da bateria.

A aplicação APPCompare está disponível para download na loja oficial do Google, o Google PlayStore. A Figura 177 traz um código QR que dá acesso ao link de download da aplicação.



Figura 177 – Código QR com link para download da aplicação

21.5 Avaliação

Esta Seção tem por objetivo mostrar a avaliação da plataforma proposta, e como os testes para análise de desempenho, acurácia e interferência da leitura foram realizados.

21.5.1 Estudo de Caso: Aplicações de Streaming de Áudio

O presente estudo de caso apresenta uma análise de sensibilidade em fatores que impactam no consumo energético de aplicações móveis em *Smartphones*. Baseado na literatura e componentes existentes em diversos dispositivos, os fatores WiFi, conexão à Redes Móveis (3G), Uso de GPS e Brilho da Tela foram escolhidos. O objetivo final é conhecer os fatores que mais impactam no consumo energético de aplicações móveis executadas nos dispositivos. Como principais contribuições do presente trabalho podem ser listadas:

1. Identificação do impacto de fatores no consumo energético de aplicações;
2. Análise dos níveis de fatores analisados; e
3. Listagem de possíveis direções para execução mais eficiente, em termos de consumo energético, de aplicações instaladas no dispositivo.

O principal fator motivacional para escolha de aplicações de *Streaming* de Vídeo foi a utilização de todos os recursos do dispositivo, cujos a aplicação APPCompare é capaz de monitorar (WiFi, 3G ou Dados Móveis, CPU, Tela, Áudio e GPS).

As Tabelas 146 e 147 apresentam informações sobre os dispositivos e suas telas, respectivamente, que foram utilizados para os experimentos realizados.

Tabela 146 – Informações Técnicas sobre os Dispositivos Utilizados

Marca	Modelo	Versão Android	CPU	Memória RAM	Memória ROM
Sansumg	Galaxy SM G532MT	6.0	1.4 GHz Quad Core	1.4 GB	16 GB
Motorola	Moto G2 XT1068	6.0	1.2Ghz Quad-Core	1 GB	8 GB

Tabela 147 – Informações Técnicas sobre as Telas dos Dispositivos Utilizados

Marca	Modelo	Resolução	Densidade de Pixels	Tipo	Cores
Sansumg	Galaxy SM G532MT	540x960 pixels	220 PPI	TFT PLS	16 milhões
Motorola	Moto G2 XT1068	720x1280 pixels	294 PPI	TFT LCD IPS	16 milhões

21.5.1.1 Experimento

Nesta seção, são descritas os ambientes de testes utilizadas nos experimentos. O objetivo é ser capaz de estipular o impacto de cada fator (Tipos de Conexão, Qualidade da reprodução e Brilho da Tela) a fim de relatar a configuração com melhor eficiência na utilização das aplicações a serem avaliadas. A Tabela 148 apresenta os fatores e níveis para combinações e análise.

Considerando os fatores descritos na Tabela 148, os fundamentos da análise experimental demonstram que o numero combinações possíveis para o experimento é: $2 \times 2 \times 2 \times 3 = 24$ (multiplicação de níveis envolvidos). Desta forma, existem 24 possíveis combinações para investigação. Os experimentos serão executados em Smartphones com sistema operacional *Android*, os dispositivos são foram Motorola XT1068, Samsung Galaxy SM-G532MT e LG-D392D. Para monitoramento e mensuração do consumo energético será utilizada a ferramenta APPCompare. Dentro da figura 178 temos representada a arquitetura do experimento.

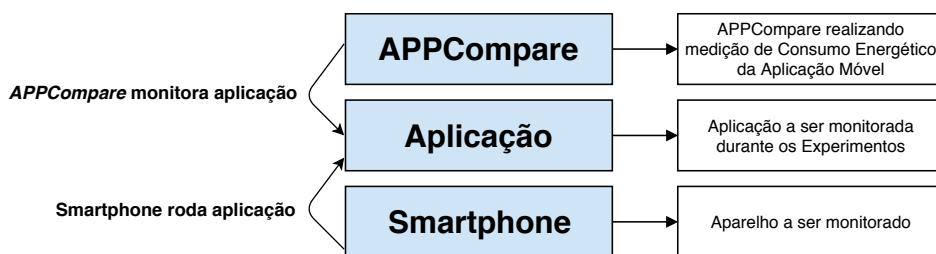


Figura 178 – Arquitetura dos Experimentos

Para esse experimento serão utilizadas três das principais aplicações de *Streaming* de áudio segundo a loja de aplicações do Google (*Google Play*), sendo elas Spotify⁵, Deezer⁶ e SoundCloud⁷. Essas aplicações serão avaliadas em cada um dos Smartphones, onde foram aplicadas todas as combinações para cada aplicação. Para os resultados coletados, métodos estatísticos serão utilizados para comparação e validação de hipótese. E, por fim, será feita uma análise comparativa sobre o consumo geral de cada uma das aplicações monitoradas.

Para um maior rigor dos experimentos realizados, foi tomado o cuidado de interromper a execução de funcionalidades que poderia interferir no consumo energético do dispositivo, como aplicações que sincronizam informações com a rede em tempo real, aplicações do sistema *Android*. Também, para todos os experimentos com utilização de conexão de rede de dados móveis, foi utilizada a mesma rede (Conexão 3G HSDPA).

21.5.1.2 Planejamento DoE

Nesta Seção, uma análise foi executada usando DoE no consumo energético em aplicações de dispositivos móveis. A Tabela 149 apresenta as combinações estudadas durante o DoE. Para cada combinação, 3 observações foram realizadas em cada aplicação, sendo três aplicações por Smartphone, resultando em 24

⁵ Spotify: <https://www.spotify.com/>

⁶ Deezer: <https://www.deezer.com/>

⁷ SoundCloud: <https://soundcloud.com/>

investigações no total. As observações tiveram uma distribuição normal considerando $\alpha = 0.05$, que é 95% de intervalo de confiança.

Tabela 148 – Fatores e seus Níveis

Fatores	Níveis	
Tipo de Conexão	WiFi	Dados Móveis
Qualidade da Reprodução	Baixa	Alta
Nível de Brilho da Tela	Baixo	Alto
Aplicativo	Spotify	Deezer
		SoundCloud

O fator Tipos de Conexão tem como propósito avaliar qual tipos de conexão utilizadas de aparelho pode causar maior consumo. Foram utilizados Rede 3G de Dados Móveis e Conexão sem Fio IEEE 802.11g. O Fator Qualidade de Conexão foi definido para a mensuração da eficiência energético do dispositivo em redes com diferentes qualidades de conexão. Foram utilizados os níveis Baixa (com rede reduzida a somente 50 KBps para o dispositivo) e Alto (com uma rede sem limitações e capaz de alcançar altas taxas de tráfego). O fator Nível de Brilho de Tela tem como objetivo avaliar a interferência do consumo energético da tela dispositivo enquanto exibida a interface da aplicação avaliada. Para os níveis, foram utilizados: Baixo (com o nível de brilho no mínimo operado pelo *display* do aparelho) e Alto (com o nível de brilho no máximo operado pelo *display* do aparelho).

Tabela 149 – Combinações de Experimentos

Tipos de Conexão	Qualidade da Conexão	Nível de Brilho da Tela	Aplicativo
Dados Móveis	Baixa	Baixo	Spotify
Dados Móveis	Baixa	Baixo	Deezer
Dados Móveis	Baixa	Baixo	SoundCloud
Dados Móveis	Baixa	Alto	Spotify
Dados Móveis	Baixa	Alto	Deezer
Dados Móveis	Baixa	Alto	SoundCloud
Dados Móveis	Alta	Baixo	Spotify
Dados Móveis	Alta	Baixo	Deezer
Dados Móveis	Alta	Baixo	SoundCloud
Dados Móveis	Alta	Alto	Spotify
Dados Móveis	Alta	Alto	Deezer
Dados Móveis	Alta	Alto	SoundCloud
WiFi	Baixa	Baixo	Spotify
WiFi	Baixa	Baixo	Deezer
WiFi	Baixa	Baixo	SoundCloud
WiFi	Baixa	Alto	Spotify
WiFi	Baixa	Alto	Deezer
WiFi	Baixa	Alto	SoundCloud
WiFi	Alta	Baixo	Spotify
WiFi	Alta	Baixo	Deezer
WiFi	Alta	Baixo	SoundCloud
WiFi	Alta	Alto	Spotify
WiFi	Alta	Alto	Deezer
WiFi	Alta	Alto	SoundCloud

21.5.1.3 Resultados

As Figuras 179, 180, 181, 182 e 183 apresentam os resultados obtidos através do monitoramento da função de reprodução de áudio através de cada uma das aplicações, utilizando a ferramenta APPCompare. Como

pode ser observado, as Figuras 179 e 180 apresentam Gráficos Pareto de efeitos principais para cada uma das aplicações, em seus determinados aparelhos. Esse gráficos tem como objetivo identificar quais fatores alcançaram o maior impacto no consumo energético do dispositivo. As figuras 182 e 183 apresentam gráficos de efeitos para cada um dos dispositivos. O gráfico de efeitos principais permite analisar o comportamento dos efeitos principais de um fator aplicado ao processo e/ou produto.

A Figura 179 mostra os efeitos para cada fator em cada aplicativo, no dispositivo Moto G2. Nos aplicativos Spotify e Deezer, o fator mais impactante é o Nível de Brilho da Tela, enquanto que na aplicação SoundCloud, o fator que empenha maior impacto sobre o consumo energético total da aplicação durante a monitoração é a qualidade de conexão.

A Figura 180 apresenta os efeitos para os fatores para o dispositivo Galaxy SM-G532MT. Para o aplicativo Spotify, o que obteve o maior impacto foi a combinação de fatores Qualidade de Conexão e Nível de Brilho da Tela, seguido pelo fator Nível de Brilho da tela. Para o aplicativo Deezer, o fator com maior relevância foi a Qualidade de Conexão. Para o SoundCloud, o fator com maior impacto foi a Qualidade de Conexão.

O dispositivo comentado acima trás a configuração de hardware mais simples (ver Tabela 146, o que pode ter sido o motivo de um comportamento diferenciado dos demais aparelhos.

Figuras 182 e 183 trazem gráficos de efeitos principais para cada um dos dispositivos em que foi feito o experimento. Foi realizado um DoE para cada aparelho, a fim de verificar os efeitos da alteração de cada fator em Smartphones distintos.

A Figura 182 trás os efeitos principais de cada fator no dispositivo Moto G2. No geral, este aparelho se apresentou com um consumo energético relativamente baixo, não ultrapassando os 14 Joules em todos os fatores. Apenas os fatores Tipos de Conexão, Nível de Brilho de Tela apresentaram diferença relevante de consumo energético com a variação de seus níveis. Somente o fator Qualidade de Conexão não apresentou diferença relevante durante a variação de seus níveis na execução do experimento.

Se a Figura 185 for observada, podem ser confirmada algumas das informações citadas anteriormente, como, por exemplo, fatores como Tipos de Conexão e Qualidade da Conexão não tem interação com o Nível de Brilho da Tela, assim, eles não interferem entre si no consumo energético. Pode-se notar também que, neste dispositivo, Qualidade da Conexão, quando em nível "Baixa", o aplicativo Spotify apresenta menor consumo energético, seguido pelo aplicativo SoundCloud, com o maior consumo. Para uma Alta qualidade de conexão, o aplicativo Deezer se mostrou mais eficiente, enquanto, inversamente ao caso anterior, o aplicativo Spotify apresentou menos eficiência energética.

A Figura 183 revela informações sobre os efeitos principais de cada fator no dispositivo Samsung Galaxy SM-G532MT. Dentre todos os fatores, o único que apresentou diferença relevante durante sua alteração, foi Aplicativo. Em ordem crescente, os aplicativos que apresentaram menor consumo energético foram Spotify, Deezer e SoundCloud. Este é, dentre todos os dispositivos avaliados, o que possui a melhor configuração de hardware, o que pode justificar o comportamento mais eficiente. Com um melhor poder de processamento (ver Tabela 146) e tudo que se refere à hardware do dispositivo é mais eficaz, no que se refere ao consumo de energia.

Na Figura 186 é mostrado o gráfico de interação dos fatores para o dispositivo Samsung Galaxy SM-G532MT. Neste aparelho, que possui configurações de hardware diferentes do dispositivo comentado anteriormente, houve uma maior interação entre os fatores. Neste dispositivo o fator Tipos de Conexão somente apresentou interação com Qualidade da Conexão e Nível de Brilho da Tela, apresentando maior consumo para o uso dados móveis, em ambas as qualidades avaliadas. O fator Qualidade da Conexão apresenta interação significante apenas com o fator Nível de Brilho da Tela. Portanto, se usada uma alta qualidade de conexão, um alto brilho de tela estará associado a um maior consumo energético, enquanto que, se usada uma baixa qualidade de conexão, um alto nível de brilho da tela resultará em um menor consumo energético.

As Tabelas 150 e 151 apresentam um resumo dos resultados obtidos, organizados por aplicativos com melhor desempenho em relação aos outros fatores e combinações ótimas de fatores para uma maior eficiência energética do dispositivo, respectivamente.

Abaixo serão criados cenários para um melhor entendimento dos resultados obtidos, além de fornecer esclarecimento onde a utilização das combinações ótimas dos fatores pode ser utilizadas, e para que tipos de usuário.

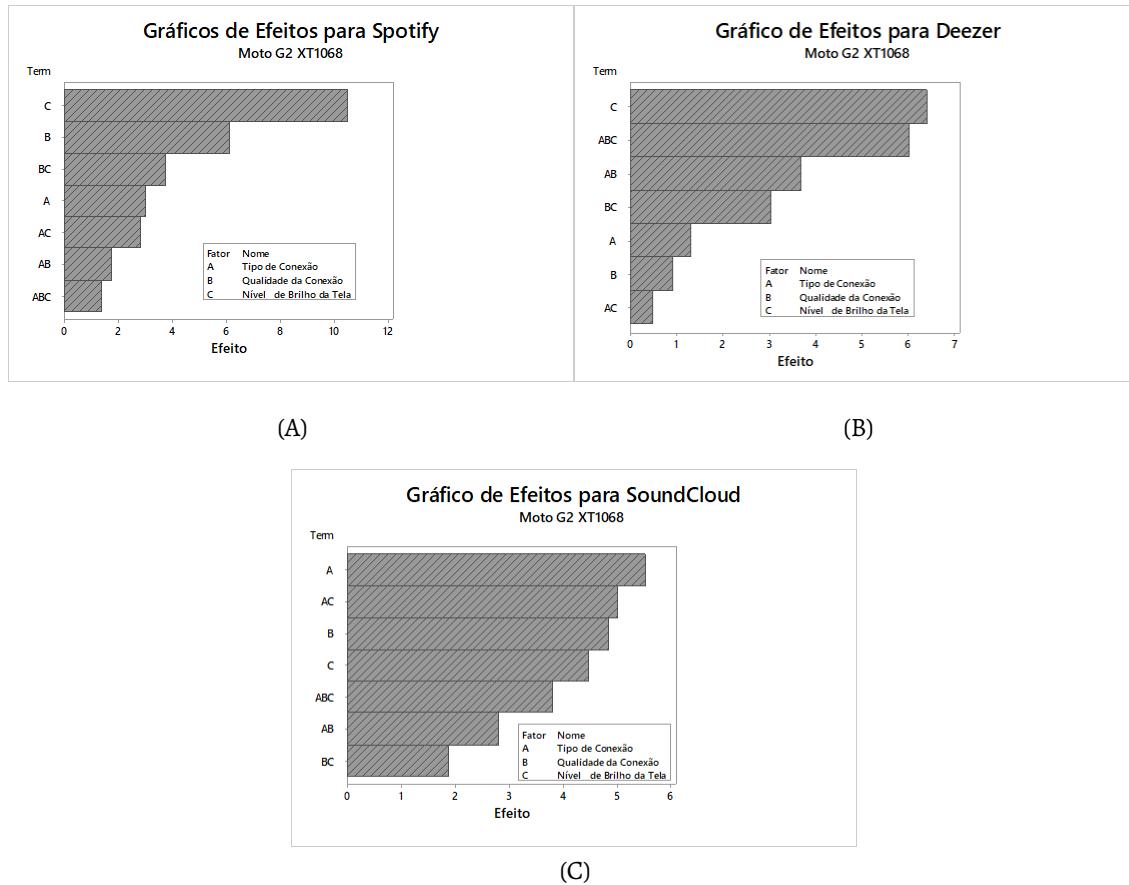


Figura 179 – Gráficos de Efeitos para Spotify (A), Deezer (B) e SoundCloud (C) no dispositivo MOTO G2 XT1068

Tabela 150 – Aplicativos mais eficientes em cada dispositivo e fatores

Fatores	Dispositivos		
	Moto G2	Sansumg Galaxy	LG-D392D
Brilho	Spotify	Spotify	SoundCloud
Qualidade de Conexão	Spotify	Spotify	SoundCloud
Tipo de Conexão	Spotify	Spotify	SoundCloud

- Cenário 01:** Um usuário deseja utilizar um dispositivo que tenha uma CPU Quad-Core de 1,2 Ghz, com um 1 GB de memória RAM, com tela de resolução 720x1280 pixels (Ver Tabelas 146 e 147), como, por exemplo, o Motorola Moto G2. Este usuário deseja utilizar o Deezer como sua aplicação de *streaming* de áudio e Dados Móveis como Tipos de Conexão utilizada. Caso o usuário em questão leia este trabalho, é aconselhado, de acordo com seu dispositivo e os resultados obtidos pelo presente trabalho, é aconselhável que o mesmo utilize WiFi como Tipo de Conexão, além de alterar sua aplicação de *streaming* para Spotify. Outras informação necessária para um menor consumo energético do dispositivo, é a utilização do nível de Brilho da Tela no mínimo. Uma Qualidade de Conexão mais baixa também pode acarretar em um consumo energético menor aos dispositivos.

- Cenário 02:** Um usuário deseja utilizar um dispositivo que tenha uma CPU Quad-core de 1,4 GHz,

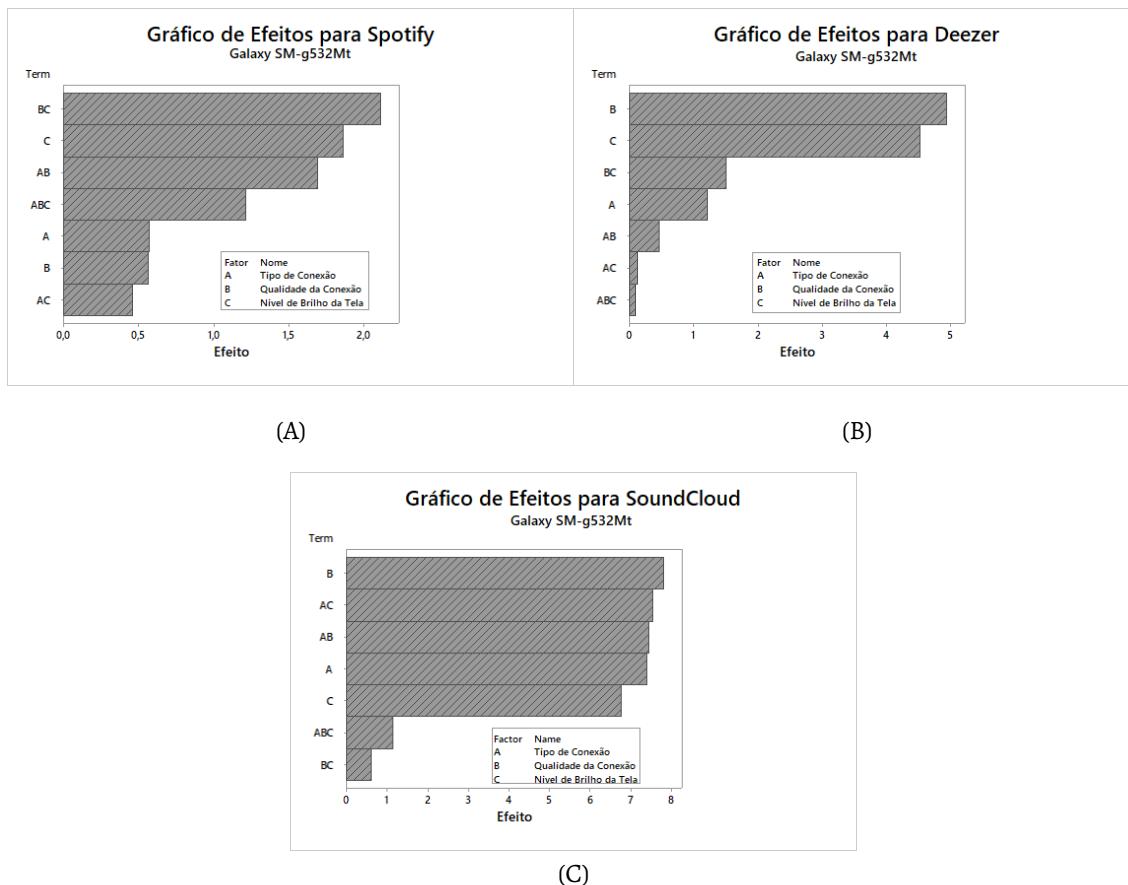


Figura 180 – Gráficos de Efeitos para Spotify (A), Deezer (B) e SoundCloud (C) no dispositivo SANSUMG GALAXY SM-G532MT

Tabela 151 – Resumo das Informações Obtidas: Combinações Ótimas dos Níveis de cada Fator em cada Dispositivo

Fatores	Dispositivos		
	Moto G2	Sansumg Galaxy	LG-D392D
Bilho	Baixo	Baixo	Baixo
Qualidade de Conexão	Baixa	Baixa	Não Interfere
Tipo de Conexão	WiFi	WiFi	WiFi
Aplicativo	Spotify	Spotify	SoundCloud

com 1,4 GB de Memória RAM, com tela de resolução de 540x960 pixels (Ver Tabelas 146 e 147), como, por exemplo, o Sansumg Galaxy SM-g532MT. De acordo com a Tabela 150, em todos os fatores, o Aplicativo Spotify se mostrou mais eficaz energeticamente que os outros, sendo a melhor escolha para este usuário. A Tabela 151 mostra que caso usuário utilize O Brilho do dispositivo no nível mais baixo, utilizar WiFi e até Conexão com Baixas qualidades, juntamente com o aplicativo Spotify, a combinação ótima de fatores seria alcançada, onde haveria o menor consumo de energia para o aparelho em questão.

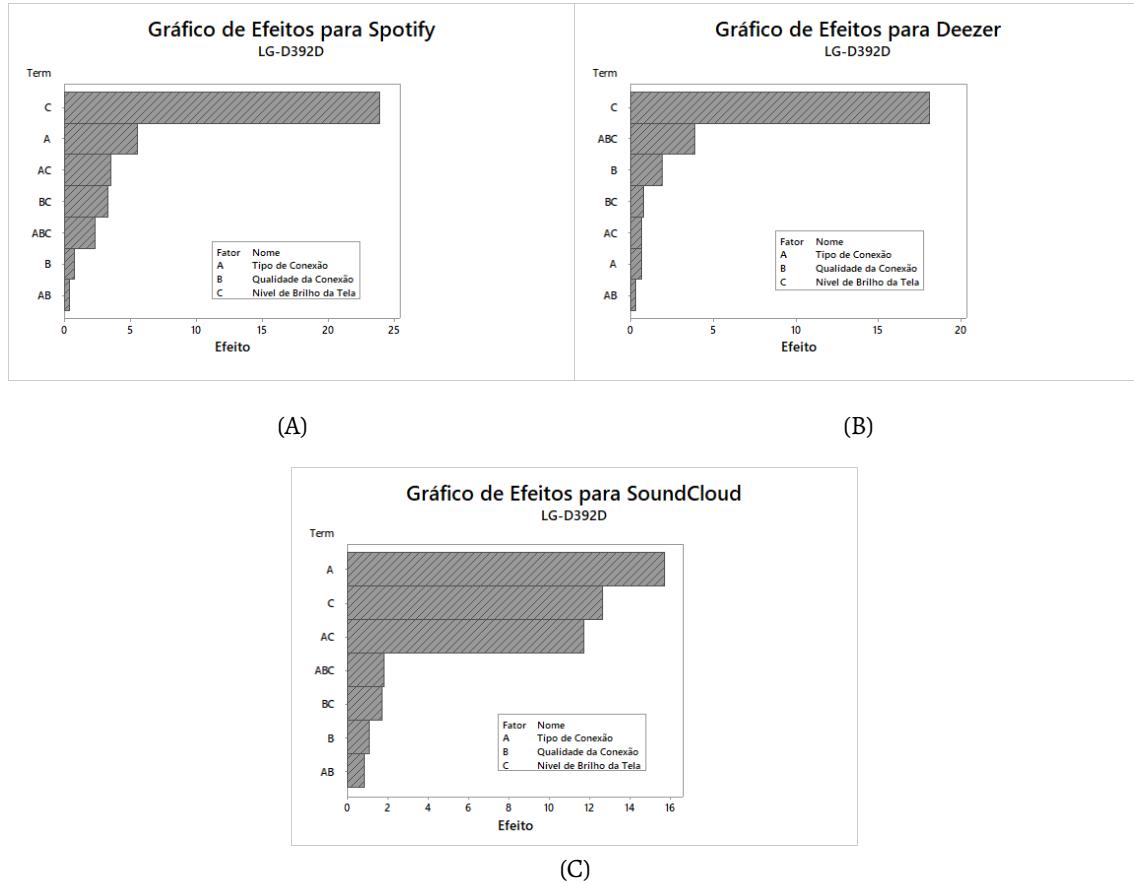


Figura 181 – Gráficos de Efeitos para Spotify (A), Deezer (B) e SoundCloud (C) no dispositivo LG D392D

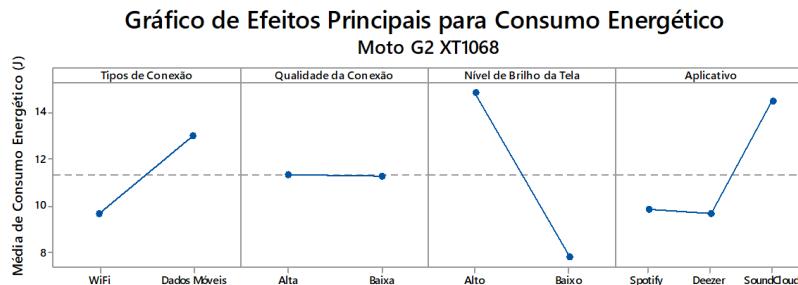


Figura 182 – Gráficos de Efeitos Principais para MOTOROLA MOTO G2 XT1068

21.5.1.4 Conclusões do Estudo de Caso

Nesse estudo de caso foi apresentada uma análise de sensibilidade nos fatores que podem impactar o consumo energético de aplicações de *streaming* de áudio. Este estudo foi executado utilizando-se da ferramenta proposta e desenvolvida por este trabalho, o APPCompare. Utilizando a metodologia DoE, provou-se, matematicamente, que uma conexão de Dados Móveis, aliada à uma Alta qualidade de reprodução de conteúdo e alto Brilho de Tela incide em um maior impacto no consumo energético dos *Smartphones*. A ordem de precedência de impacto no consumo energético foi: 1º) Tipo de Conexão, 2º) Qualidade da Reprodução, 3º) Nível de Brilho da Tela. Dentre todos os fatores, todas as combinações e todas os dispositivos monitorados,o

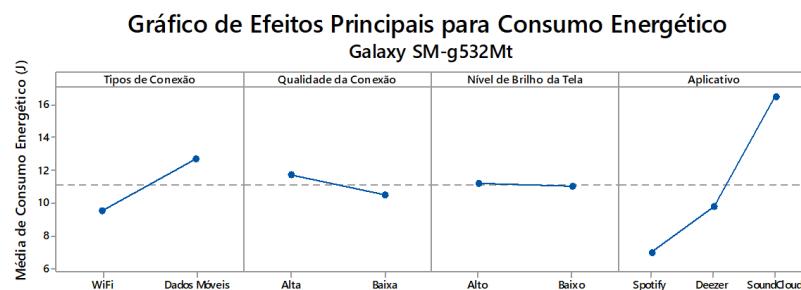


Figura 183 – Gráficos de Efeitos Principais para SAMSUNG SM-G532MT

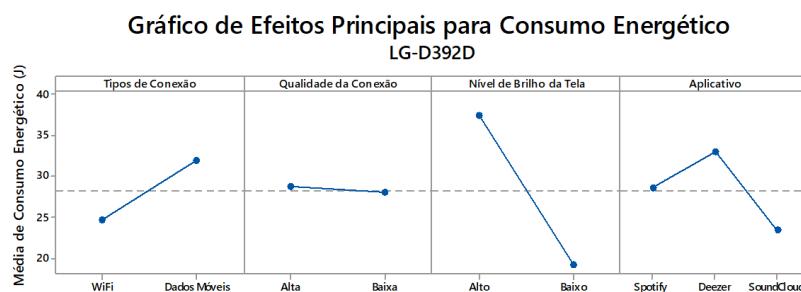


Figura 184 – Gráficos de Efeitos Principais para LG D392D

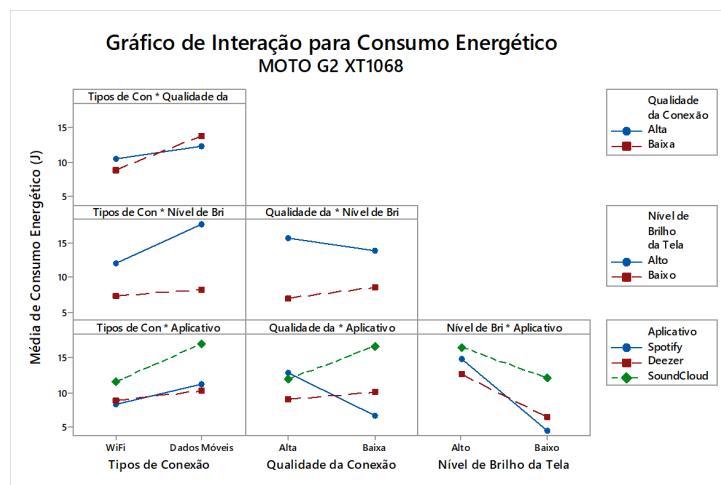


Figura 185 – Gráficos de Interação para Motorola Moto G2 XT1068

dispositivo Galaxy SM-G532MT apresentou o menor consumo médio, e a configuração Tipos de Conexão: WiFi, Alta Qualidade de Reprodução e e Baixo Brilho de Tela apresentou a menor a eficiência energética dentre as combinações monitoradas. Como trabalhos futuros, pretende-se avaliar a influência da tipos de tecnologia de transmissão (3G, 4G, WiFi A/B/G/N 2.4Ghz e 5Ghz) e Tipos de Arquiteturas de *Hardware* no consumo energético do dispositivo móvel.

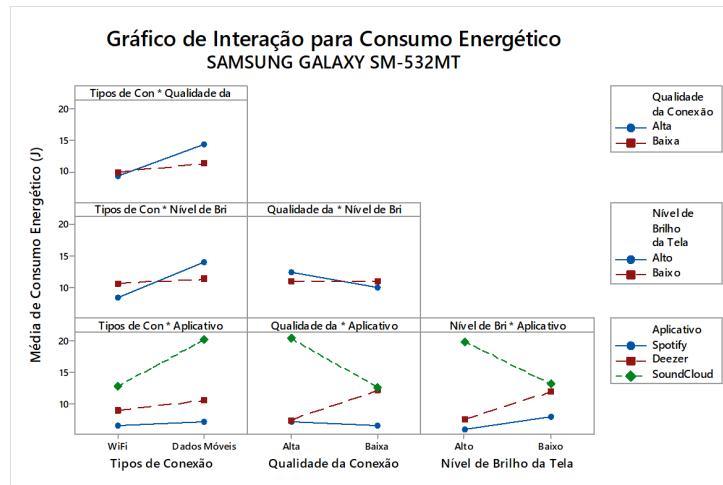


Figura 186 – Gráficos de Interação para Samsung Galaxy SM-532MT

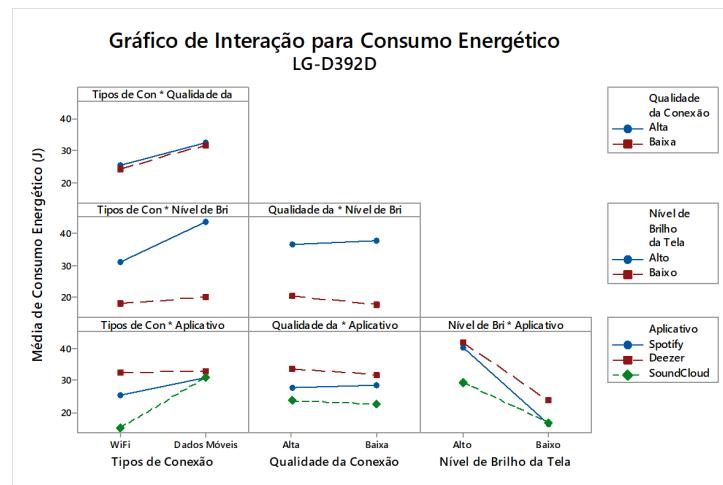


Figura 187 – Gráficos de Interação para LG-D392D

21.5.2 Avaliação de Interferência

Esta Seção tem como objetivo realizar uma avaliação dos principais serviços propostos pela aplicação APPCompare, como monitoração e avaliação do consumo energético, assim como testar a aplicação deste ferramenta em situações reais, se utilizando de comparativos com ferramentas próprias para esta finalidade. Além disso, era desejado avaliar a interferência da execução da aplicação proposta no consumo energético total do dispositivo.

Com o objetivo de avaliar a precisão da medição do consumo energético de aplicações móveis realizado pela aplicação APPCompare, é necessário um teste com um dispositivo físico que realize a mesma tarefa, para que seja feita uma comparação e confirmação da precisão dos resultados apresentados ao usuário da aplicação. Para esta finalidade será utilizado o dispositivo Monitor de Potência de Alta Tensão (*High Voltage Power Monitor - HVPM*), produto da empresa *Monsoon Solutions Inc*⁷. A Figura 188 apresenta a arquitetura utilizada no experimento.

⁷ HIGH VOLTAGE POWER MONITOR, Monsoon Solutions Inc: <https://www.msoon.com/online-store/High-Voltage-Power-Monitor-HVPM-p90002590>

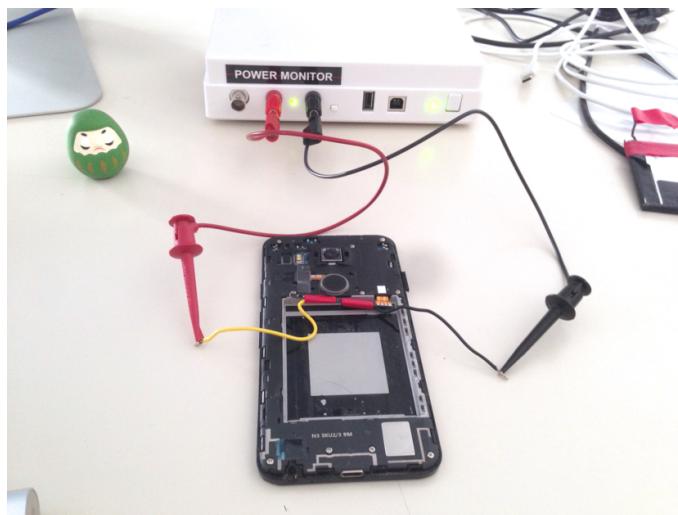


Figura 188 – Arquitetura do Experimento Realizado

Deseja-se, ao final desta avaliação, responder duas perguntas: “Qual a interferência da execução do monitoramento de consumo energético das aplicações no consumo energético do dispositivo?” e “Qual a precisão do resultado obtido do consumo energético da aplicação monitorada pela ferramenta APPCompare?”

Para atingir esses objetivos, serão feitos dois tipos de experimentos. O primeiro será a execução de uma aplicação somente com a monitoração do HVPM, e outra com a execução da medição do APPCompare e, também, monitoração do HVPM.

Para a execução deste experimento foram utilizados três dispositivos móveis distintos, sendo eles LeNovo Vibe K5 e LG G3 Beat. As Tabelas 152, 153 e 154 apresentam as especificações utilizadas para a realização do experimento, especificações técnicas sobre os dispositivos utilizados neste experimento e informações sobre suas telas, respectivamente.

Tabela 152 – Especificações dos Experimentos Realizados

Quantidade de Dispositivos Utilizados	3 Dispositivos
Versão do Android Utilizada	5.0 (em dois dispositivos) e 5.1 (em um dispositivo)
Aplicação Monitorada	YouTube
Resolução Utilizada	720p / HD
Vídeo Utilizado	Link para acesso: https://youtu.be/vRN4pvhOznI
Quantidade de Observações coletadas pelo APPCompare	30 observações
Tempo de Stand-By para Início do Monitoramento	15 segundos

Tabela 153 – Informações Técnicas sobre os Dispositivos Utilizados

Marca	Modelo	Versão Android	CPU	GPU	Memória RAM	Memória ROM
LG	G3 Beat	5.0	1.2 GHz Quad Core	Adreno 305	1 GB	8 GB
LeNovo	Vibe K5	5.1	Quad-core 1.5 GHz Cortex-A53 + Quad-core 1.2 GHz Cortex-A53	Adreno 405	2 GB	16 GB

Tabela 154 – Informações Técnicas sobre a Tela dos Dispositivos Utilizados

Marca	Modelo	Resolução	Densidade de Pixels	Tipo	Cores
LG	G3 Beat	720 x 1280 pixel	294 ppi	IPS LCD	16 milhões
LeNovo	Vibe K5	1080 x 1920 pixel	441 ppi	IPS LCD	16 milhões

Para esta avaliação eram esperados resultados que comprovem o grau de interferência do APPCompare no consumo energético do aparelho monitorado, assim como a precisão dos resultados obtidos através do

monitoramento do consumo energético das aplicações.

Foram executados 10 experimentos em cada dispositivo, e a Tabela 155 apresenta as médias dos resultados obtidos através da utilização do HVPM e APPCompare.

Tabela 155 – Resultados obtidos com o *Monsoon Power Monitor* e APPCompare (Comparativo em Tabela).

Celular	Consumo sem APPCompare (Joules)	Consumo com APPCompare (Joules)	Consumo Obtido pelo APPCompare (Joules)
LeNovo Vibe k5	1,57	1,71	2,0007
LG G3 Beat	1,83	2,04	3,87

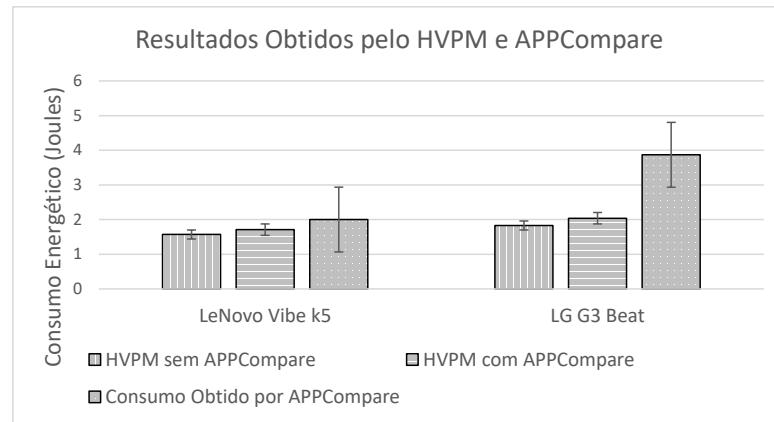


Figura 189 – Resultados obtidos com o *Monsoon Power Monitor* e APPCompare (Comparativo em Gráfico).

Ao observar a Tabela 155 e Figura 189 é possível notar que, observando as diferenças entre consumos energéticos obtidas pelo HVPM e a ferramenta desenvolvida, APPCompare, não é apresentado grande grau de interferência no consumo energético total do dispositivo. Não são apresentadas diferenças relevantes no consumo energético total do dispositivo. Se comparado o consumo obtido pelo APPCompare com o consumo obtido pelo HVPM, é possível notar a baixa diferença entre os valores (0,29 Joules e 1,83 Joules, para os dispositivos LeNovo Vibe K5 e LG K3 Beat, respectivamente). Com os valores obtidos é comprovada a baixa interferência e alta precisão da aplicação executando monitoração do consumo de dispositivos móveis variados.

O Consumo Energético dos dispositivos também foi capturado pelo APPCompare durante todos os testes (mostrado na quarta coluna da Tabela 155). É possível notar que o dispositivo que mais se destacou foi o LeNovo Vibe K5, com somente 2,0007 Joules de consumo energético médio. Esse fato se deve, principalmente, à sua melhor configuração de *hardware*, com melhores processadores e placa de processamento gráfico (GPU - *Graphics Processing Unit*, ou Unidade de Processamento Gráfico) (ver Tabelas 153 e 154). Em seguida, está o dispositivo LG G3 Beat, com consumo médio de 3,87 Joules médio.

21.5.3 Conclusões do Experimento

Foi realizada uma avaliação de interferência e precisão das medições obtidas através do monitoramento do consumo energético da aplicação YouTube em diversos dispositivos móveis. Essa medição foi realizada pela aplicação proposta e desenvolvida neste trabalho, o APPCompare. Um comparativo com uma ferramenta acoplada aos dispositivos utilizados (*High Voltage Power Monitor* (HVPM) da empresa *Monsoon Solutions Inc.*) foi realizado, a fim de obter um contraste para equiparação das informações obtidas. Foi possível notar que

a aplicação desenvolvida pelo presente trabalho apresenta baixa interferência no consumo dos dispositivos avaliados (até 1,83 Joules de interferência), além de, também, obter uma leitura do consumo energético preciso da aplicação à ser monitorada. Esta avaliação provou a aplicabilidade da ferramenta em estudos sobre consumo energético de aplicações específicas, assim como dispositivos móveis em geral.

21.6 Conclusão

Este trabalho desenvolveu o protótipo de um sistema para monitoração de consumo energético para aplicações instaladas no dispositivo móvel (*Smartphones* e *Tablets*) denominado *APPCompare*. A ferramenta desenvolvida tem como objetivo auxiliar no estudo de consumo energético de Dispositivos e Aplicações móveis. A aplicabilidade deste sistema foi analisada com experimentos em diversos aplicativos, e em vários dispositivos. Por meio das análises realizadas, foi possível avaliar a aplicabilidade desta ferramenta dentro do meio científico, no campo da avaliação de desempenho de sistemas.

A aplicação desenvolvida permite a execução e monitoração do consumo energético de um ou mais aplicativos, realizar um comparativo estatístico dos resultados obtidos pelo monitoramento, além de permitir o envio de todas as informações coletadas para servidor remoto. Essa combinação de *features* pode permitir, por exemplo, todo o povoamento de bases de dados com informações sobre estudos de aplicações ou dispositivos em específico. Dentre outras contribuições deste trabalho, estão a realização de arquivos de *log* para *backup* local das informações obtidas, possibilidade de configuração de experimento, como escolha da quantidade observações a serem coletadas, tempo de *stand-by* para início do teste e escolha de níveis de confianças para as avaliações estatísticas a serem utilizadas pelo métodos paramétricos que utilizados no comparativo das aplicações à serem monitoradas e avaliadas.

A Ferramenta *APPCompare* foi aplicada em testes práticos a fim de obter informações sobre sua usabilidade, funcionalidade e precisão dos resultados em testes reais. A precisão da monitoração e medição do consumo energético das aplicações monitoradas foi comprovadas através destes testes. Com base nos resultados obtidos, o modelo de consumo energético utilizado, aliado à uma precisa avaliação estatística, qualifica a aplicação para ser utilizada em ambientes reais e pesquisas que se façam necessários estudos sobre comportamento energético de aplicações ou dispositivos móveis.

Dentre algumas limitações deste trabalho, estavam a disponibilidade de uma quantidade maior de dispositivos à serem avaliados, uma vez que, uma maior quantidade apresentaria uma consistência maior ao comparativo realizado.

Para trabalhos futuros, pretende-se aprimorar a precisão dos resultados obtidos, com melhorias no modelo de consumo energético utilizado, com o objetivo de tornar as medições de consumo o mais próximas possível do real. Outro ponto à ser melhorado será a adição de monitoração de fatores isolados de hardware do dispositivo. Atualmente o *APPCompare* funciona adquirindo informações sobre os pontos mais relevantes que causam consumo ao dispositivo (WiFi, 3G, GPS, CPU, Tela e Áudio) ao mesmo tempo. Portanto, pretende-se avaliar cada um destes fatores de forma isolada, com o objetivo de identificar o consumo somente em determinadas partes do dispositivo, possibilitando estudos do consumo energético de *hardware* específico, através da monitoração de *software*.

Avaliação de Desempenho de Placas de Rede 10GbE: Uma Estratégia Baseada em Análise de Sensibilidade

Autores: Paulo Henrique e Francisco Airton Silva

Resumo

Com a evolução dos meios de comunicação, cada vez mais dados são transmitidos pelas redes de computadores. Os computadores possuem placas de rede que são componentes eletrônicos obrigatórios para realizar a comunicação entre máquinas. Existe uma variedade de placas com capacidades distintas, desde 100MbE (Megabit Ethernet) até 40MbE, ou mais. A placa de 10GbE é uma das placas mais utilizadas por companhias provedores de Internet. Frameworks de captura e processamento de pacotes de alta velocidade são utilizados, conjuntamente com as placas de rede. Os frameworks são utilizados a fim de processarem grandes quantidades de pacotes, sem descartá-los. No entanto, é difícil saber qual combinação de placa e framework adotar ao se analisar diversos fatores de desempenho. Gerentes de infraestruturas de redes não possuem ao seu alcance estudos, que os auxiliem na tomada de decisão de quais placas e frameworks adotar de acordo com as suas necessidades. O presente trabalho apresenta uma análise de sensibilidade avaliando o impacto de duas marcas de placas de rede com capacidade 10 GbE e dois frameworks de captura de pacotes. O objetivo é indicar em relação a qual métrica cada combinação tem impacto mais significativo. Diferentes combinações destacaram-se em cenários específicos. A placa Chelsio juntamente com o framework Netmap foi capaz de oferecer uma taxa de vazão maior para os pacotes. A placa Chelsio e o framework PF_Ring exigem menos recursos computacionais para processar pacotes menores. Para processar pacotes maiores a placa Intel e o framework Netmap são menos exigentes. Em relação aos Pacotes Descartados todas as combinações apresentaram resultados semelhantes. Diferentes combinações foram indicadas tornando este trabalho uma ferramenta de auxílio para os profissionais de rede.

Palavras-chaves: NIC. DoE. Framework E/S. Pareto. Sensibilidade. Provedor

22.1 Introdução

Um dos produtos do crescimento das redes de computadores foi o surgimento da *Internet*. A Internet é uma rede de computadores que interconecta centenas de milhões de dispositivos de computação ao redor do mundo ([KUROSE; ROSS, 2013](#)). Com o surgimento da *Internet* e a sua consequente popularização, as pessoas podem comunicar-se e compartilhar dados a todo momento.

À medida que a rede se tornava mais popular, a quantidade de conteúdo só aumentava e os serviços oferecidos ficaram mais complexos. Dentre os serviços, destacaram-se sites de pesquisa, comércio eletrônico (*e-commerce*), bancos online (*bankline*), serviços públicos governamentais(*e-governament*), entre vários outros. Toda a comodidade proporcionada por esses serviços online, somente repercutiu no crescimento desenfreado da rede com quantidades cada vez mais expressivas de usuários ([BRITO, 2018](#)).

Com o aumento gradativo de dados transferidos e compartilhados, os provedores de *Internet* precisam se aperfeiçoar a fim de garantir um serviço de boa qualidade. O tráfego global da *Internet* em 2021 será equivalente a 127 vezes o volume de toda a *Internet* global em 2005. Ao redor do mundo, o tráfego da *Internet* chegará a 30 GB *per capita* até 2021, acima dos 10 GB per capita em 2016. O número de dispositivos conectados a redes IP será três vezes maior que a população global em 2021 ([CISCO, 2017](#)).

De acordo com uma pesquisa da ([STATISTA, 2018b](#)) há uma previsão de crescimento do volume de dados do tráfego IP do consumidor mundial até 2021. Em 2021, o tráfego IP global consumido deverá atingir 232.655 petabytes por mês a uma taxa de crescimento anual composta de 24%. O tráfego IP de Internet consumido totalizou 52.678 PB por mês em 2016 e deverá aumentar para 127.152 PB mensais em 2020. O tráfego IP consumido via dados móveis foi de 5.953 PB por mês em 2016. Na América Latina em 2016, o tráfego atingiu 532 petabytes por mês. Espera-se que esse número aumente para 1,65 exabytes até 2021 ([STATISTA, 2018a](#)).

O mercado domiciliar de internet no Brasil é atendido prioritariamente pelas pequenas empresas provedoras de acesso à *Internet*: 94% daquelas com 1 a 9 pessoas trabalhando, e 92% e 91% daquelas com 10 a 19 e 20 a 49, respectivamente (entre os provedores de grande porte, aqueles com 250 a 499 e mais de 500 pessoas trabalhando, 56% e 52%, respectivamente, atendem esse mercado). Nesse sentido, os pequenos provedores desempenham um papel de fundamental importância para o desenvolvimento da *Internet* no país. Os pequenos provedores são importantes no que se refere ao atendimento à população das regiões mais afastadas dos grandes centros urbanos([CETIC.BR, 2016](#)).

A camada de transporte da pilha de protocolos TCP/IP, juntamente com o *backbone*, é onde está o principal "gargalo" das operadoras de telecomunicações do Brasil. O investimento nestas camadas está aquém da demanda e, sem uma estrutura sólida neste sentido, uma provedora não consegue crescer. O desequilíbrio causado por essa falta de estrutura é percebido na degradação dos serviços a longo prazo. Por outro lado, da mesma forma que o investimento é necessário, surgem as dificuldades de capacidade de investimento das empresas ([BRITO, 2016](#)).

Os pequenos e médios provedores cada vez mais precisam ir em busca de equipamentos melhores, na qual apresentam altos custos de aquisição. Portanto encontrar soluções mais baratas e que atendam suas demandas de forma mais eficiente é um dos problemas enfrentados por essas empresas.

As interfaces de rede (NIC) com taxas de dados de 1 GbE e 10 GbE (Gigabit Ethernet) são comuns em sistemas conectados a *Internet*([EMMERICH et al., 2015](#)). Toda infraestrutura de rede deve ser avaliada visando uma adequada prestação de serviço, utilizando técnicas de análise. *Sniffing* é uma técnica voltada ao monitoramento e análise de tráfego em rede. Monitorar e observar o tráfego utilizando técnicas como o *sniffing* é um desafio quando se lida com taxas de transmissão e recepção de dados da ordem de 10 Gb/s ou mais.

A tarefa de monitorar um tráfego elevado com ferramentas de *sniffing* é exigente, sendo necessário aplicar poder computacional significativo para tal tarefa ([MORENO et al., 2015a](#)). Aquisições de placas de rede com transferências 10GbE ou mais se tornam uma solução adotada pelos provedores. As placas de rede apesar de possuirem capacidade para receber mais pacotes, possuem limitações para processar uma grande quantidade de pacotes. Com a limitação para tratar milhões de pacotes, as placas acabam descartando-os, o que prejudica um estudo da rede.

Os frameworks de análise de pacotes em altas velocidades não só capturam pacotes mais rapidamente, como também capturam pacotes com mais eficiência. São exemplos bem conhecidos de frameworks de captura de pacotes: **Netmap** (RIZZO, 2012a), **PF_Ring** (DERI; RING, 2011; ORTEGA, 2015), **DPDK** (INTEL, 2014), entre outros.

Em uma análise de desempenho com placas de rede é importante utilizar um filtro de pacotes de alta velocidade para poder coletar as métricas, com uma perda pequena de informações. A análise do tráfego com perda pequena de pacotes, é importante na compreensão de todos os aspectos e peculiaridades do tráfego. Assim, os administradores de rede podem tomar decisões com mais confiança, pois terão mais informações ao seu dispor.

22.1.1 Definição do Problema

Apesar de serem tecnologias importantes na medição de qualidade dos serviços de rede, há dificuldades em encontrar estudos que avaliam essas ferramentas. Pequenos provedores precisam dispor de placas de rede 10GbE para suportar os seus respectivos tráfegos e o uso de frameworks é importante no processo de análise e monitoramento. Nenhum trabalho pesquisado até o momento realizou uma avaliação de quais fatores possuem mais impacto no tráfego de pacotes de uma rede. Os poucos trabalhos que realizam avaliações isoladas de cada tecnologia, não consideraram o efeito de múltiplos fatores conjuntamente. Para pequenos e médios provedores, saber quais Placas de Rede e Framework atendem de forma eficiente as suas necessidades é importante. Já que as companhias provedoras não possuem muitos recursos para serem investidos em poder computacional.

22.1.2 Objetivos

22.1.2.1 Objetivo Geral

O objetivo geral deste trabalho é auxiliar gerentes de infraestruturas computacionais na escolha de placas de rede e framework de processamento de tráfego em situações específicas, oferecendo um estudo de desempenho detalhado sobre tais combinações.

22.1.2.2 Objetivos Específicos

1. Identificar o impacto dos fatores no Throughput obtido em cada combinação (Placa e Framework).
2. Identificar o impacto dos fatores no Uso da CPU obtido em cada combinação (Placa e Framework).
3. Identificar o impacto dos fatores no Percentual de Pacotes descartados em cada combinação (Placa e Framework).
4. Identificar as interações entre os fatores nas variáveis respostas (Throughput, CPU e Pacotes Descartados).

22.1.3 Organização do Trabalho

O restante deste trabalho está organizado nas seguintes seções: a Seção 22.2 (Referencial Teórico) apresenta uma visão geral sobre os conteúdos abordados nesse trabalho; a Seção 22.3 (Trabalhos Relacionados) apresenta outros trabalhos que realizam avaliação de desempenho em placas ou frameworks de captura de pacotes; a Seção 22.4 (Desenvolvimento) apresenta o ambiente de testes montado e o funcionamento de seus componentes, bem como a análise de sensibilidade realizada sobre ele e, por fim, a Seção 22.5 (Conclusão) apresenta as considerações finais, a contribuição e trabalhos futuros.

22.2 Referencial Teórico

Nesta seção será apresentado uma visão geral sobre o que são os frameworks de captura de pacotes em altas velocidades e a sua importância. Abordando também a placas de rede ou NIC, as tecnologias utilizadas

no trabalho e as suas principais características. Esta seção ainda trata da análise de sensibilidade e sua importância em várias áreas.

22.2.1 Análise de Pacotes

A análise de tráfego em tempo real permite investigar o fluxo de dados compartilhados entre dispositivos comunicantes em um dado momento. Essa investigação permite identificar uma série de problemas que por vezes não são mostrados pelas aplicações, estações de trabalho ou equipamentos. Alguns exemplos de problemas encontrados são: anomalias na rede, pontos de bloqueio, equipamentos e cabeamentos defeituosos, mensagens de sistema não mostradas pelas aplicações, falhas de segurança, tráfego de vírus na rede, pontos de bloqueio na rede, entre outros. (FILHO, 2013)

Este assunto de análise de pacotes está bastante presente na literatura com muitos trabalhos se destacando (PARK; WILLINGER, 2000; BENSON; AKELLA; MALTZ, 2010; MORENO et al., 2015b; LEE; LEVANTI; KIM, 2014). Medições detalhadas de tráfego são necessárias para avaliar os requisitos de capacidade e projetar de forma eficiente a rede (FRALEIGH et al., 2003). Os tópicos de pesquisa que podem se beneficiar do monitoramento em nível de pacote são:

- Desenvolver modelos de tráfego que permitam os operadores de rede determinar a quantidade de provisionamento de recursos necessária em sua rede (BARAKAT et al., 2002).
- Avaliar os trade-offs entre diferentes níveis de granularidade no roteamento e estudar a dinâmica do tráfego entre os POP's (*Point of Presence*) (MEDINA et al., 2002; BHATTACHARYYA; DIOT; JETCHEVA, 2001).
- Desenvolvimento de algoritmos para detectar anomalias de rede, como ataques de negação de serviço e loops de roteamento (HENGARTNER et al., 2002).
- Estudar o desempenho do TCP e identificar onde o congestionamento está ocorrendo na rede (JAISWAL et al., 2007).
- Avaliação da capacidade da rede de oferecer suporte a novos serviços de valor agregado, como telefonia e qualidade de serviço (QoS) (BOUTREMANS; IANNACCONE; DIOT, 2002).

Uma técnica utilizada no processo de análise de tráfego é o *sniffing*. O *sniffing* trata-se de uma técnica de captura de pacotes de rede. Quando os pacotes são transmitidos da origem para o destino, eles percorrem vários dispositivos intermediários. Um nó cuja NIC está definida no modo promíscuo recebe todas as informações percorridas na rede. Cada NIC possui um endereço físico diferente e exclusivo de outra rede. Quando o pacote chega ao NIC, o endereço de hardware do frame corresponde ao endereço físico que o NIC possui, mas se for definido em modo promíscuo, todos os pacotes chegarão a essa NIC. Quando a NIC aceita pacotes, os pacotes são copiados para a memória do driver e depois passam para o kernel e o kernel passa para o aplicativo de análise (KING, 2002). Um exemplo de topologia em que um programa de *sniffing* é utilizada dentro de um provedor é demonstrada na Figura 190.

Como demonstrado no Figura 190, em geral um dispositivo é destinado como um servidor que recebe o tráfego. Esses servidores são equipados com NIC's de 10GbE ou mais, para suportar a quantidade de pacotes. O servidor de monitoramento está conectado diretamente a um *switch* ou roteador, que encaminha o tráfego para o mesmo. Por fim dentro desse servidor um programa de captura de pacotes envia as informações do tráfego ao administrador de rede do provedor.

22.2.2 Placas de Rede 10GbE

10 Gigabit Ethernet (10GE, 10GbE ou 10 GigE) é um grupo de tecnologias de redes de computadores para transmissão de frames Ethernet com frequência de 10 gigabits por segundo. Foi definida inicialmente pelo

¹ <http://idijournal.com/network-asset-diagram.html>

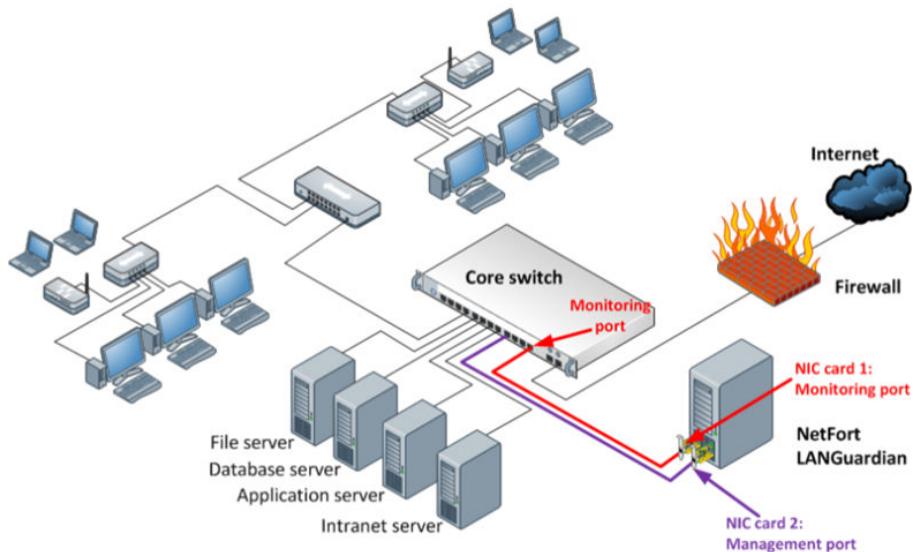


Figura 190 – Exemplo de Topologia de Monitoramento de um Provedor¹.

padrão IEEE 802.3ae-2002. Diferente dos padrões de Ethernet anteriores, no 10 Gigabit Ethernet é definido apenas links ponto-a-ponto full-duplex geralmente conectados por switches de rede (PALMER, 2012).

Para os fins deste trabalho foram utilizadas as placas 10GbE Intel X520-DA2 e Chelsio T520-SO-CR. As placas serão utilizadas nas medições, bem como na avaliação de seu impacto na qualidade do tráfego de pacotes. A placa de rede da Intel é demonstrada na Figura 191 e da fabricante Chelsio na Figura 192.



Figura 191 – Placa de Rede Intel X520-DA2².

O adaptador de rede convergente Intel Ethernet X520 possui conectividade SFP+. A placa X520-DA2 utiliza o Controlador Ethernet 10 Gigabit Intel, denominado Ethernet 82599. O Ethernet 82599 pertence a terceira geração do controlador de rede de 10 GbE da Intel. A placa Intel x520-DA2 foi encontrada com um preço de \$179,99³.

O T520-SO-CR da Chelsio é um adaptador de 10 Gigabit Ethernet Unified Wire, baseado na tecnologia de quinta geração (T5) da Chelsio. Possui porta dupla sem memória com uma interface de barramento de host PCI Express 3.0. A placa T520-SO-CR foi encontrada com um preço de \$219,00⁵

² <https://ark.intel.com/pt-br/products/39776/Intel-Ethernet-Converged-Network-Adapter-X520-DA2>

³ https://www.amazon.com/dp/B0038KVYJO/ref=psdc_13983711_t1_B01IR7SUEK

⁴ <https://www.chelsio.com/wp-content/uploads/2013/10/T520-SO-CR.pdf>

⁵ https://www.amazon.com/dp/BOOFAU898K/ref=emc_b_5_t



Figura 192 – Placa de Rede Chelsio T520-SO-CR ⁴.

22.2.3 Frameworks de Captura de Pacotes

Nesta seção são apresentados os frameworks utilizados na pesquisa e na avaliação de desempenho. É apresentado a arquitetura de cada Framework, assim como o funcionamento dos mesmos. Suas principais estruturas internas são explicadas e exibidas.

22.2.3.1 Netmap

Netmap é um sistema que permite aos aplicativos de leitura e processamento de pacotes realizar um acesso muito rápido aos pacotes na interface de rede, tanto no lado do recebimento quanto do lado da transmissão (RIZZO; LANDI, 2011).

O Netmap foi implementado para vários adaptadores de rede de 1 e 10 Gbit/s. O Netmap utiliza drivers de Placa de Rede e um módulo de kernel modificados em uma tentativa de acelerar a captura de pacotes. O módulo do kernel implementa um buffer de anel. Um buffer de anel é uma memória pré-alocada que, uma vez preenchida, é sobreescrita desde o início de uma forma semelhante a um anel circular. O Netmap tem sua própria API para extrair dados do buffer de anel, na qual inclui uma versão modificada da biblioteca libpcap para suportar aplicativos que utilizam essa biblioteca, como o Wireshark. A Figura 193 ilustra a arquitetura de funcionamento do Netmap.

O framework é construído em torno de uma região de memória compartilhada - acessível aos aplicativos kernel e de espaço de usuário - que contém buffers e descritores para todos os pacotes gerenciados por uma interface, como demonstrado na Figura 193. Os buffers de pacote têm um tamanho fixo, suficiente para armazenar um pacote de tamanho máximo. Isso não implica em fragmentação e em um formato de pacote fixo e simples. Descritores - um por buffer - são extremamente compactos (oito bytes cada) e armazenados em uma matriz circular que mapeia um-para-um para o anel NIC. Eles fazem parte de uma estrutura de dados chamada *netmap_ring*, que também contém alguns campos adicionais, incluindo o índice (*cur*) do primeiro buffer para enviar ou receber, e o número (*avail*) dos buffers disponíveis para transmissão ou recepção. Os buffers e descritores do Netmap são alocados apenas uma vez - quando a interface é ativada - e permanecem ligados à interface (RIZZO, 2012b).

22.2.3.2 PF_Ring

O PF_RING é um framework de distribuição gratuita criando uma pilha circular de cópias de pacotes diretas entre o núcleo do sistema e os aplicativos de leitura de pacotes a nível de usuário (ORTEGA, 2015). Como o Netmap, o PF_RING é uma estrutura que suporta captura de pacotes de até 10 Gbps. O PF_Ring utiliza

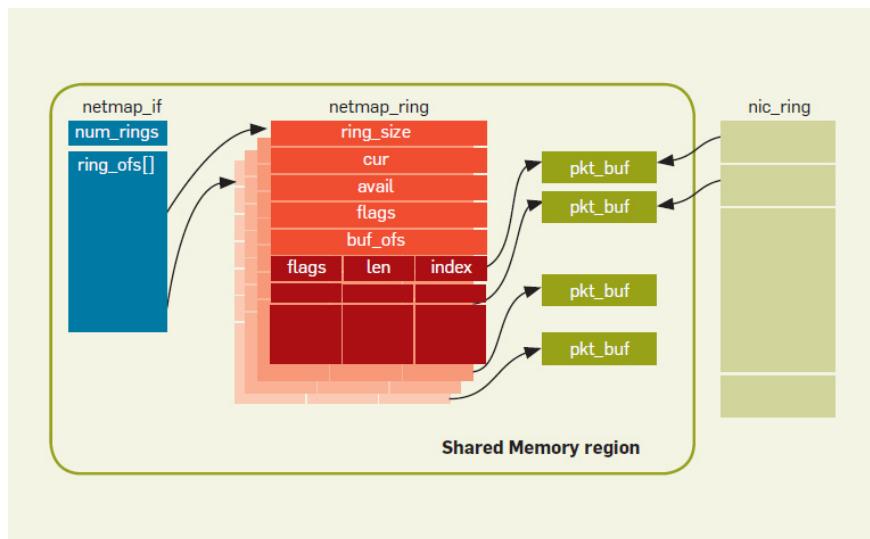


Figura 193 – Estrutura do Framework Netmap

um módulo de kernel e drivers de NIC modificados. Ele também faz uso de um buffer de anel e requer que os aplicativos usem sua API ou uma versão modificada do libpcap. Com a versão modificada da libpcap, programas de captura de pacotes como o Wireshark ou Tcpdump, podem fazer uso do PF_Ring. O PF_Ring e o Netmap por ignorarem o kernel perdem o processamento de pacotes no Netfilter e não há filtragem de pacotes envolvida. O PF_RING, ao contrário do Netmap, oferece solução em Filtragem de Pacotes de Hardware. A arquitetura do PF_Ring é ilustrada na Figura 194.

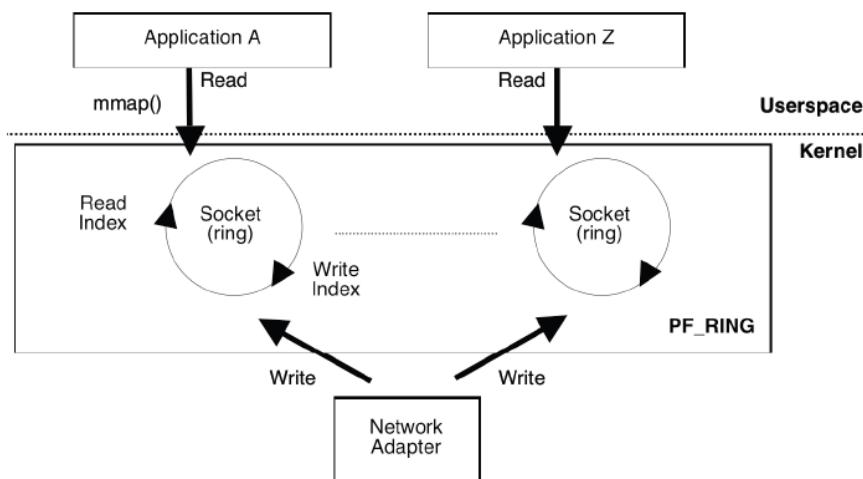


Figura 194 – Arquitetura do Framework PF_Ring

O PF_RING utiliza buffers circulares nos adaptadores de rede para copiar e guardar os pacotes recebidos. O buffer é alocado quando um *socket* do PF_RING é criado, e desalocado quando o *socket* é desativado. Cada *socket* têm seu próprio buffer circular. Se um *socket* PF_RING estiver ligado a um *syscall* adaptado, esse adaptador será utilizado no modo só de leitura até o *socket* ser destruído. Sempre que um pacote é recebido do adaptador (geralmente via DMA, direct memory access), o driver passa o pacote para as camadas superiores. No caso do soquete do PF_RING, todos os pacotes de entrada são copiados para o anel do *socket* ou descartados, se necessário. Se o buffer estiver cheio, o pacote é descartado.

Os pacotes recebidos para adaptadores com *socket* PF_RING limitados, por padrão, não são encaminha-

dos para as camadas superiores, mas são descartados depois de serem copiados para os anéis. Essa prática aumenta o desempenho geral, pois os pacotes não precisam ser manipulados pelas camadas superiores, mas apenas pelo anel. O *buffer* de anel de *socket* é exportado para aplicativos de espaço do usuário via função *mmap()*. Os aplicativos do espaço do usuário que desejam acessar o buffer precisam abrir o arquivo e, em seguida, chamar o *mmap()* para obter um ponteiro para o *buffer* circular. O *kernel* copia pacotes no anel e move o ponteiro de gravação para frente. Os aplicativos do espaço do usuário fazem o mesmo com o ponteiro de leitura. Novos pacotes recebidos sobrescrevem pacotes que foram lidos pelos aplicativos do espaço do usuário. A memória não é alocada ou desalocada por pacotes lidos/gravados no *buffer*, mas é simplesmente sobreescrita (DERI et al., 2004).

22.2.4 Metodologia *Design of Experiments*

O Desing of Experiments (DoE - Projeto de Experimentos) está bem estabelecido e é aplicado com sucesso em vários campos da indústria, como produto para desenvolvimento de projetos, (GUNST, 1996), química (LAZIC, 2006) e engenharia de software (KUHN; REILLY, 2002). Existe um quantidade considerável de livros sobre a metodologia (SIEBERTZ; BEBBER; HOCHKIRCHEN, 2017; SELTMAN, 2012; ANTONY, 2006); Portanto, a metodologia geral aqui será descrita brevemente. Em cada parte da aplicação, será exemplificado cada caso metodológico para o estudo de caso.

O conceito geral de DoE foi criado por uma série de experimentos reais simulados em um sistema ou modelo de sistema sob observação. Em cada experimento, um ou vários parâmetros de projeto são alternados e o impacto no comportamento do sistema é avaliado. Quais parâmetro são alterados e como eles são alterados é definido usando um plano de experimento. O objetivo é obter o máximo de informações possíveis usando a menor quantidade de experimentos, e experimentos com o menor custo em termos de esforço computacional ou setup físico. O comportamento do sistema com base nas mudanças de parâmetros é observado usando conjuntos de saídas. No contexto do DoE, as saídas podem ser referenciadas como “indicadores de performance”, os parâmetros de projeto como “fatores” e os valores das configurações de “Níveis”.

Para analisar o impacto de cada fator no sistema e a interação com os outros fatores, várias combinações de fatores precisam ser testadas e, portanto, exige um experimento. Devido à grande quantidade de combinações possíveis, este é um esforço muitas vezes inviável em termos de tempo e custos. O DoE oferece uma coleção de métodos - referidos como planos experimentais ou tabelas de design - para reduzir a quantidade de experimentos necessários para encontrar informações precisas com o menor número de experimentos possível. O tipo de plano usado depende do objetivo do experimento. Encontrar a tabela de projeto ideal, por exemplo, a menor quantidade de experimentos necessários para descrever o comportamento dos sistemas corretamente, tem sido objeto de intensa pesquisa. Usando um conjunto de métodos estatísticos de avaliação nos dados resultantes do experimento, o impacto, efeitos e interações dos fatores em relação aos indicadores de performance escolhidos é avaliado. Os resultados do experimento podem ser usados para formular um modelo substituto matemático, também chamado de metamodelo em disciplinas de engenharia (MILLER et al., 2014; THOMAS et al., 2014). Neste trabalho foi aplicado a metodologia DoE para avaliação da arquitetura proposta. Três tipos de gráficos são adotados usualmente em estudos com DoE: Gráfico de Pareto, Gráfico de Efeitos Principais e Gráfico de Intereração (ESPEJO, 2006).

22.2.4.1 Gráfico de Pareto

O gráfico de Pareto permite detectar qual o efeito da interação de fatores é mais importante para o processo ou estudo de otimização de projeto com o qual se deve lidar. Exibe os valores absolutos dos efeitos e desenha uma linha de referência no gráfico. Qualquer efeito que ultrapasse essa linha de referência é potencialmente importante. Um gráfico de Pareto é construído como na figura 195, por exemplo. O gráfico mostra que os fatores B (tool geometry) e C (cutting angle) e a interação AC são mais importantes.

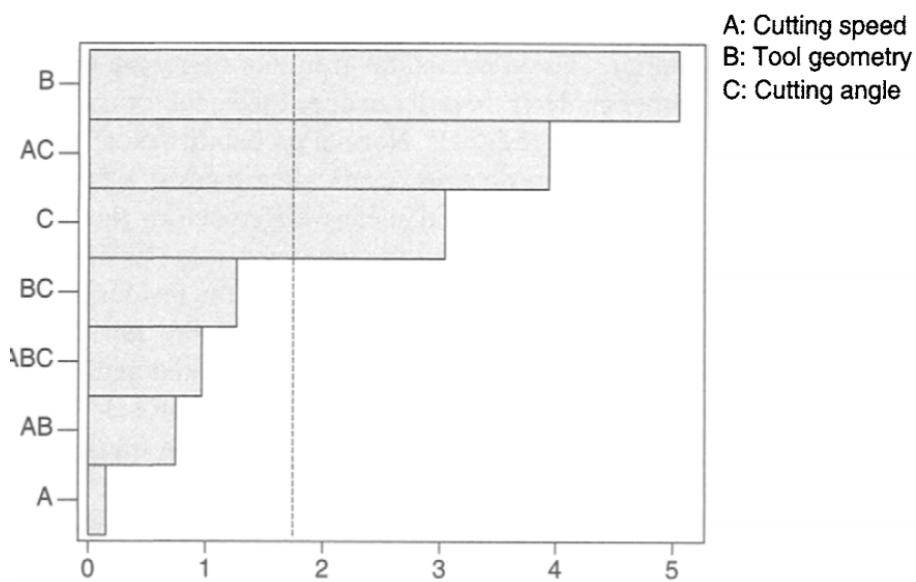


Figura 195 – Gráfico de Pareto com Efeitos Padronizados

22.2.4.2 Gráfico de Efeitos Principais

O “Gráfico de Efeitos Principais” é o gráfico dos valores médios de resposta em cada nível de um parâmetro de projeto ou variável de processo. Pode-se usar esse gráfico para comparar a força relativa dos efeitos de vários fatores. O sinal e a magnitude de um efeito principal nos diriam o seguinte:

- O sinal de um efeito principal nos diz a direção do efeito, ou seja, se o valor médio da resposta aumenta ou diminui.
- A magnitude nos diz a força do efeito.

Se o efeito do parâmetro de projeto ou processo é positivo, isso implica que a resposta média é maior em nível alto do que em nível baixo da configuração do parâmetro. Em contraste, se o efeito for negativo, isso significa que a resposta média em nível baixo de configuração do parâmetro é maior que em nível alto. A figura 196 ilustra o efeito principal da temperatura na resistência à tração de uma amostra de aço. Como pode ser visto na figura, a resistência aumenta quando as configurações da temperatura variam para o nível baixo (ou seja, 1 para 1). O efeito de um parâmetro de projeto ou processo (ou fator) pode ser calculado matematicamente usando a simples equação 22.1:

$$E_f = \bar{F}_{(+1)} - \bar{F}_{(-1)} \quad (22.1)$$

Quando $\bar{F}_{(+1)}$ significa a resposta média na configuração de alto nível de um fator, e $\bar{F}_{(-1)}$ significa a resposta média na configuração de baixo nível de um fator.

22.2.4.3 Gráfico de Interação

O processo de "Interações", identifica efeitos importantes e determina sua magnitude, logo, as interações entre os efeitos são cruciais. As interações ocorrem quando o efeito de um fator depende do nível de outro fator. Uma medida de design sempre aborda vários fatores. Entender como esses fatores interagem em que magnitude permite para escolher a melhor combinação de medidas, revelando combinações de fatores com efeito cumulativo ou degradante. A interação $I_{A,B}$ entre dois fatores A e B pode ser calculada usando a equação 22.2.

$$I_{A,B} = \frac{1}{2}(E_{A,B(+1)} - E_{A,B(-1)}) \quad (22.2)$$

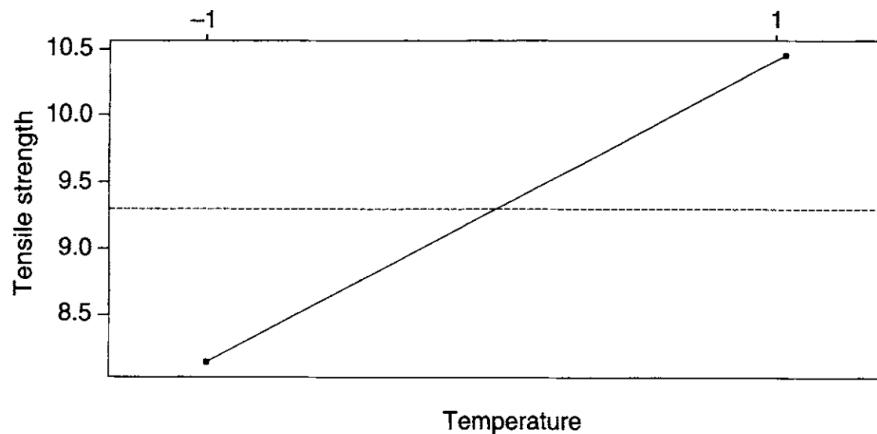


Figura 196 – Gráfico de Efeitos Principais da Temperatura na Resistência à Tração

O $E_{A,B(+1)}$ é o efeito do fator 'A' no nível alto do fator 'B' e $E_{A,B(-1)}$ é o efeito do fator 'A' no nível baixo do fator 'B'.

Para determinar se dois parâmetros de processo estão interagindo ou não, pode-se usar uma ferramenta gráfica simples, porém poderosa, chamada de gráficos de interação. Se as linhas no gráfico de interação forem paralelas, não haverá interação entre os parâmetros do processo. Isso implica que a mudança na resposta média do fator 'A' não depende dos níveis do fator 'B'. Por outro lado, se as linhas não são paralelas, existe uma interação entre os fatores. Quanto maior o grau de afastamento de ser paralelo, mais forte o efeito de interação. Para interação sinérgica, as linhas no gráfico não se cruzam. Por exemplo, a Figura 197(a) é um exemplo de interação sinérgica.

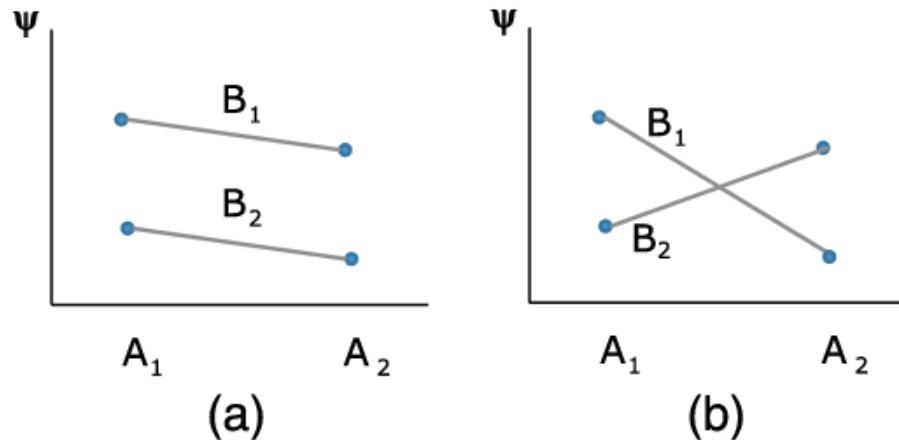


Figura 197 – Gráfico de Interações Sinérgica e Antagônica

Na interação antagônica, as linhas no gráfico se cruzam. Isso pode ser ilustrado na Figura 197(b). Neste caso, a mudança na resposta média para o fator A no nível ' A_1 ' é alta em comparação ao nível ' A_2 '. As mudanças nos níveis do fator 'A' para a resposta média, indica uma dependência do fator 'A' em relação aos níveis do fator 'B'.

22.3 Trabalhos Relacionados

Esta seção apresenta alguns trabalhos produzidos acerca de análise de desempenho de NIC's 10GbE ou frameworks de captura de pacote. Na apresentação dos trabalhos foi destacado as correlações com o trabalho

proposto bem como as diferenças encontradas. Na Tabela 156 são exibidos os critérios de comparação dos trabalhos relacionados com o trabalho atual. Na Tabela 156 é também destacado em quais quesitos os trabalhos pesquisados, foram diferentes do presente trabalho.

Tabela 156 – Trabalhos Relacionados

Trabalho	Análise de Sensibilidade	Métricas Avaliadas	Considerou NIC como fator?	Considerou Framework como fator?
(GALLENMÜLLER, 2014)	Não	Throughput e CPU	Não	Sim
(LARSEN; LEE, 2014)	Não	Throughput e Latência	Sim	Não
(AMMENDOLA et al., 2015)	Não	Latência e Bandwidth	Sim	Não
(BARBETTE; SOLDANI; MATHY, 2015)	Não	Throughput	Não	Sim
(BARKER, 2015)	Não	Perda de Pacote	Não	Sim
(GALLENMÜLLER et al., 2015)	Não	Throughput, CPU e Latência	Não	Sim
(NURIKA et al., 2016)	Não	Throughput	Sim	Não
(KIM; NA, 2017)	Não	Throughput e Perda de Pacotes	Não	Sim
(MARCUZZO; SANTOS, 2017)	Não	Throughput e Latência	Não	Sim
Trabalho Proposto	Sim	Throughput, CPU e Perda de Pacote	Sim	Sim

O primeiro critério de avaliação dos trabalhos foi quais trabalhos fizeram um DoE visando uma análise de sensibilidade dos fatores em relação as métricas utilizadas. Este trabalho foi o único trabalho que realizou a análise de sensibilidade entre os demais trabalhos. O segundo critério foi as métricas utilizadas na avaliação de desempenho. O presente trabalho apresentou mais métricas analisadas em relação a maioria dos demais trabalhos. Por fim os demais critérios foram se os demais trabalhos utilizaram NIC's e Frameworks como fatores. Este trabalho foi o único que utilizou diferentes combinações de NIC e Framework.

No Trabalho de (GALLENMÜLLER, 2014) um modelo é desenvolvido, demonstrando a relação fundamental entre custos e desempenho destes frameworks de processamento de pacotes. Esta aplicação é gradualmente modificada durante esta tese para demonstrar cenários de processamento de pacotes mais complexos. Assim, os fatores que influenciam a taxa de transferência, como a carga da CPU, a conexão da memória ou o número de pacotes processados simultaneamente, podem ser examinados mais de perto.

A semelhança deste do trabalho proposto com o de (GALLENMÜLLER, 2014) é que o objetivo a ser alcançado em ambos é que os insights obtidos podem ser aplicados para realizar aplicações de processamento de pacotes. É escolhido a estrutura que melhor se adapte aos requisitos da aplicação. A diferença principal é que (GALLENMÜLLER, 2014) não leva em conta diferentes placas para realizar a análise, se limitando apenas aos frameworks. O presente trabalho busca realizar uma análise de sensibilidade a fim de identificar com exatidão os fatores mais impactantes na avaliação de desempenho.

Em (LARSEN; LEE, 2014) os autores analisam o estado atual dos avanços da arquitetura de E/S (Entrada e Saída) de alto desempenho e explora os benefícios e as limitações. Com base nas medições e análises, é demonstrado como a E/S é afetada por restrições de latência e throughput. Por fim, os autores sugerem opções para melhorar o desempenho de E/S, com base nessas medidas.

Levando em consideração a parte de análise de (LARSEN; LEE, 2014) percebe-se que as diferenças entre o seu trabalho e este são as métricas analisadas e a ausência de uma análise de sensibilidade. Em (LARSEN; LEE, 2014) os mesmos buscam comparar diferentes fatores em relação a placa de rede, sem a utilização de frameworks de captura de pacotes.

No trabalho de (AMMENDOLA et al., 2015) é apresentado o NaNet-10, uma placa de interface de rede de baixa latência. A NaNet-10 é dedicada a sistemas em tempo real, oferecendo quatro canais de E/S de 10 GbE. Os autores buscam avaliar o NaNet-10 e o NaNet-1, o NaNet-1 trata-se de um modelo semelhante ao NaNet-10, sendo o NaNet-10 uma atualização do NaNet-1.

O trabalho proposto nesta monografia mostra-se bem diferente do trabalho de (AMMENDOLA et al., 2015) na questão das métricas avaliadas e também do tipo de avaliação. (AMMENDOLA et al., 2015) realiza uma análise comparativa dessas placas, levando em consideração as métricas de Latência de Bandwidth. Nenhum framework de captura de pacotes foi analisado, e não foi realizada uma análise referente aos impactos nos fatores.

Em (BARBETTE; SOLDANI; MATHY, 2015) foi analisado e comparado quantitativamente vários frameworks de E/S de pacotes, mostrando sua superioridade ao encaminhamento baseado em kernel. Através de uma combinação de técnicas existentes e melhorias próprias, os autores derivaram princípios gerais modernos para o design dos frameworks processadores de pacotes.

O presente trabalho diferencia-se do trabalho de (BARBETTE; SOLDANI; MATHY, 2015) na questão da realização dos experimentos. Em (BARBETTE; SOLDANI; MATHY, 2015) eles avaliam a métrica de Throughput coletado, sendo que como fator é utilizado os frameworks de captura de pacote com diferentes configurações. Cada configuração é um fator diferente utilizado na medição. O presente trabalho com a adição da placa como fator apresenta mais combinações. Por fim este presente trabalho realiza uma análise de sensibilidade ao contrário de (BARBETTE; SOLDANI; MATHY, 2015), que se restringe a uma análise comparativa.

No trabalho de (BARKER, 2015) é realizada uma análise de desempenho entre diferentes frameworks. A análise realizada pelos autores, busca identificar qual framework apresenta um melhor desempenho na questão do descarte de pacotes. Por fim, no final, é apontado em que situação cada framework se destaca, e qual o melhor cenário que cada um deve ser utilizado.

O presente trabalho difere-se do trabalho de (BARKER, 2015) no tipo de análise realizada. Em (BARKER, 2015) é realizada uma comparação entre os frameworks com base na métrica "pacotes descartados". O trabalho proposto busca uma análise de quais fatores impactam mais as variáveis respostas, sendo realizado um DoE para tal tarefa. Este trabalho utiliza o fator NIC e Framework, ao contrário do trabalho (BARKER, 2015) que utiliza apenas os Framework como fator.

O trabalho de (GALLENMÜLLER et al., 2015) são analisados vários frameworks para o pacote de alto desempenho E/S. É introduzido um modelo para estimar e avaliar o desempenho dessas estruturas de processamento de pacotes. Além disso, os autores analisaram o desempenho dos quadros mais proeminentes baseados em medições representativas em cenários de encaminhamento de pacotes. Por fim, foi analisado o *trade-off* entre o Throughput e a latência.

A diferença principal entre este trabalho e o trabalho de (GALLENMÜLLER et al., 2015) é a análise de sensibilidade dos fatores. Ambos os trabalhos apresentam métricas semelhantes, bem como o uso de frameworks como fatores. Em (GALLENMÜLLER et al., 2015) os autores não realizam medições com diferentes placas de rede como fatores.

O trabalho de (NURIKA et al., 2016) trata-se uma expansão de uma publicação, dos mesmos autores, sobre otimização simultânea de várias placas de rede em uma simulação de data center, usando algoritmo genético. Os autores realizaram diferentes configurações na placa de rede avaliada. Cada configuração da placa de rede é destinada a diferentes características de transmissão de dados. Nas medições realizadas foi registrada a taxa de transferência, comparando-a com a taxa de transferência da configuração padrão da placa de rede (não otimizada).

O trabalho de (NURIKA et al., 2016) busca uma análise comparativa da placa de rede otimizada com a placa em seu estado não otimizado. O presente trabalho mostra-se diferente do trabalho de (NURIKA et al., 2016), pois utiliza diferentes frameworks de captura de pacotes como fator. O trabalho proposto realiza uma análise de sensibilidade com um maior número de métricas a serem analisadas.

O trabalho de (KIM; NA, 2017) fornece um método de implementação de comunicação unidirecional baseado em COTS (*Commercial-Off-The-Shelf Software*). Em seguida, foi verificado a vantagem do método de comunicação unidirecional baseado em COTS. Os autores implementaram uma amostra usando a Intel 82580 NIC e PF_RING. Por fim, foi apresentado uma possibilidade que pode contribuir para a realização da comunicação on-line por meio de experimentos sobre desempenho e confiabilidade.

A principal diferença entre o trabalho proposto e o trabalho de (KIM; NA, 2017) é o uso da métrica de CPU e também o uso de diferentes placas de redes na análise deste trabalho. Em (KIM; NA, 2017) os autores utilizam uma placa de rede 10GbE com diferentes analisadores de pacotes, porém a placa não é usada como fator na pesquisa. Neste trabalho proposto uma análise de sensibilidade dos fatores é realizada e (KIM; NA, 2017) limita-se a uma análise comparativa.

O trabalho de (MARCUZZO; SANTOS, 2017) teve como objetivo apresentar uma visão geral sobre algumas das técnicas de aceleração de pacotes mais utilizadas. Dois frameworks onde essas técnicas são implementadas, tiveram o seu desempenho avaliado. As métricas medidas foram a vazão, latência e utilização de recursos e comparadas com a pilha de rede tradicional do Linux. A avaliação teve como objetivo demonstrar sua eficiência em um cenário com alto consumo de recursos de rede.

A avaliação de desempenho da utilizada em (MARCUZZO; SANTOS, 2017) foi semelhante em relação a este trabalho, no que se refere as métricas utilizadas. O trabalho de (MARCUZZO; SANTOS, 2017) realizou

uma avaliação de desempenho comparativa, utilizando diferentes frameworks como fatores. O presente trabalho se difere, portanto, na utilização de frameworks e placas de rede como fatores na avaliação. Neste trabalho outra diferença notada em relação ao trabalho de (MARCUZZO; SANTOS, 2017) foi o DoE realizado e a análise de sensibilidade dos fatores.

22.4 Análise de Sensibilidade

Esta seção descreve os experimentos realizados para analisar os fatores que tiveram mais influência na acurácia das métricas analisadas através da realização de uma análise de sensibilidade utilizando as métricas: Throughput, Uso de CPU e Perda de Pacotes.

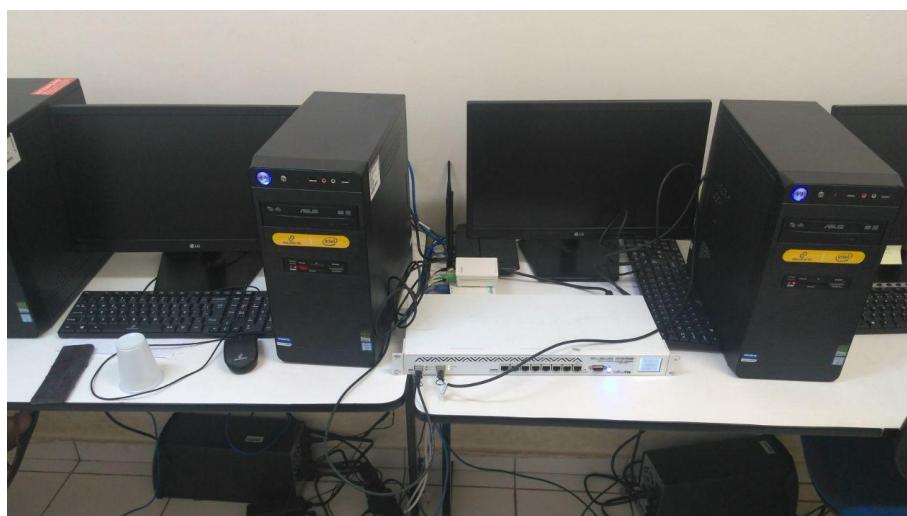
22.4.1 Ambiente de Testes

O ambiente de experimentos foi configurado como ilustra a Figura 198. Foram adotados dois computadores e um aparelho *Mikrotik CCR1036-8G-2S+*⁶. Nos experimentos, o *Mikrotik* — também conhecido como *RouterBoard* — foi responsável pela geração do tráfego. A medida de envio do tráfego é PPS (pacotes por segundo), na qual o dispositivo gerava uma rajada de pacotes a cada segundo. A quantidade de pacotes por segundo de cada rajada individual depende do tamanho do pacote utilizado. Pacotes menores são gerados mais rapidamente em relação aos pacotes de tamanho maior, logo mais pacotes por segundo são gerados e enviados mais rapidamente pela *RouterBoard*.

O uso do *CCR1036-8G-2S+* é justificado pelos vários parâmetros de rede disponíveis em sua ferramenta de geração de tráfego, denominada *Traffic-generator*. Esses parâmetros de rede podem ser utilizadas e modificadas para testes, dando uma maior flexibilidade na hora de montar o tráfego a ser gerado. A *RouterBoard* também possui 2 portas que suportam tráfegos de 10Gb/s e que são compatíveis com o *Plugue sfp+*⁷. O *Plugue sfp+* faz parte cabo óptico utilizado, que foi o meio físico empregado para transmitir as informações.

Em cada um dos computadores foi instalada uma placa de rede diferente. As máquinas de igual configuração possuem Processador *Intel i3* com 8GB de Memória RAM e Disco Rígido de 500GB, rodando o Sistema Operacional *Linux Ubuntu 16.04 LTS*. As placas de rede avaliadas foram especificamente a *Intel X520-DA2* e a *Chelsio T520-SO-CR*. Como mencionado anteriormente, os frameworks de leitura e captura de pacotes *PF_Ring* e *Netmap* foram adotados, dado a grande quantidade de trabalhos que os utilizam.

Figura 198 – Cenário Físico dos Testes



⁶ <https://mikrotik.com/product/CCR1036-8G-2Splus>

⁷ <https://produto.mercadolivre.com.br/MLB-1086368142-sfpcisco-para-sfpmikrotik-cabo-twinax-1-metro-JM>

Para realizar a avaliação de desempenho, 3 métricas foram mensuradas, são elas: *Throughput*, Perda de Pacotes e Uso da CPU. Essas métricas são definidas na RFC 2544, que trata da padronização de testes em dispositivos de rede (BRADNER; MCQUAID, 1999), com exceção da métrica de Uso da CPU. A métrica de Uso da CPU está sendo utilizada, pois foi uma métrica de uso comum nos trabalhos pesquisados.

Na execução de um DoE, o primeiro passo é definir os fatores e níveis do experimento. Para poder realizar a medição das métricas foram definidos três fatores: Placa de Rede, Framework de Captura de Pacotes e Tamanho de Pacote. As métricas de Placa e Framework possuem dois níveis, que são as placas e os frameworks utilizados. A métrica Tamanho de Pacote possui 5 níveis diferentes, a saber: 64 bytes, 128 bytes, 256 bytes, 512 bytes e 768 bytes. As variações de tamanho do pacotes também são definidas na RFC 2544, bem como a utilização do protocolo UDP nos pacotes gerados.

A Tabela 157 apresenta os fatores e níveis escolhidos com o objetivo de identificar o fator com maior impacto em cada métrica coletada. Esses fatores e níveis foram utilizados para cada métrica, sendo que cada métrica analisada teve um DoE realizado.

Tabela 157 – Fatores e Níveis dos DoE's

Fator	Nível I	Nível II	Nível III	Nível IV	Nível V
Placa	Chelsio	Intel	-	-	-
Framework	Netmap	Pf_Ring	-	-	-
Tamanho de Pacote (Bytes)	64	128	256	512	768

O próximo passo é combinar os fatores e níveis para definir as especificações do experimento. A Tabela 158 apresenta as possíveis combinações dos fatores e níveis das métricas analisadas. O desenho do experimento resultou em 20 combinações para cada DoE.

Tabela 158 – Combinação dos Fatores

Nº	Placa	Framework	Tamanho
1	Chelsio	Netmap	64
2	Chelsio	Netmap	128
3	Chelsio	Netmap	256
4	Chelsio	Netmap	512
5	Chelsio	Netmap	768
6	Chelsio	PF_Ring	64
7	Chelsio	PF_Ring	128
8	Chelsio	PF_Ring	256
9	Chelsio	PF_Ring	512
10	Chelsio	PF_Ring	768
11	Intel	Netmap	64
12	Intel	Netmap	128
13	Intel	Netmap	256
14	Intel	Netmap	512
15	Intel	Netmap	768
16	Intel	PF_Ring	64
17	Intel	PF_Ring	128
18	Intel	PF_Ring	256
19	Intel	PF_Ring	512
20	Intel	PF_Ring	768

Para a realização dos experimentos foi utilizado o tempo padrão de 60 segundos, na qual a cada segundo uma rajada de pacotes era enviada e lida. Para cada combinação de níveis de fatores um teste individual de 60 segundos foi executado, coletando 60 medições para a respectiva combinação. Ao fim do teste de 60 segundos, as métricas eram mensuradas em cada combinação. As métricas Throughput e Uso de CPU,

tiveram suas médias calculadas em cada combinação ao fim do teste, conforme a equação 22.3.

$$\bar{X} = \frac{1}{60} \sum_{i=1}^{60} x_i \quad (22.3)$$

Onde \bar{X} é a média final dos valores em cada combinação e x_i a quantidade de pacotes lidos em cada rajada. A métrica Pacotes Descartados foi calculada com base no percentual de perda dos pacotes, que foi calculado conforme as equações 22.4 e 22.5.

$$Dx = \sum_{i=1}^{60} tx_i - \sum_{i=1}^{60} rx_i \quad (22.4)$$

$$P = \frac{Dx \times 100}{\sum_{i=1}^{60} tx_i} \quad (22.5)$$

A expressão 'Dx' é a quantidade de pacotes descartados que é obtido através da diferença total entre tx_i que é a quantidade de pacotes enviados em uma única rajada e rx_i que é a quantidade de pacotes recebidos da rajada enviada. Por fim é calculado o percentual de pacotes descartados, representado pela expressão 'P', entre a quantidade total de pacotes enviados e a perda de pacotes obtida.

As métricas foram coletadas utilizando aplicativos provenientes dos próprios frameworks. No caso, quando o framework Netmap era utilizado nos experimentos, o programa, chamado *pkt-gen*⁸ era utilizado. Através do *pkt-gen*, o Throughput e a quantidade de pacotes eram coletados, da rajada de pacote recebido naquele segundo. O framework PF_Ring possui um programa de leitura de pacotes que foi utilizado para medir as métricas Throughput e Perda de Pacotes, chamado *pfcnt*⁹.

Na RouterBoard foi coletado a quantidade de pacotes enviados, enquanto o número de pacotes recebidos foi coletado pelos programas *pkt-gen* e *pfcnt*. Com a quantidade pacotes enviados e recebidos, foi possível mensurar a quantidade de pacotes descartados. Por fim a métrica de Uso de CPU foi mensurado através de dois *scripts* desenvolvidos para tal finalidade.

22.4.2 Resultados Obtidos

A seguir são descritos os resultados obtidos de acordo com os dados coletados no ambiente de testes. As métricas: Throughput, Uso de CPU e Perda de Pacotes serão analisadas com base nos gráficos gerados. Nesta sessão serão interpretados os resultados e explicar os eventos ocorridos, extraindo as informações pertinentes e descobertas durante os testes.

22.4.2.1 Throughput

A Figura 199 apresenta o gráfico de Pareto para os fatores relacionados ao Throughput. Neste estudo foi observado que os fatores com impactos mais elevados sobre a medição do Throughput foram: Tamanho de Pacote e Framework. Quando o impacto de um fator é dito alto, quer dizer que alterando o nível daquele fator são obtidos valores bem distintos.

O Tamanho do Pacote possui maior impacto dentre todos os fatores, portanto o Tamanho dos Pacotes de um tráfego de dados é determinante na confiabilidade da medição. Dado que se sabe os fatores com maiores impactos, o próximo passo é saber quais níveis tiveram melhores resultados.

A Figura 200 mostra o gráfico de Efeitos Principais para o fator Throughput. O gráfico representa a resposta média para cada nível de fator ligado por uma linha. No gráfico de efeitos principais se a linha está na horizontal, paralelo ao eixo x, indica a ausência de efeito. Quanto maior a inclinação da linha, maior é a magnitude do efeito principal.

Todos os fatores apresentaram efeitos em seus níveis, sendo o fator Placa o que obteve um menor grau de efeito entre seus respectivos níveis. A Placa de Rede da fabricante Chelsio apresentou uma taxa média de

⁸ <https://github.com/luigirizzo/netmap/tree/master/apps/pkt-gen>

⁹ <https://github.com/avis/PF.RING/blob/master/userland/examples/pfcnt.c>

Figura 199 – Influência dos Fatores na Acurácia (Throughput).

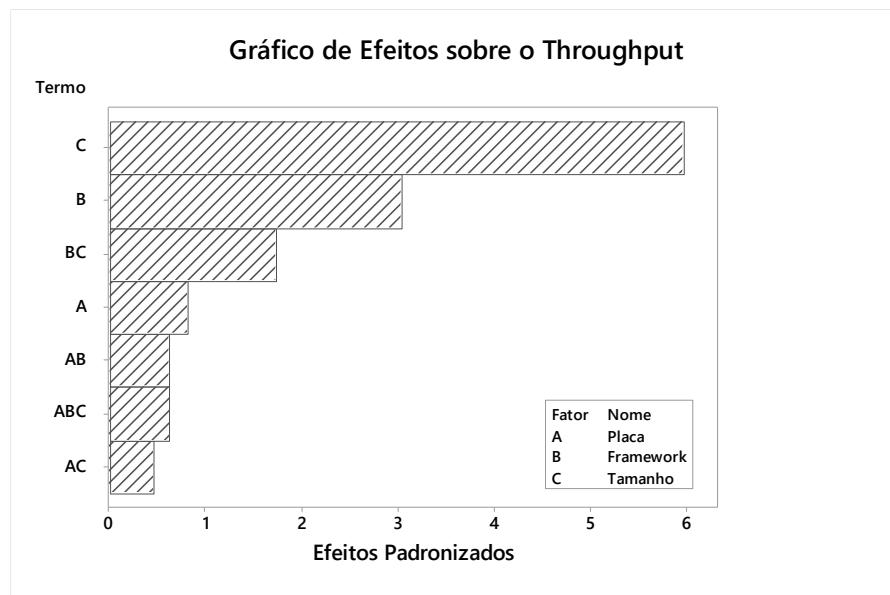
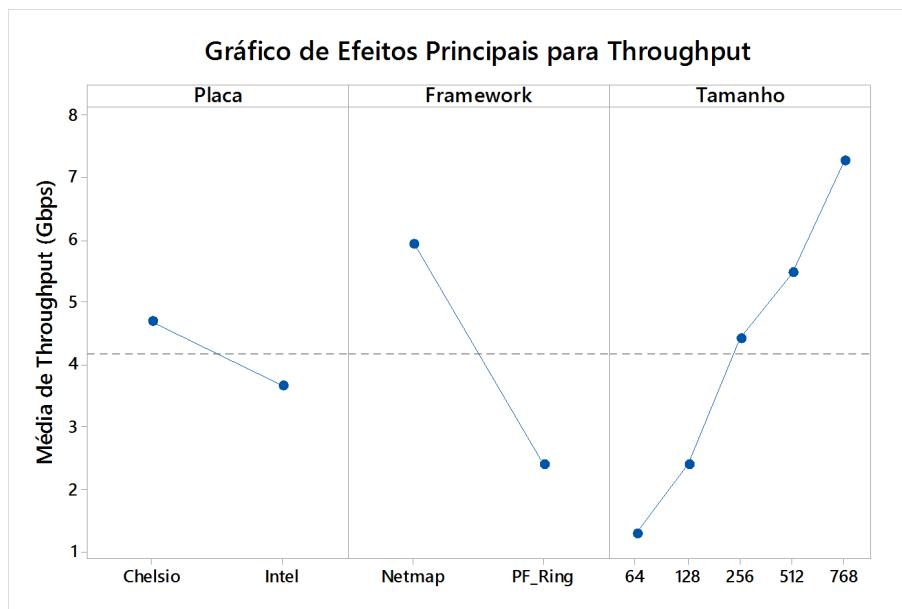


Figura 200 – Taxa Média de cada Nível dos Fatores



Throughput maior. Para atender a um tráfego com as mesmas especificações, a placa da Chelsio consegue atingir uma Taxa de Transferência maior que a placa da Intel.

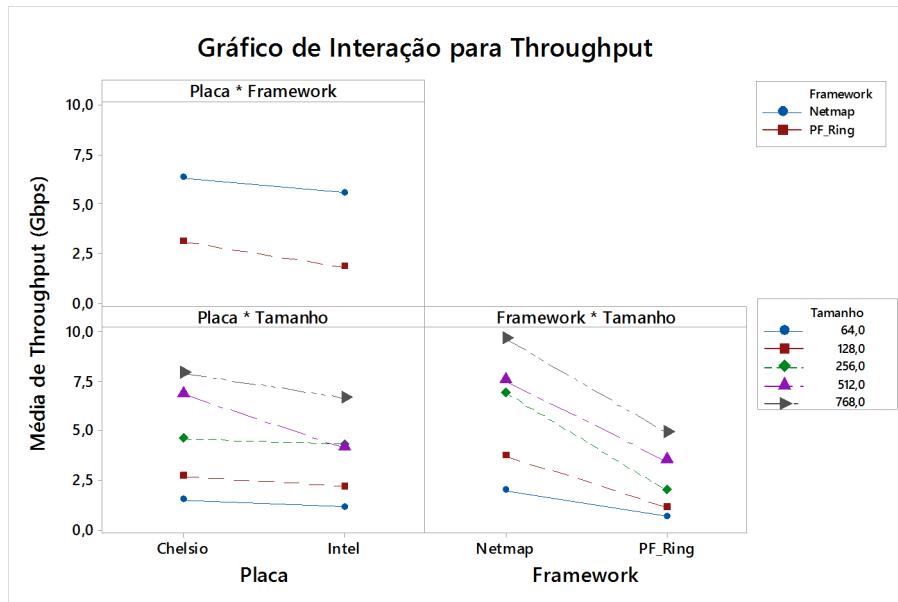
O Framework Netmap apresenta um Throughput médio maior que o Framework PF_Ring, isso significa que o Netmap consegue obter uma maior taxa de transferência média de pacotes em relação ao PF_Ring. Com uma Taxa de Transferência maior, significa que mais pacotes podem ser lidos, o que significa menos pacotes descartados e por consequência um serviço de rede melhor.

Analizando os níveis de Tamanho de Pacote nota-se que quanto maior o tamanho deles, maior o Throughput utilizado. Quanto maior o pacote, mais dados são transportados e portanto maior a vazão ou Throughput necessário para transferi-los.

A fim de saber se a relação entre um fator com a variável resposta medida depende de outro fator categórico, utiliza-se o gráfico de Interações. A Figura 201 mostra os gráficos de interações, para cada

combinação de fatores e suas respectivas interações com a métrica Throughput.

Figura 201 – Interação dos Níveis dos Fatores



Ao observar os gráficos nota-se que a Placa da Chelsio com o Framework Netmap obtêm a maior taxa de Throughput médio. Utilizando também a Placa Intel com o Framework Netmap é obtido um desempenho semelhante, porém inferior.

A placa Chelsio utilizando todos os níveis dos pacotes obteve um Throughput médio maior que a Placa Intel. Nota-se que os níveis de pacote 512 bytes e 768 bytes apresentam um grau de interação maior, logo as placas são sensíveis à variação desses níveis.

Ao analisar a relação entre o Throughput médio e o fator Framework observa-se que há interação de todos os níveis do fator Tamanho de Pacote. Com a interação ocorrendo o Throughput médio dos frameworks é dependente das variações dos níveis do fator Tamanho de Pacote. O Framework Netmap demonstrou uma taxa de Throughput médio superior ao Framework PF_Ring com diferentes combinações de Tamanho de Pacotes.

Portanto, um dispositivo equipado com a placa da Chelsio juntamente com o Framework Netmap tem mais capacidade de leitura de dados e por conseguinte menos pacotes são descartados. Essa combinação se mostra a mais adequada para redes que exigem altas taxas de vazão de dados para manter a qualidade de seus serviços.

22.4.2.2 CPU

A Figura 202 apresenta o gráfico de Pareto para os fatores relacionados ao Uso de CPU. O Tamanho do Pacote possui maior impacto dentre todos os fatores, logo o mesmo é determinante no impacto de consumo de dados da CPU. Os fatores Placa de Rede e Framework não apresentaram significância, no que diz respeito ao impacto ocasionado na acurácia da métrica de Uso da CPU. Assim, como na métrica Throughput, o fator Tamanho de Pacote, continua sendo o fator mais impactante também na medição do percentual de Uso da CPU. Da mesma forma que da análise anterior o próximo passo é saber quais os resultados dos níveis de cada fator.

A Figura 203 mostra o gráfico de Efeitos Principais para o fator Uso de CPU. Como todos os fatores não apresentam linhas de ligação entre os níveis na horizontal, conclui-se que todos os níveis dos fatores afetam a resposta, ou seja, possuem efeito. Os fatores Framework e Tamanho de Pacote, apresentam uma escala de magnitude maior do efeito principal.

Figura 202 – Influência dos Fatores na Acurácia (Uso de CPU).

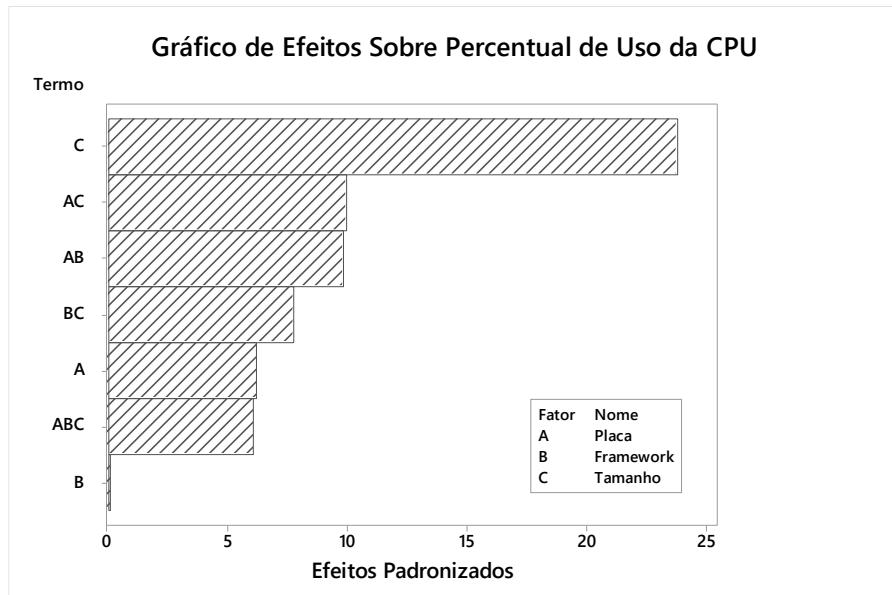
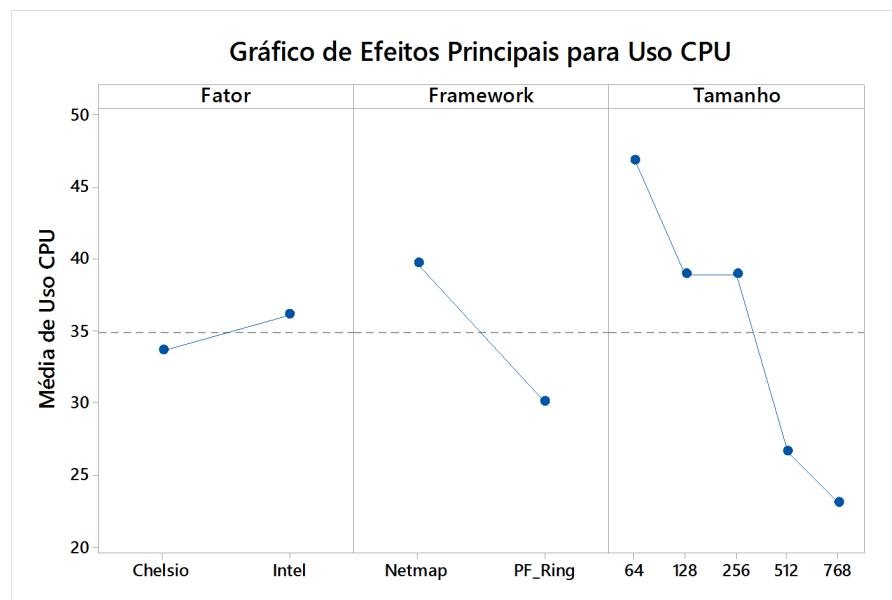


Figura 203 – Percentual Médio de cada Nível dos Fatores

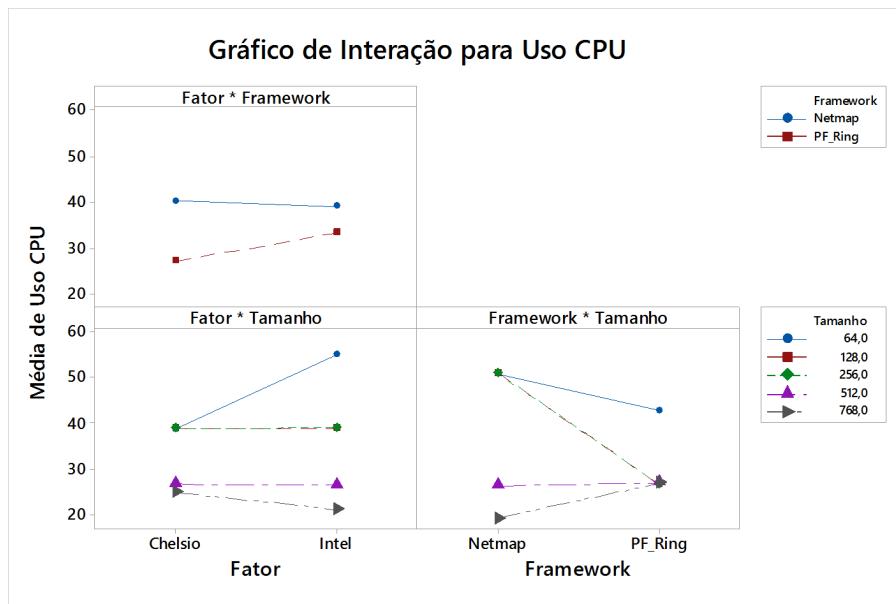


Analizando o gráfico nota-se que as Placas de Rede não apresentaram uma diferença grande de consumo. A Placa de Rede da Chelsio foi ligeiramente melhor, porém não há diferença significativa na variável resposta, em relação a Placa de Rede da Intel. O framework Netmap apresentou um consumo médio de CPU mais alto em relação ao framework PF_Ring. Os pacotes de tamanho menor apresentaram um consumo de CPU elevado, os pacotes de 64 bytes apresentaram uma média maior. Os pacotes de tamanho maior apresentaram baixa exigência de poder computacional. Cada pacote ao ser recebido pela Placa de Rede e lido pelo Framework tem o seu cabeçalho processado. Quanto mais pacotes chegam a interface de Rede mais CPU é exigida para processar o cabeçalho de cada pacote. Pacotes menores por chegarem em quantidades maiores exigem mais poder computacional, consequentemente mais processamento deve ser alocado pela CPU.

A Figura 204 mostra os gráficos de Interação dos níveis de cada fator, permitindo uma análise mais deta-

lhada em relação ao gráfico de efeitos principais. O gráfico de Interação analisa as combinações específicas dos níveis, a fim de chegar a conclusão qual afeta menos a variável resposta.

Figura 204 – Interação dos Níveis dos Fatores



Como analisado no gráfico de efeitos principais, a placa Chelsio apresentou um consumo menor de CPU, juntamente com o framework PF_Ring. A Combinação da Placa da Chelsio com o Framework Netmap, foi a que mais exigiu recursos da CPU. A placa da Chelsio apresentou um consumo de CPU menor em todos os níveis de Tamanho de Pacote, com a exceção do tamanho 768 bytes. A placa Intel, por outro lado exigiu muito processamento com o Tamanho de pacote 64 bytes. Portanto nota-se que a placa da Chelsio apresenta um consumo de CPU mais eficiente com pacotes menores e intermediários, já para pacotes grandes a escolha pela placa da Intel é a mais acertada.

Em relação aos frameworks, o PF_Ring é o que apresenta os melhores resultados para níveis de tamanho inferior: 64 bytes, 128 bytes e 256 bytes. O Framework Netmap, porém, apresentou um percentual médio de uso de CPU menor com o pacote maior: 768 bytes. Portanto o Framework PF_Ring consome menos CPU com pacotes menores, já para pacotes maiores a tendência é o Netmap ser melhor.

Por fim conclui-se para a variável resposta Uso De CPU, as melhores combinações são Placa Chelsio e Framework PF_Ring. Para pacotes grandes a Combinação da Placa Intel e Netmap, apresentou bons resultados e é indicada nesta situação.

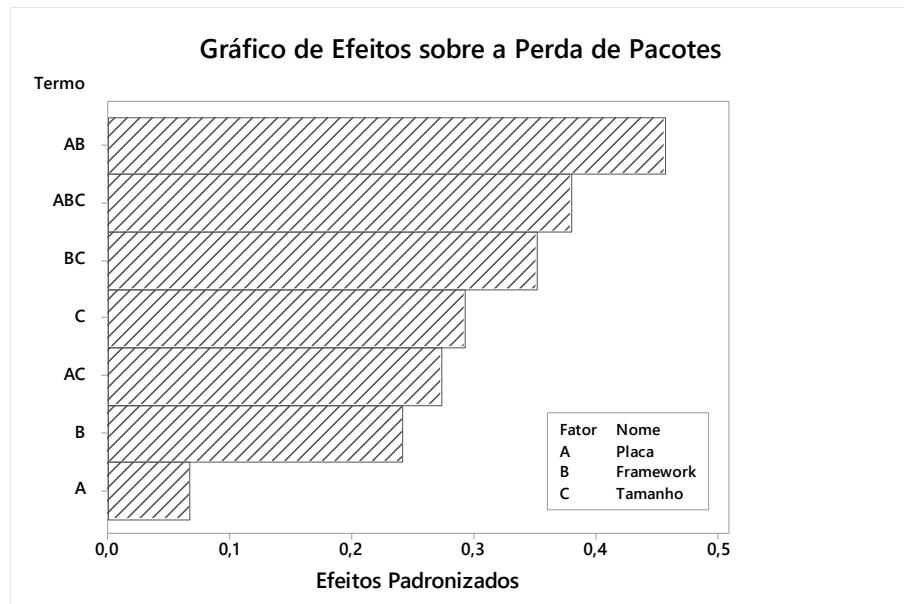
22.4.2.3 Pacotes Descartados

Nesta parte da seção é demonstrada os principais resultados obtidos da métrica Pacotes Descartados. Na métrica Pacotes Descartadas, o percentual médio de pacotes descartados foi mensurado a partir dos pacotes descartados durante o experimento. A Figura 205 apresenta o gráfico de Pareto para os fatores relacionados a variável resposta Pacotes Descartados.

Diferentemente do observado nas outras métricas, os fatores não apresentaram impactos significativos sobre a variável Pacotes Descartados. As combinações desses fatores acabaram sendo mais impactantes. O fator Placa de Rede foi o que menos significativo, indicando que esse fator não acarreta em mudanças no experimento.

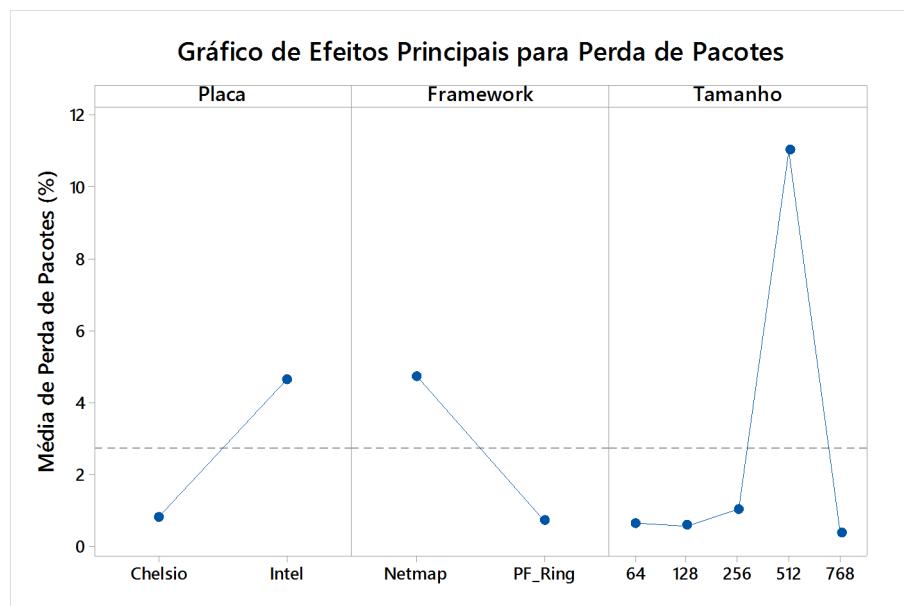
A Figura 206 mostra o gráfico de efeitos principais da métrica Pacotes Descartados. Observando o Gráfico nota-se que a variável resposta é afetada pelos níveis dos fatores: Placa de Rede e Framework. A placa de Rede da Chelsio e o framework PF_Ring apresentaram um percentual menor de pacotes descartados. Em relação ao Tamanho do Pacote os níveis não afetam de forma significativa a variável resposta, exceto o

Figura 205 – Influência dos Fatores na Acurácia (Perda de Pacote).



tamanho 512 bytes. A suspeita de que o tamanho de pacote 512 bytes tenha atingido o pico ilustrado na Figura 206 é algum conflito de configurações entre o Netmap e a Placa Intel. Esse conflito teria causado alguma interferência na alocação dos anéis de memória do Netmap, ocasionando essa perda de pacotes.

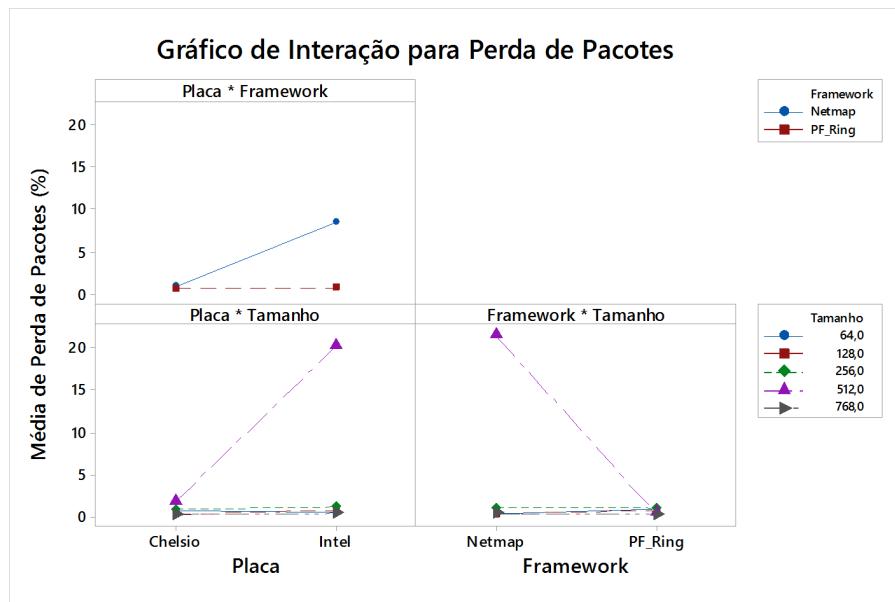
Figura 206 – Percentual Médio de cada Nível dos Fatores



O Figura 207 mostra os gráficos de interação da métrica Pacotes Descartados. Observando os gráficos, chega-se a mesma conclusão do gráfico de efeitos principais, na qual os níveis do fator (Tamanho de Pacote) não apresentam interações, logo não afetam a variável resposta. O nível 512 bytes, porém apresentou uma elevada perda de pacotes diante da combinação da placa Intel e o framework Netmap. O primeiro gráfico indica que a relação entre a Placa de Rede e o Throughput apresenta pouca dependência do framework PF_Ring já que o mesmo não apresentou interação.

A perda de pacotes foi baixa diante de todas as combinações de níveis. O uso de frameworks é vantajoso

Figura 207 – Interação dos Níveis dos Fatores



para as grandes redes na questão da baixa perda de pacotes. Como demonstrado, no gráfico de Pareto o fator Placa de Rede não apresenta dependência significativa em relação aos níveis do fator Tamanho de Pacote na métrica avaliada. Portanto a escolha de uma placa utilizando como critério a métrica Pacotes Descartados, fica a cargo dos administradores de rede. A combinação da Placa Chelsio e do Framework PF_Ring se mostrou a mais eficiente em relação ao tamanho de pacote 512 bytes.

22.4.2.4 Resumo dos Resultados

Nesta parte da seção é apresentada as principais conclusões dos resultados obtidos. Na Tabela 159 estão as combinações que obtiveram o melhor desempenho para cada métrica analisada. Na Tabela são exibidas as métricas utilizadas neste trabalho: Throughput, Uso de CPU e Pacotes Descartados. Ao lado de cada métrica há os fatores utilizados na análise de sensibilidade. A combinação trata-se de quais os níveis dos fatores foram melhores para cada métrica.

Tabela 159 – Melhores Combinações Obtidas

Métrica	Combinação Ótima		
	Placa	Framework	Tamanho de Pacote (Bytes)
Throughput	Chelsio	Netmap	64,128,256,512,768
Uso de CPU	Chelsio	PF_Ring	64,128,256
	Intel	Netmap	768
Pacotes Descartados	Chelsio	PF_Ring	64,128,256,512,768

Com o conhecimento das melhores combinações, diferentes cenários foram hipoteticamente definidos, à fim de saber qual combinação obtém melhor desempenho em um cenário específico. Os cenários são uma forma de orientar os administradores de rede e demais profissionais a escolher a melhor combinação em situações específicas de tráfego.

- Cenário 1:** Neste cénario, há uma rede que está enfrentando problemas de gargalo. A rede não consegue "absorver" o tráfego de pacotes. Para essa rede é necessário obter uma boa taxa de Throughput. Observando a Tabela 159 nota-se que a combinação Chelsio e Netmap é a melhor, inclusive para diversos tamanhos de pacotes.

- **Cenário 2:** Neste cenário, um gerente de rede deseja adquirir uma solução que exige menos recursos para tratar o tráfego de dados. Observando a Tabela 159, há duas possibilidades a serem utilizadas a fim de economizar recursos. Caso o tráfego possua muitos pacotes grandes sendo transmitidas em curto espaço de tempo, o ideal é adotar a Placa Intel e o framework Netmap. Caso o tráfego possua pacotes menores, a Placa Chelsio e o framework PF_Ring são as melhores escolhas.
- **Cenário 3:** A rede hipoteticamente está enfrentando perdas elevadas de pacotes. A perda de pacotes ocasiona degradação dos serviços de internet. Deseja-se adotar uma combinação que consiga ler os pacotes com uma perda mínima. Nesse caso a Placa Chelsio e o framework PF_Ring são boas escolhas a serem utilizadas.

22.5 Conclusão

Este trabalho desenvolveu análises de artefatos de rede (Placas de Rede e Framework de captura e processamento de pacotes). A análise visa auxiliar os administradores de redes, gerentes de infraestruturas, dentre outros profissionais da área a escolher tais artefatos. Busca-se compreender as situações que os mesmos se destacam, indicando qual combinação atende melhor os contextos específicos de tráfego, que esses profissionais precisam. Em específico as placas de redes Chelsio T520-SO-CR e Intel X520-DA2 foram submetidas a experimentos juntamente com os frameworks Netmap e PF_Ring. Foi analisado o comportamento dos mesmos em situações comuns de tráfego de redes, como tráfegos com pacotes de tamanho diferentes. A aplicabilidade do estudo em um ambiente real foi verificada através de análises de sensibilidade. Por meio dessas análises foi possível montar um tráfego que pudesse ser utilizado em um ambiente real. Com os resultados coletados e gerados, notou-se que cada métrica analisada apresentou diferentes situações de uso dos artefatos de rede. Cada situação de uso, levou em conta os impactos dos fatores nas métricas, bem como as interações desses fatores na variável resposta. Para cada variável resposta uma diferente combinação de Placa de Rede e Framework se mostrou mais viável ou não. Portanto, diferentes combinações foram indicadas, conforme os resultados, permitindo assim uma ferramenta de auxílio na tomada de decisão para os profissionais de rede.

Este trabalho teve algumas limitações, uma delas está na quantidade de placas de rede. A utilização de um número maior de Placas de Rede traria mais consistência na validação deste trabalho. No entanto, isso não foi possível devido ao elevado custo financeiro de tais aquisições. Outra limitação foi o uso das configurações padrões de cada Framework. Configurações mais avançadas como *Zero Copy* no Netmap e PF_Ring, poderiam ter sido realizadas, porém não foi possível devido a limitações de hardware das máquinas. Este trabalho poderia ter adotado o uso de gráficos de boxplot para facilitar a visualização dos dados de forma mais clara. Muitos gráficos de boxplot foram gerados devido a quantidade de combinações de níveis dos fatores, sendo optado a não inclusão desses gráficos por questões visuais.

Para trabalhos futuros, será realizada uma análise de sensibilidade com mais variáveis respostas, como uso de memória e velocidade de transmissão média. Nos experimentos seria feita a inclusão de fatores como *Threads* e diferentes serviços de rede, utilizando um mesmo protocolo. Outro ponto a ser abordado é a questão energética das placas de redes, visualizando qual a mais eficiente, levando em conta os fatores utilizados e que poderão ser inclusos. A questão energética é hoje de grande importância em estudos e trabalhos científicos. Mensurar os gastos energéticos, também é importante para empresas levando em conta a necessidade de redução de custos.

23

CAPÍTULO

SISMAC: Um Sistema para o Monitoramento de Colmeias em Tempo Real

Autores: Ricardo Bruno e Francisco Airton Silva

Resumo

A apicultura é uma atividade sustentável, de fácil manutenção e de baixo custo que movimenta um mercado bastante rentável, gerando milhares de empregos no Brasil. As abelhas são os insetos explorados nessa atividade, com importância vital para a agricultura global, fornecendo segurança alimentar. Porém as abelhas são insetos sensíveis a mudanças em seu habitat, isso pode provocar o abandono das colmeias causando perdas aos apicultores. Tentando amenizar essas perdas esse trabalho desenvolveu o protótipo de um sistema para monitoramento das colmeias de abelhas, denominado SISMAC. O sistema é composto por um aplicação web e mobile (Android), mecanismo de alerta através do mensageiro Telegram e um módulo de monitoramento instalado na colmeia. Para verificar a acurácia do sistema foi feita uma análise de sensibilidade através do método planejamento de experimentos (DoE) de os resultados proporcionaram a melhor configuração para instalação do sistema na colmeia. O sistema funcionou normalmente por 17 horas consecutivas em um ambiente real. Concluindo-se que é viável a utilização de tecnologias como o SISMAC na apicultura.

23.1 Introdução

As abelhas são os polinizadores mais importantes para as culturas que dependem da polinização animal para a produção. Juntas, essas culturas representam 35% da produção global de alimentos. Por isso, esses insetos são vitais para uma agricultura econômica e sustentável e para a segurança alimentar. Além disso, as abelhas também polinizam uma variedade de flores silvestres e, portanto, contribuem para a biodiversidade de muitos ecossistemas. Devido a esse crucial papel desempenhado pelas abelhas, as perdas severas e inexplicáveis de colônias de abelhas ao redor do mundo, recentemente relatadas, atraíram muita atenção e estimularam muitas atividades de pesquisa (ROLKE et al., 2016).

Para colocar isso em números: culturas que são independentes da polinização animal representam 65% da produção global de alimentos, deixando até 35% dependente da polinização dos animais, dos quais 90% são realizadas por abelhas do gênero *Apis Mellifera* (espécie utilizada neste estudo). Portanto, embora a

humanidade não morra se as abelhas forem extintas, elas ainda são polinizadores muito importantes em todo o mundo, pois garantem que a dieta humana não seja muito empobrecida (ROLKE et al., 2016).

As abelhas são insetos sensíveis às mudanças em seu habitat. Fatores como seca, falta de alimento, doenças, temperaturas altas ou baixas podem interferir diretamente na produção delas. As próprias abelhas realizam a regulação de temperatura dentro da colmeia, no entanto, se as temperaturas forem muito altas ou muito baixas, as abelhas deixam as colmeias em um processo conhecido como enxameação por abandono (BASTOS, 2016).

De acordo com dados da Associação Brasileira dos Exportadores de Mel (ABEMEL), o Brasil foi um grande produtor de mel, chegando a ocupar a 4^a posição no ranking mundial de exportação em 2009. Segundo (MARTINS; MAGALHÃES, 2015), a falta de chuvas regulares nos últimos anos na região Nordeste afetou fortemente a produção e reduziu em 52,1 a exportação regional, e em 26,2 a produção nacional.

Isso fez com que o país chegassem a cair sete posições no ranking até 2013, passando a ocupar a 11^a posição, ocupando atualmente a 9^a posição. Percebe-se com esses dados o quanto a produção apícola nacional é sensível a alterações climáticas. Essa estatística poderia ser amenizada com o uso de técnicas mais modernas de manejo e inovações tecnológicas no setor.

Para (MURPHY et al., 2015), as tecnologias podem ser usadas para monitorar as colmeias em termos de temperatura, umidade e outras variáveis. Por exemplo, é possível identificar problemas de colônia, evitando a enxameação das abelhas. A Internet das Coisas (*Internet of Things — IoT*) é um paradigma que interconecta a inteligência com o monitoramento sensorial aplicado recentemente na apicultura, como nos trabalhos de (MEITALOVIS et al., 2009; ESKOV; TOBOEV, 2011; SOMBRA et al., 2013; STALIDZANS; BERZONIS, 2013).

23.2 Contexto e Problema

Este trabalho tenta atenuar as perdas dos apicultores através de um sistema para monitoramento de temperatura e umidade em tempo real nas colmeias. Usando também o método de planejamento de experimentos, do inglês (*Design of Experiments - DoE*) combinando algumas configurações de sensores para realizar uma análise de sensibilidade sobre o impacto dos fatores na precisão da temperatura e umidade medidos pelos sensores na colmeia. Com isso, busca-se identificar a melhor configuração do módulo de monitoramento para esse sistema.

23.3 Objetivos

23.3.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver um sistema de monitoramento de temperatura e umidade de colmeias visando tornar o manejo das abelhas mais eficiente.

23.3.2 Objetivos Específicos

1. Implementação de um módulo baseado em sensores para monitorar a temperatura e umidade das colmeias.
2. Prover um mecanismo para alertar os apicultores quando as colmeias necessitarem de mais cuidado.
3. Criação de um sistema *Mobile* e *Web* para o apicultor acompanhar a saúde das colmeias.
4. Realizar uma análise de sensibilidade sobre o impacto dos fatores na precisão do sistema proposto.

23.4 Organização do Trabalho

O restante deste trabalho está organizado nos seguintes capítulos: o Capítulo 23.5 (Referencial Teórico) apresenta uma visão geral sobre os conteúdos abordados nesse trabalho; o Capítulo 23.11 (Trabalhos Relacionados) apresenta outros trabalhos que falam de monitoramento de colmeias; o Capítulo 23.12 (Desenvolvimento)

apresenta o sistema proposto e o funcionamento de seus módulos, bem como a análise de sensibilidade realizada sobre ele e, por fim, o Capítulo 23.16 (Conclusão) apresenta as considerações finais, a contribuição e trabalhos futuros.

23.5 Referencial Teórico

Neste capítulo será apresentado uma visão geral sobre o que é a apicultura e as abelhas *Apis Mellifera*, bem como sua importância. Abordando também a IoT, as tecnologias utilizadas no trabalho e o monitoramento de colmeias. Este capítulo ainda trata da análise de sensibilidade e sua importância em várias áreas.

23.6 Apicultura

A apicultura (técnica de criar abelhas) é um tripé da sustentabilidade: social, econômico e ambiental. Isso porque é uma atividade que gera ocupação no campo e proporciona renda para quem resolve explorá-la comercialmente sem que haja consequências para o ecossistema, tornando ela uma atividade sustentável (SANTOS; RIBEIRO, 2009). No Brasil, a apicultura forma uma cadeia produtiva composta por mais de 300 mil apicultores e uma centena de unidades de processamento de mel, que juntos empregam, temporária ou permanentemente, quase 500 mil pessoas (BACAXIXI et al., 2011).

23.6.1 Abelha *Apis Mellifera*

As abelhas do gênero *Apis Mellifera*, objeto de estudo deste trabalho, São insetos de origem africana que se adaptam facilmente a diferentes ambientes (OLIVEIRA, 2010). No Brasil, esses insetos se adaptaram muito bem às condições climáticas. As abelhas do gênero *Apis Mellifera* são as únicas disponíveis mundialmente em larga escala para a polinização, garantindo a manutenção de plantas e animais (OLIVEIRA; CUNHA, 2005).

As abelhas têm o mel como principal produto, mas também produzem cera, pólen e própolis. A cera é benéfica na produção de produtos farmacêuticos, cosméticos e lentes telescópicas; já o pólen é usado como suplemento alimentar composto de vitaminas, minerais, fibras e outros compostos. A própolis, por sua vez, é uma mistura complexa, formada por material resinoso coletado pelas abelhas dos ramos, flores, pólen e brotos, que é então misturada às secreções salivares das abelhas. A própolis é usada pelas abelhas para proteger a colônia do lado de fora, e também é explorada pelos seres humanos para a fabricação de remédios (PEREIRA; SEIXAS; NETO, 2002).

23.6.2 Termoregulação

As abelhas têm seu metabolismo e atividade influenciados pela temperatura corpórea, que por sua vez, é quase inteiramente dependente da temperatura do ambiente. Temperaturas baixas normalmente dificultam a atividade, enquanto que altas temperaturas estimulam o animal (ALMEIDA, 2008). De acordo com (KRIDI; CARVALHO; GOMES, 2014), esses insetos são excelentes termorreguladores, conseguindo manter a temperatura dentro da colmeia entre 33 e 36 graus Celsius, sendo essa faixa de temperatura ideal para que a postura da rainha, o desenvolvimento das larvas e o acúmulo de mel não sejam prejudicados. Porém, segundo (ALMEIDA, 2008), quando a temperatura atinge 41°C na colmeia, as abelhas a abandonam. No entanto, as abelhas só o fazem após longa exposição à essas altas temperaturas, comprovando que a termorregulação é uma das atividades mais importantes para a permanência das abelhas em um local.

A termorregulação também é importante para que as abelhas mantenham as propriedades essenciais do mel através de um controle rigoroso da umidade interna na colmeia que deve ficar entre 45% e 55% de umidade, fazendo com que a umidade do mel não ultrapasse 20%, influenciando no tempo de maturação, no sabor, na conservação e em sua palatabilidade (ESKOV; TOBOEV, 2011).

23.7 Monitoramento de Colmeias

O monitoramento de colmeias permite ao apicultor o acompanhamento contínuo e sistemático das variáveis ambientais dentro da colmeia, tendo como objetivo identificar e avaliar as condições dos recursos naturais em um determinado momento ou ao longo do tempo. O monitoramento de colmeias permite, ainda, compreender melhor a relação das ações do apicultor em suas colmeias (MAGALHÃES, 2012).

Sistemas de monitoramento modernos podem fornecer ao apicultor dados em tempo real e informações sobre parâmetros importantes de uma colmeia. Com base nessas informações, o apicultor pode tirar conclusões e, em caso de necessidade, tomar medidas para obter uma maximização da produção da colmeia. Tendo isso em vista, novas tecnologias podem ser facilmente aplicadas na apicultura para medir parâmetros necessários de uma colmeia (KRIDI; CARVALHO; GOMES, 2014).

23.8 Internet das Coisas

Em 2012 a União Internacional das Telecomunicações (*International Telecommunication Union – ITU*)¹ criou a recomendação ITU-T Y.2060, que apresenta uma visão geral para IoT. Tal recomendação define a IoT como uma infraestrutura global que permite serviços avançados através da interligação de coisas (físicas e virtuais), com base em informações já existentes e tecnologias de comunicação. A internet das coisas cresce em ritmo acelerado com previsões para que o número de dispositivos conectados até 2025 chegue à mais de 75 bilhões (COLUMBUS, 2016).

A IoT pode gerar um grande impacto em todas as áreas, incluindo eletrônica de consumo, saúde, e de maneira transversal, na forma como a sociedade consome informação. Com a IoT as informações podem ser acessadas a qualquer momento e de qualquer lugar através de *Internet*, através de tecnologias de rede, a IoT compõe-se de dispositivos heterogêneos conectados à rede (DIMITROV, 2016).

23.9 Análise de Sensibilidade

A análise de sensibilidade, consiste num estudo que tem por objetivo determinar o grau de variação dos resultados e dos indicadores de viabilidade de um determinado projeto face a alterações nas variáveis de entrada determinando a viabilidade do projeto. A análise de sensibilidade permite, desta forma, traçar diversos cenários no projeto e verificar até que ponto a sua viabilidade se mantém face a alterações, com diversos graus de intensidade, nas suas variáveis mais importantes (LIMA, 2011).

Aplicada inicialmente nas finanças a análise de sensibilidade se espalhou em muitas outras áreas, tais como Sociologia, Recursos Humanos, Auditoria, Contabilidade, Engenharia, Economia, Medicina e Estatística. Na análise de sensibilidade, devem ser destacadas as variações proporcionais de cada variável, ou seja, quanto uma variável tem que oscilar para provocar uma determinada alteração na variável de saída (MATTOS; VASCONCELLOS, 1989).

23.9.1 Planejamento de Experimentos

Os métodos tradicionais podem ser considerados eficientes nas análises de sensibilidades individuais, ou em situações de variações conjuntas (cenários), porém a questão da interação entre as variáveis não é relevada. Em situações reais, muito pouco se pode concluir observando variações individuais. Pode existir uma grande relação de dependência entre variáveis, o que torna necessária a busca de um método mais eficiente para sua modelagem esse método pode ser o DoE (ANDERSON; WHITCOMB, 2000).

O DoE é um método utilizado para se planejar experimentos, ou seja, para definir quais dados, em que quantidade e em que condições devem ser coletados, durante um determinado experimento, buscando basicamente satisfazer dois grandes objetivos: a maior precisão estatística possível na resposta e o menor custo. O DoE é, portanto, uma técnica de extrema importância, pois, seu emprego permite resultados mais confiáveis, economizando dinheiro e tempo. O DoE fornece as ferramentas para criação de planejamentos

¹ <https://www.itu.int/rec/T-REC-Y.2060-201206-I>

de experimentos e análise gráfica de resultados e na melhoria de processos, de maneira rápida e eficiente ([MINITAB, 2017a](#)).

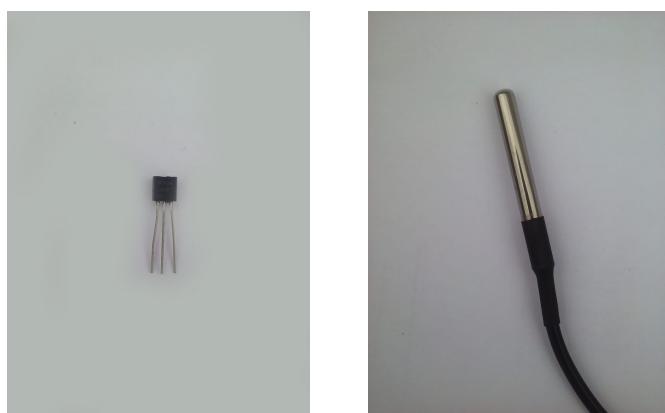
23.10 Tecnologias Utilizadas

Para realização deste trabalho foram utilizados alguns componentes eletrotônicos (sensores e módulos), sendo esses componentes essenciais na criação do protótipo do módulo implantado na colmeia e para a realização da análise de sensibilidade do SISMAC como um todo. Também foram usados plataformas de IoT essenciais para implementação do sistema juntamente com linguagens de programação.

23.10.1 Sensores de Temperatura

LM35: O sensor LM35 Figura 208 (a) é um dos sensores de temperatura usados neste trabalho, ele apresenta uma saída analógica de tensão linear proporcional à temperatura em que ele se encontra no momento, tendo em sua saída um sinal de 10mV para cada grau Celsius de temperatura, sendo um dos sensores de temperatura mais utilizados. A faixa de medição do LM35 é de 0° C a +100° C com acurácia de $\pm 0.5^\circ \text{C}$.

DS18B20: O sensor DS18B20 Figura 208 (b) é o segundo sensor de temperatura usado, é bastante preciso ($\pm 0.5^\circ \text{C}$) permitindo medições em ambientes úmidos e molhados, pois conta com a proteção de uma capsula de metal. O DS18B20 proporciona leituras de temperatura de 9 a 12 bits, onde as informações são enviadas através de uma interface, de modo que apenas um fio precisa ser conectado a um microcontrolador.



(a) LM35

(b) DS18B20

Figura 208 – Sensores de Temperatura Utilizados

23.10.2 Sensores de Umidade

DHT11: O sensor DHT11 Figura ?? (a) faz parte dos sensores de umidade usados no projeto, este sensor tem capacidade de medida na faixa de 20% a 90%, permitindo ainda a medição de temperaturas de 0 a 50 Celsius. Sua faixa de precisão para umidade é $\pm 5\%$, e temperatura é ± 2 graus.

DHT22: O sensor DHT22 Figura ?? (b) também faz parte dos sensores de umidade deste projeto, ele permite medições entre 0 a 100%, assim como o DHT11, ele também permite a medição de temperatura. É um sensor com baixo consumo de corrente, e que possui internamente um sensor de umidade capacitivo e um termistor, além de um conversor analógico/digital para comunicação com o microcontrolador.

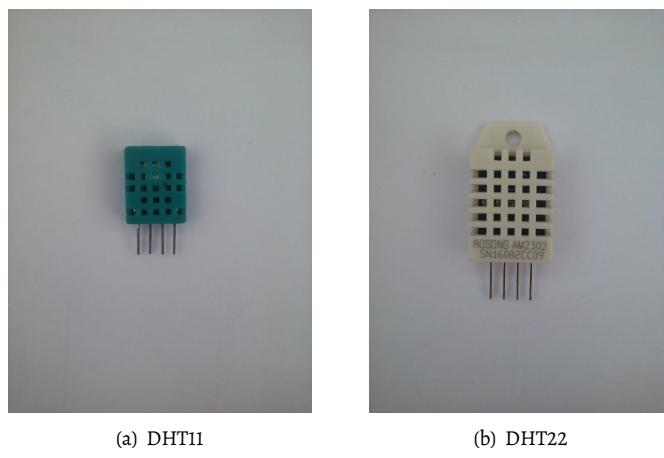


Figura 209 – Sensores de Temperatura Utilizados

23.10.3 Arduino

O *Arduino*² é uma plataforma de prototipagem eletrônica de código aberto, formado por um *hardware* (placa Arduino) como pode ser visto na Figura 210, e um ambiente de programação integrado (*Integrated programming environment - IDE*). De acordo com o site oficial O *Arduino* ficou bem popular para o desenvolvimento de produtos IoT, onde vários engenheiros, estudantes, desenvolvedores e fabricantes de todo o mundo usam o *Arduino* para inovar em casas inteligentes, agricultura, veículos autônomos e entre outros.



Figura 210 – Arduino Mega

23.10.4 Arduino Ethernet Shield w5100

O *Arduino Ethernet Shield w5100*³ Figura 211 permite a conexão de uma placa Arduino com a internet. Suportando até quatro conexões de soquete simultâneas. Para utilização dessa placa é necessária a biblioteca *Ethernet*. A *Ethernet Shield* se conecta a uma placa *Arduino* usando pinos e soquetes. O *Arduino Ethernet Shield* tem como principal conexão o RJ-45, que usa um cabo de rede, para ser ligado ao um computador ou ao um roteador. A placa possui ainda um *slot* para um cartão de memória, que pode ser usado para armazenar arquivos para usar através da rede.

23.10.5 Linguagens de Programação

HTML O HTML⁴ (*HyperText Markup Language*) é a base da estrutura de toda página web, é uma linguagem de marcação utilizada para desenvolvimento de projetos web. O HTML é formado por tags que são utilizadas para informar ao navegador como deve ser apresentado o website.

² <https://www.arduino.cc/>

³ <https://bit.ly/2Hbgn80>

⁴ <https://www.w3schools.com/html/default.asp>



Figura 211 – Arduino Ethernet Shield w5100

CSS O CSS⁵ (*Cascades Style Sheets*) descreve o estilo de um documento HTML, ou seja, define como os elementos serão exibidos na tela. O principal benefício do CSS é a separação entre o formato e o conteúdo de um documento HTML.

JavaScript O JavaScript é uma linguagem de *script* orientada a objetos, multiplataforma, pequena e leve. Dentro de um ambiente de *host* o JavaScript pode ser ligado aos objetos deste ambiente para prover um controle programático sobre eles. O núcleo do JavaScript pode ser estendido para uma variedade de propósitos como cliente ou servidor complementando assim a linguagem (FLANAGAN, 2006).

23.10.6 Plataformas

Thinger IO Thinger IO⁶ é a plataforma de código aberto para a IoT, essa plataforma coleta dados de aplicativos, sensores e redes sociais. Através da IoT a Thinger IO, pode oferecer acesso à uma ampla gama de plataformas de prototipagem como Arduino, Raspberry e Sigfox. A Thinger IO tem como principal vantagem a coleta primária e de dados, dos sensores conectados diretamente a ela. A Thinger IO pode utilizar tanto o HTTP (*Hypertext Transfer Protocol*) como o HTTPS (*Hyper Text Transfer Protocol Secure*) para transferência dos valores coletados.

Keen IO O Keen IO⁷ é uma plataforma de armazenamento e análise que permite aos desenvolvedores realizarem análises enviadas a partir de diversas fontes (sites, aplicativos, plataformas). Um das principais vantagens do Keen IO são as suas *Application Programming Interfaces* (APIs) que permitem ao desenvolvedor transmitir, analisar, exibir e proteger dados. Dentre as linguagens que podem ser utilizadas no Keen IO estão JavaScript, Ruby, Python, PHP, Java e Objective-C.

Telegram O Telegram⁸ é um aplicativo de mensagens para *smartphones* que tem como objetivo enviar e receber mensagens de texto e imídia de e para usuários individuais. Existem muitos aplicativos de mensageiros, mas o que o diferencia é a possibilidade de criação de *bots*. Onde apartir destes *bots* pode-se enviar notificações para um usuário ou um grupo de usuários de forma totalmente gratuita. Os *bots* são criados através de uma API fornecida pelo próprio Telegram. Essa API suporta as linguagens de programação Phyton, Java, C, C++ e Lua.

23.11 Trabalhos Relacionados

Este capítulo apresenta alguns trabalhos produzidos acerca do monitoramento de colmeias de abelhas. Na apresentação dos trabalhos foi destacado as correlações com o trabalho proposto bem como as diferenças encontradas.

(MEITALOVS et al., 2009) desenvolveram um sistema de monitoramento de colmeias para reconhecer condições de pré-aquecimento de uma colônia de abelhas. Para isso, os autores propuseram a colocação de alguns sensores em colmeias permitindo a descoberta de razões particulares na mudança da colmeia. No sistema, os sensores enviam os dados coletados para um computador, que os armazenava no Microsoft

⁵ <https://www.w3schools.com/css/default.asp>

⁶ <https://thinger.io/>

⁷ <https://keen.io/>

⁸ <https://telegram.org/faq>

Access de onde o usuário poderia visualizá-los. Os autores concluirão que o sistema era capaz de detectar o super aquecimento de uma colmeia dado a velocidade de mudanças de temperatura relativamente baixa dentro da mesma. Destacando ainda que o sistema poderia ser facilmente usado em escala pelos apicultores.

O trabalho de (MEITALOVS et al., 2009) se relaciona com o trabalho proposto pelo fato de ambos proporem uma forma de monitoramento de colmeias. Porém, no nosso, trabalho todo o processo de coleta, envio, armazenamento e análise dos dados é feito de forma totalmente autônoma, sem a necessidade de intervenção por parte do apicultor.

(KVIESIS et al., 2015) propuseram um sistema de rede de sensores de umidade e temperatura para monitoramento *online* de colônias de abelhas, a fim de facilitar a vida dos apicultores. As propriedades monitoradas neste trabalho foram temperatura e umidade, porém os autores não deram uma explicação sobre o porquê dessas propriedades serem escolhidas. O sistema consistia em sensores enviando dados para um banco de dados que poderia ser consultado por uma aplicação *Web* para exibição dos dados ao usuário. Os autores concluirão que sistemas desse tipo pode minimizar a intervenção do apicultor nas colônias, maximizando a saúde delas.

Assim como trabalho proposto (KVIESIS et al., 2015) desenvolveu uma aplicação para a visualização dos dados coletados, entretanto, a aplicação desenvolvida é apenas *Web*, enquanto que a nosso conta também com uma aplicação *mobile*. Outra diferença é a análise de sensibilidade realizado no sistema desenvolvido.

Já (KRIDI; CARVALHO; GOMES, 2016) propuseram em seu trabalho criar um sistema proativo capaz de detectar, via rede de sensores sem fio, o aquecimento interno em colmeias de abelhas. Para garantir temperaturas adequadas ele utilizou técnicas de *clustering* por similaridade para descobrir os padrões térmicos. O experimento foi realizado em um apiário com duas colmeias diferentes contendo o mesmo número de quadros, crias e população. No decorrer do experimento, foi possível perceber os problemas que se enfrenta no monitoramento em locais inóspitos. Tendo que mudar algumas vezes a estratégia de instalação dos sensores na colmeia. Ao final, o autor verificou que o monitoramento das colmeias é capaz de detectar informações específicas sobre o comportamento das abelhas em condições de temperatura adversas.

Assim como o nosso, (KRIDI; CARVALHO; GOMES, 2016) tem o objetivo de monitorar variáveis em uma colmeia, no entanto, apenas a temperatura é monitorada. No trabalho proposto aqui temperatura e a umidade também são monitoradas, além disso, nosso trabalho tem aplicações para visualização dessas variáveis em tempo real.

(MURPHY et al., 2016) desenvolveram uma rede de sensores heterogêneos para monitorar uma variedade de parâmetros, a fim de descrever as condições internas das colônias. Para viabilizar os testes, os autores criaram um protótipo do sistema que foi instalado em uma colmeia para realização dos experimentos. O sistema fazia seis leituras diárias de cada sensor, armazenava os dados da temperatura em um cartão de memória que era recolhido a cada 24 horas para análise dos dados. Ao final, os pesquisadores concluíram que as mudanças climáticas têm forte influência sobre a temperatura e umidade no interior da colmeia.

Assim como em nosso trabalho, (MURPHY et al., 2016) criou um sistema para monitorar parâmetros de uma colmeia, todavia, diferentemente do nosso, a coleta dos dados resultantes desse monitoramento era realiza-dá de forma manual. Sem nenhuma conexão com a internet ou com um sistema remoto que possibilitasse uma melhor apresentação das condições dentro da colmeia.

(DUTRA, 2016) desenvolveu um sistema computacional para a apicultura com o objetivo geral de monitorar as variáveis de uma colmeia de abelhas, possibilitando o registro histórico e a visualização dos dados coletados. sistema coleta os dados, armazena, processa e envia-os a um aplicativo *Web*, que tem a função de exibi-los, esse envio é feito por *Ethernet* ou através de um cartão de memória.

O sistema de (DUTRA, 2016) é bem semelhante ao proposto neste trabalho com as variáveis monitoradas (temperatura e umidade), sendo as mesmas. As diferenças entre os trabalhos é a análise de sensibilidade e notificação de problemas na colmeia realizada apenas neste trabalho.

(SILVA, 2017b) Oferece em seu trabalho uma solução para a perda de abelhas devido à vários fatores. Sendo essa solução um sistema de monitoramento que utiliza sensores, o sistema conta com um aplicativo *mobile* para visualização dos dados pelo apicultor. O autor conclui que um sistema desse tipo pode auxiliar o apicultor no cuidado do apiário e nas tomadas de decisões referente as suas colmeias.

Semelhante ao anterior o trabalho de (SILVA, 2017b) tem analogia com o trabalho proposto pela construção de um sistema de monitoramento com conexão a internet a aplicativo para visualização. Todavia, o sistema proposto nesse trabalho conta com mecanismo de notificação, duas plataformas para apresentação dos dados e a análise de sensibilidade realizada apenas no trabalho proposto aqui.

Todos os trabalhos citados se relacionam com o trabalho proposto pela ênfase no monitoramento de colmeias de abelhas, porém com o intuito de se saber quais das variáveis mais afetavam a acurácia de um sistema de monitoramento de colmeias, foi feita uma análise de sensibilidade utilizando DoE. O trabalho proposto se diferencia também dos demais por contar com uma plataforma *Web* e *mobile* para apresentação dos dados coletados, além também de utilizar uma forma totalmente gratuita de notificação sendo esse forma os *bots* do mensageiro Telegram. A Tabela ?? resume a comparação deste trabalho com os relacionados, usando os seguintes critérios: Análise de sensibilidade, Propriedades monitoradas, Exibição e Notificação.

23.12 Desenvolvimento

Neste capítulo será apresentado o sistema proposto, incluindo sua arquitetura, a construção e o funcionamento de seus módulos(aplicativos *web* e *mobile*, mecanismo de alerta e o protótipo de monitoramento). Será apresentado também a análise de sensibilidade feita no sistema, bem como sua validação e resultados da mesma.

23.13 Sistema de Monitoramento Automático de Colmeias

O SISMAC é um sistema para monitorar a temperatura e umidade interna das colmeias em tempo real utilizando componentes de baixo custo. O Funcionamento do SISMAC e a interação dos seus módulos pode ser observado na Figura 212. Tudo começa dentro da colmeia (1) quando os sensores de temperatura e umidade (2) coletam os dados da colmeia e os envia para o Arduino com uma *shield ethernet w5100* acoplada. Nesse momento, será feito a verificação da temperatura obtida com o objetivo de se identificar valores abaixo ou acima do normal e então disparar uma mensagem de alerta pelo Telegram (3). Em seguida, todos os dados (temperatura e umidade) são enviados pela *internet* através da plataforma Thinger IO, por meio de uma conexão entre o Arduino e a plataforma. A partir daí, os dados são armazenados e analisados pela plataforma Keen IO (4). Quando, por último, as informações sobre a colmeia são disponibilizadas ao apicultor por uma aplicação *web* e ou *mobile*, juntamente com possíveis alertas através do mensageiro Telegram (5).

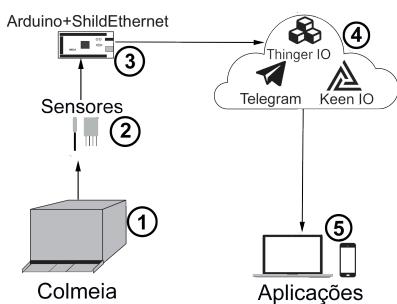


Figura 212 – Diagrama de Interação dos Módulos do SISMAC

23.13.1 Aplicação Web

Seguindo o objetivo primordial deste trabalho que é desenvolver um sistema para monitoramento de colmeias em tempo real. Foi desenvolvido uma aplicação *web* que tem como objetivo aumentar a comodidade no monitoramento. A aplicação foi desenvolvida em HTML e JavaScript apartir das APIs de desenvolvimento da plataforma Keen IO abordada na Subseção 23.10.6 deste trabalho. A Figura 213 mostra a aplicação *web*, a aplicação permite que usuário monitore a saúde da colmeia no âmbito da temperatura e umidade.

A aplicação tem uma interface bem simples, com os níveis de temperatura e umidade sendo informados logo na parte superior da tela sendo fornecido também as temperaturas mínima e máxima das últimas 24 horas. Logo abaixo são fornecidos gráficos que dão uma melhor visão dos dados colhidos durante o dia.

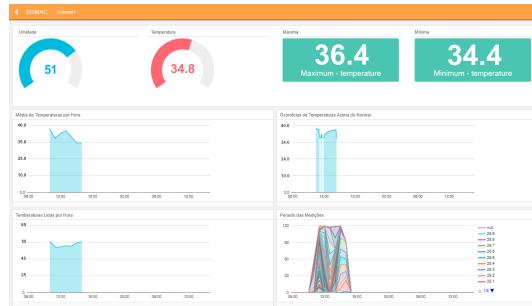


Figura 213 – Aplicação Web

23.13.2 Aplicação Mobile

Visando ainda mais a comodidade dos usuários, foi desenvolvido um aplicativo *mobile* com base na aplicação *web*. O aplicativo foi desenvolvido para a plataforma Android. Assim como a aplicação *web* o aplicativo *mobile* apresenta os dados referentes a temperatura e umidade da colmeia além dos gráficos, também presentes no aplicativo. A Figura 214 mostra a tela de exibição dos dados, como a tela de telefones celulares é relativamente pequena, é necessário a rolagem da tela para que se possa visualizar todos os dados.



Figura 214 – Aplicação Mobile

23.13.3 Notificação

Para que o apicultor não precise acessar constantemente um dos aplicativos, a fim de verificar a temperatura da colmeia, o SISMAC também possui uma funcionalidade para notificar, por meio do mensageiro instantâneo Telegram, quando houver picos de temperatura fora do normal. Cada notificação é disparada no intervalo de 10 minutos, caso existam pelo menos 5 picos simultâneos, contendo a seguinte mensagem: "Tem Algo Errado na Colmeia, Média de Temperatura: Média".

O principal diferencial dessa função é a sua total gratuidade, possível graças aos *bots* do Telegram abordado na Seção 23.10.6 deste trabalho. Para ter essa função em mãos o apicultor teria apenas que instalar o aplicativo do mensageiro Telegram em seu celular Android sem custo algum.

23.14 Análise de Sensibilidade no SISMAC

Esta seção descreve os experimentos realizados para analisar os fatores que tiveram mais influência na acurácia do sistema. Através da realização de uma análise de sensibilidade.

23.14.1 Planejamento

O SISMAC tem a capacidade de medir a temperatura e a umidade, desta forma tornou-se necessária a realização de dois DoE's visto que os sensores utilizados para medir a temperatura são diferentes dos utilizados para medir a umidade e com algumas variáveis independentes diferentes. Na execução de um DoE, o primeiro passo é definir os fatores e níveis do experimento.

A Tabela 160 apresenta os fatores e níveis escolhidos com o objetivo de identificar o fator com maior impacto na precisão do SISMAC no que diz respeito a temperatura. Da mesma forma, a Tabela 161 apresenta os fatores e níveis escolhidos para identificar o fator que mais impacta na precisão do SISMAC no que diz respeito a umidade.

Tabela 160 – Fatores e Níveis Temperatura

Fator	Nível I	Nível II
Período do Dia	Manhã	Tarde
Posição do Sensor	Centro	Lateral
Presença Própolis	Total	Parcial
Tipo de Sensor	LM35	DS18B20

Tabela 161 – Fatores e Níveis Umidade

Fator	Nível I	Nível II
Período do Dia	Manhã	Tarde
Posição do Sensor	Centro	Lateral
Proteção	Com	Sem
Tipo de Sensor	DHT11	DHT22

O próximo passo é combinar os fatores e níveis para definir as especificações do experimento. A Tabela 162 apresenta as possíveis combinações dos fatores e níveis da temperatura. A Tabela 163 apresenta as possíveis combinações dos fatores e níveis da umidade. O desenho do experimento resultou em 16 combinações para cada DoE.

Tabela 162 – Combinações Temperaturas

Nº	Combinações
1	Manhã - Centro - Parcial - DS18B20
2	Manhã - Centro - Total - DS18B20
3	Manhã - Lateral - Parcial - DS18B20
4	Manhã - Lateral - Total - DS18B20
5	Manhã - Lateral - Parcial - LM35
6	Manhã - Lateral - Total - LM35
7	Manhã - Centro - Parcial - LM35
8	Manhã - Centro - Total - LM35
9	Tarde - Lateral - Parcial - DS18B20
10	Tarde - Lateral - Total - DS18B20
11	Tarde - Centro - Parcial - DS18B20
12	Tarde - Centro - Total - DS18B20
13	Tarde - Centro - Parcial - LM35
14	Tarde - Centro - Total - LM35
15	Tarde - Lateral - Parcial - LM35
16	Tarde - Lateral - Total - LM35

Tabela 163 – Combinações para Umidade

Nº	Combinações
1	Manhã - Centro - Com - DHT22
2	Manhã - Centro - Sem - DHT22
3	Manhã - Lateral - Com - DHT22
4	Manhã - Lateral - Sem - DHT22
5	Manhã - Lateral - Com - DHT11
6	Manhã - Lateral - Sem - DHT11
7	Manhã - Centro - Com - DHT11
8	Manhã - Centro - Sem - DHT11
9	Tarde - Lateral - Com - DHT22
10	Tarde - Lateral - Sem - DHT22
11	Tarde - Centro - Com - DHT22
12	Tarde - Centro - Sem - DHT22
13	Tarde - Centro - Com - DHT11
14	Tarde - Centro - Sem - DHT11
15	Tarde - Lateral - Com - DHT11
16	Tarde - Lateral - Sem - DHT11

23.14.2 Coleta dos Dados

Os valores de temperatura e umidade foram coletados em uma colmeia padrão Langstroth pertencente a Associação de Apicultores da Comunidade Várzea, situada na cidade de Isaías Coelho-PI. A Figura 215 apresenta a colmeia usada para realização dos experimentos. Para cada combinação (descritas nas Tabelas 162 e 163), foram coletadas 30 medições de temperatura, o que resultou em 480 observações para cada DoE.

As coletas foram realizadas em quatro dias. O período de coletas da parte da manhã teve início às 7:30 e término às 10:30, e com início às 13:00 e término às 16:00 horas pela tarde. Foi usado um termômetro digital



Figura 215 – Colmeia Langstroth Utilizada

HT3320⁹ (com acurácia de $\pm 0,5^{\circ}\text{C}$ para temperatura e $\pm 1\%$ para umidade) para comparação com sensores. O cálculo da acurácia baseou-se na diferença do valor da medição do termômetro HT3320 com o sensor. Por exemplo, supondo que o termômetro digital obteve 35°C , e o sensor $34,5^{\circ}\text{C}$ a acurácia do sensor em relação ao termômetro seria de cerca de 98,5%.

23.14.3 Análise de Dados para Temperatura

A Figura 216 apresenta um Gráfico de Pareto para os fatores relacionados a temperatura. O Gráfico de Pareto descreve a importância de um efeito por seu valor absoluto, dispondo uma linha vermelha vertical. Cada barra significa o nível de impacto do fator (ou combinação de fatores) sobre a variável dependente. A linha vermelha tracejada representa a magnitude mínima de efeitos estatisticamente significativos (quantificada como 1.97), considerando-se o critério de significância estatística $\alpha = 0.05$. Ao passo que a barra do fator transpassa esta linha, sua influência será maior sobre a acurácia da variável dependente.

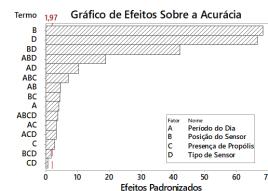


Figura 216 – Influências dos Fatores na Acurácia (Temperatura).

Quando o impacto de um fator é dito alto, quer dizer que alterando o nível daquele fator são obtidos valores bem distintos. Neste estudo foi observado que os fatores com impactos mais elevados sobre a acurácia no que diz respeito a medição de temperatura foram: Posição do Sensor e Tipo de Sensor. A Posição do Sensor possui maior impacto dentre todos os fatores. Dessa forma, a localização do sensor é determinante na confiabilidade da medição. Dado que se sabe os fatores com maiores impactos, o próximo passo é saber quais níveis tiveram melhores resultados.

Para saber quais níveis se destacaram, foi realizado um teste T para 2 amostras nos fatores que mais influenciaram na acurácia do SISMAC para a temperatura, o teste T para 2 amostras se aplica a planos amostrais onde se deseja comparar dois grupos independentes. As Figuras 217, 218, 219 e 220 apresentam os resultados do teste T, usando o gráfico de boxplot. O boxplot é um gráfico utilizado para avaliar a distribuição empírica dos dados, o boxplot fornece informação sobre localização, dispersão, assimetria, comprimento da cauda dos dados analisados.

Observando o gráfico da Figura 217 constatou-se que o sensor situado na posição central da colmeia teve uma acurácia média de 95,43% (melhor acurácia), sendo assim o melhor local para o posicionamento do sensor de temperatura seria o centro, uma vez que o fator posição é o que mais interfere na acurácia. Isso ocorre porque possivelmente nas bordas da colmeia, a temperatura sofre mais influência do ambiente externo. Com a acurácia do SISMAC sendo afetada significativamente, na mensuração da temperatura o que consequentemente pode apresentar um quadro que não condiz com a realidade da colmeia.

⁹ <https://bit.ly/2K1frEg>

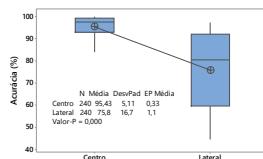


Figura 217 – Boxplot Posição do Sensor (Temperatura)

Em termos do Tipo de Sensor a Figura 218, contatou-se que o sensor DS18B20 teve melhor resultado com média de acurácia de 95,19%. O sensor DS18B20 tem o circuito protegido por uma cápsula de metal, enquanto que o sensor LM35 não. Acreditamos que a não proteção do sensor LM35 faz com que ele fique sujeito a interferência externas.

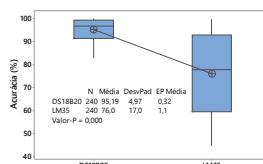


Figura 218 – Boxplot Tipo de Sensor (Temperatura)

Os fatores Período do Dia e Presença de Própolis não tiveram tanto impacto sobre a acurácia do sistema. Isso pode ser observado nas Figuras 219 e 220, que mostram claramente os níveis de ambos os fatores com média de acurácia tendo valores bem próximos uns dos outros. Com esses resultados concluiu-se que o sistema funcionaria bem em qualquer horário do dia (manhã ou tarde). Assim como, também, a presença ou não de própolis não impactou na acurácia do sistema.

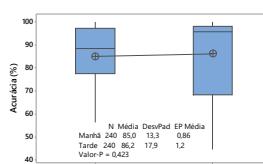


Figura 219 – Boxplot Período do Dia (Temperatura)

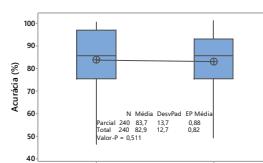


Figura 220 – Boxplot Presença de Própolis (Temperatura)

No gráfico de barras que pode ser visto na Figura 221 estão dispostas as médias de acurácia de cada combinação, sendo a melhor combinação a 11 (Tarde, Centro, Parcial, DS18B20) Tabela 162. Seguida da combinação 2 (Manhã, Centro, Parcial, DS18B20) Tabela 162. Com esses resultados foi possível montar o módulo de monitoramento da melhor forma possível.

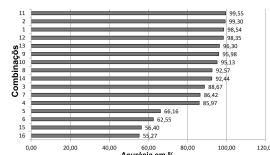


Figura 221 – Acurácia Média (Temperatura)

23.14.4 Análise de Dados para Umidade

A Figura 222 mostra o gráfico de Pareto desta vez, para os fatores relacionados a umidade. A magnitude mínima de efeitos estatisticamente significativos teve a mesma quantificação do gráfico de Pareto para os fatores relacionados a temperatura (1.97). Neste caso observou-se que os fatores com impacto mais elevado sobre a acurácia no que diz respeito a medição de umidade foram: Posição do Sensor e Proteção. A Posição do Sensor continuou a possuir maior impacto dentre todos os fatores.

Nessa análise percebeu-se ainda que o tipo de sensor (DHT11 ou DHT22) não tiveram muito impacto sobre a acurácia, mas sim a proteção que eles receberam. Da mesma forma que na análise anterior o próximo passo é saber quais níveis tiveram melhores resultados, nessa análise.

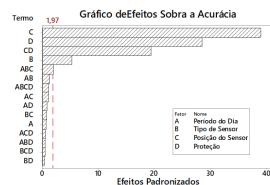


Figura 222 – Influências dos Fatores na Acurácia (Umidade).

As Figuras 223, 224, 225 e 226 apresentam gráficos boxplot utilizados para medir os efeitos de cada fator na acurácia no que diz respeito a umidade. Observando a Figura 223, novamente a posição central da colmeia apresentou uma melhor acurácia com média de 95,31%, com isso constatou-se que o melhor local para o posicionamento do sensor de umidade também seria no centro.

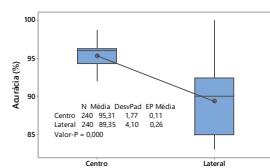


Figura 223 – Boxplot Posição do Sensor (Umidade)

O segundo fator que mais afetou a acurácia na medição de umidade foi a presença ou não de uma proteção sobre o sensor, a proteção sobre o sensor se trata de uma rede de *nylon* utilizada para recobrir os sensores de umidade utilizados na análise (DHT11 e DHT22). Com a média de acurácia ficando em 94,51% nos casos em que os sensores tinham a proteção. A proteção aumentou significativamente a acurácia do SISMAC, acreditamos que isso ocorreu por que a rede evitou o contato direto do sensor com as abelhas com o mesmo, tendo contato apenas com ar da colmeia.

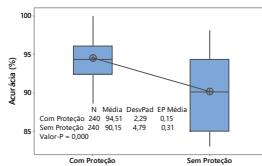


Figura 224 – Boxplot Proteção (Umidade)

Nessa análise os fatores tipo de sensor e período do dia tiveram pouca ou quase nem uma influência sobre a acurácia do sistema, esses resultados podem ser observados nas Figuras 225 e 226 onde a diferença das médias de acurácia oscila entre valores inferiores a 1% Mais uma vez o período do dia não interferiu na acurácia do sistema. Com o tipo de sensor também não interferindo na acurácia, contrariando as expectativas, já que o sensor DHT22 é considerado melhor que o sensor DHT11 (de acordo com o fabricante), nesse senário a diferença entre os sensores não interferiram com expressão nos resultados da análise.

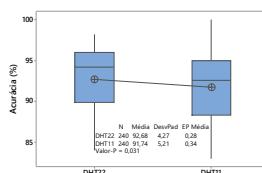


Figura 225 – Boxplot Tipo de Sensor (Umidade)

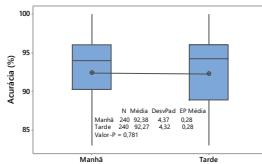


Figura 226 – Boxplot Período do Dia (Umidade)

Assim como na análise anterior as melhores combinações puderam ser descobertas como mostra o gráfico de barras na Figura 227, nesse caso a melhor combinação obtida foi a 1 (Manhã, Centro, Com Proteção, DHT22) Tabela 163. Seguida da combinação 11 (Tarde, Centro, Com Proteção, DHT22) Tabela 163. Com esses resultados o módulo para monitoramento da umidade pode ser montada da maneira mais viável.

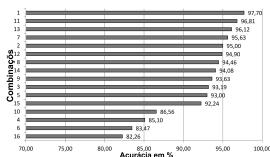


Figura 227 – Acurácia Média (Umidade)

23.15 Validação do Sistema

Esta seção descreve como foi feito o planejamento e realização da validação do SISMAC em um ambiente real. Bem como os resultados obtidos com essa validação.

23.15.1 Módulo de Monitoramento

Com base nos resultados da análise de sensibilidade, foi construído um módulo de monitoramento para realização da validação do SISMAC. O módulo presente na Figura 228 é composto por uma placa Arduino Mega, uma placa *ethernet* w5100, um sensor DS18B20, um sensor DHT22 e um cabo de rede RJ-45 de cerca de 35 metros para conexão com o roteador de *Internet*.

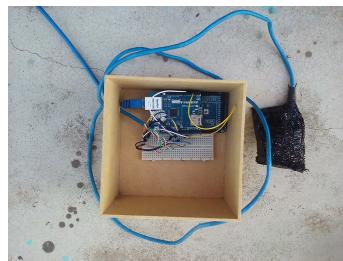


Figura 228 – Módulo de Monitoramento

23.15.2 Monitoramento

A validação do SISMAC foi realizada em uma colmeia Langstroth, povoada com abelhas do gênero *Apis Mellifera*. A colmeia utilizada novamente cedida pela Associação de Apicultores da Comunidade Várzea. Para que a realização dos testes de validação fossem viáveis, a colmeia utilizada teve que ser transportada para as proximidades da sede da associação dos apicultores pelo fato de esse ser o único local com conexão com *Internet* em que poderia-se colocar uma colmeia povoada.

A Figura 229 mostra a colmeia com o módulo de monitoramento já instalado. O módulo foi instalado na parte superior da colmeia com os sensores ficando posicionados no centro das colmeias sendo esse o melhor local para ambos os sensores de acordo com a análise de sensibilidade realizada nesse trabalho. O monitoramento no ambiente real teve duração de 17 horas com o início oficial do monitoramento as 10:00 e término as 03:05 com um total de 206 para cada grandeza monitorada (temperatura e umidade).



Figura 229 – Colmeia Utilizada na Validação

A Figura 230 mostra um gráfico de linhas com o histórico de temperatura da colmeia, durante todo o tempo de validação do sistema (17 horas), o gráfico mostra que a temperatura interna da colmeia se manteve praticamente por todo o tempo em níveis normais (entre 33 e 36 °C), ao mesmo tempo, que a temperatura externa oscilava de acordo com horário do dia.

O gráfico mostrou a independência da temperatura interna de colmeia em relação ao seu exterior, sendo essa situação a normal de uma colmeia saudável. Durante o tempo de validação a temperatura máxima atingiu 36,4 °C e temperatura mínima atingiu 34,4 °C. Isso mostrou que o SISMAC conseguiu através do sensor utilizado, monitorar a temperatura de uma colmeia por um período de tempo longo.

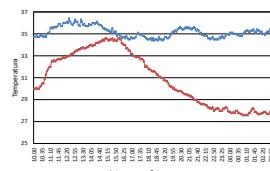


Figura 230 – Histórico de Temperatura

Para o monitoramento da umidade da colmeia o SISMAC também mostrou êxito, como pode ser observado na Figura 231 que mostra através de um gráfico de linhas o histórico de umidade dentro da colmeia no decorrer da 17 horas de monitoramento. Com a umidade também se mantendo em níveis normais com o nível mais alto de umidade monitorado sendo de 56% e o mais baixo foi 45%. Isso ocorria ao mesmo tempo, que umidade externa oscilava de acordo com o horário ultrapassando os 75% de umidade na parte da noite.

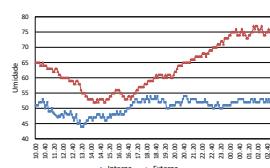


Figura 231 – Histórico de Umidade

23.16 Conclusão

Este trabalho desenvolveu o protótipo de um sistema para o monitoramento automático de colmeias utilizando IoT, denominado SISMAC, tendo como intuito o auxílio no manejo de abelhas mais eficiente. A aplicabilidade do sistema em um ambiente real foi verificada através de análises de sensibilidade. Por meio dessas análises foi possível conceber um sistema que pudesse ser utilizado em um ambiente real. Possibilitando sua validação sem que houvesse dúvidas quanto a sua capacidade de monitoramento. Nos testes de validação, o sistema mostrou boa eficácia atingindo as expectativas de funcionamento por um período de tempo longo com os sensores funcionando normalmente no decorrer das 17 horas de monitoramento. Outra contribuição gerada por este trabalho, foi a obtenção de resultados positivos quanto à aplicabilidade de tecnologias como a utilizada; em colmeias. Resultados possíveis graças a análises estatísticas realizadas no protótipo construído.

Este trabalho teve algumas limitações, uma delas está na validação que foi realizada com apenas uma colmeia, uma vez que a utilização de um número maior delas traria mais consistência na validação deste trabalho. No entanto, isso não foi possível devido ao número limitado de componentes eletrônicos disponíveis (apenas um componente de cada). Outra limitação é a infraestrutura necessária para realização de testes em uma escala maior. Uma vez que o apiário mais próximo ficava cerca de 1 km de distância de um ponto de energia.

Para trabalhos futuros, pretende-se monitorar, por meio de novos sensores, outras variáveis presentes em uma colmeia, como peso, ruído e qualidade do ar. Verificando ainda, se tais sensores funcionam corretamente em uma colmeia. O SISMAC na sua constituição atual não tem capacidade de intervenção em uma colmeia. Por tanto esse será outro ponto abordado em trabalhos futuros, o estudo de formas não invasivas de um sistema intervir automaticamente em uma colmeia caso seja necessário. A autonomia energética também será outro ponto abordado em trabalhos futuros, visto que locais para criação de abelha geralmente são afastadas de redes energéticas, o que dificulta bastante a aplicação e a instalação de módulos de monitoramento.

Avaliação de Desempenho de Contêineres sob Dois Aspectos: Planejamento de Arquiteturas e Armazenamento de Dados

Autores: Breno Santos e Francisco Airton Silva

Resumo

O conceito de virtualização computacional, apesar de antigo, se tornou lugar-comum na viabilização de aplicações que compartilham recursos computacionais na infraestrutura da nuvem. As máquinas virtuais (do inglês *virtual machines*, VMs) trouxeram benefícios como a possibilidade de realizar *live migration* para prover tolerância a falha e balanceamento de carga. Porém, apesar das vantagens, as VMs adicionam camadas extras de abstração, resultando em perda de eficiência. Como alternativa as VMs, atualmente os contêineres se apresentam como uma solução mais leve e com bom desempenho. O Docker é uma plataforma para contêineres das mais populares, que fornece uma maneira automatizada de hospedar aplicações. Este trabalho realiza duas avaliações de desempenho baseadas em análise numérica e de sensibilidade com o objetivo de analisar a utilização de recursos durante o planejamento de uma arquitetura, e identificar os fatores que mais impactam na eficiência de um Sistema de Gerenciamento de Banco de Dados (SGBD), utilizando Docker como contêiner.

Palavras-chaves: contêiner, docker, desempenho, SPN, DoE, SGBD.

24.1 Introdução

A virtualização é uma tecnologia estabelecida e em expansão. Muitos aplicativos, sistemas de gerenciamento, modelos e simuladores foram criados para promover e melhorar seu desempenho. Algumas das principais vantagens da virtualização incluem flexibilidade, capacidade, poder de processamento, crescimento da demanda e eficiência energética. No entanto, recentemente esse paradigma vem mudando. Um novo modelo de virtualização a nível do sistema operacional que permite instâncias com vários espaços de usuário adquiriu apoio de grandes empresas. Esse modelo de virtualização é chamado de contêiner. Esta

tecnologia é semelhante a Máquinas Virtuais (VMs), porém os contêineres não exigem uma camada de emulação para serem executados, tornando seu funcionamento computacionalmente mais leve que as VMs (DUA; RAJA; KAKADIA, 2014). Os contêineres são criados com os recursos e processos do Sistema Operacional anfitrião e tem a capacidade de criar um ambiente de execução que isola as aplicações de forma que elas entendam estar funcionando em Sistemas Operacionais(SO) independentes, quando na verdade estão compartilhando o núcleo do SO anfitrião com outros contêineres(SILVA, 2017a). O Docker é uma plataforma para contêineres das mais populares, que fornece uma maneira automatizada de hospedar aplicações.

O Docker permite a criação de contêineres em menos de um segundo, eliminando a sobrecarga bem conhecida da inicialização de *hypervisors* (FINK, 2014). Esta plataforma vem sendo bastante utilizada para atender um novo paradigma arquitetural que é a Internet das Coisas (Internet of Things - IoT) (AMENTO et al., 2018; HSIEH et al., 2018; MORABITO, 2017; NIU et al., 2017). Sabemos que alguns trabalhos na área de IoT utilizam bancos de dados para armazenar informações, como (JIANG et al., 2014) que propõe uma estrutura de armazenamento de dados para IoT e (GUPTA et al., 2016) que coletam dados de sensores e os enviam para um servidor de banco de dados para que sejam analisados e mantidos de forma estática. Visto isso, durante o planejamento da arquitetura do sistema é importante considerar o gasto de recursos e realizar avaliações de desempenho dos contêineres junto a bancos de dados, como um modo de auxiliar na melhoria da qualidade de serviço de aplicações. Com relação ao planejamento da arquitetura, não são encontrados estudos com análises de desempenho aprofundadas, mas alguns trabalhos analisaram contêineres Docker com banco de dados relacionais, tais como (VELÁSQUEZ; MUÑOZ-ARCENTALES; RODRIGUEZ, 2018; XAVIER et al., 2016), que se concentraram na avaliação do desempenho da Rede e Disk I/O, respectivamente. No entanto, esses trabalhos também não realizam uma análise de sensibilidade aprofundada, tornando necessária uma investigação que utilize diversos fatores e níveis de forma conjunta, pois dessa forma é possível extrair informações mais precisas.

24.1.1 Definição do Problema

Avaliar o desempenho dos contêineres é importante para auxiliar administradores de infraestruturas computacionais a adequarem as arquiteturas de acordo com suas necessidades. Apesar de já ser uma tecnologia bastante conhecida, até o momento não existem trabalhos que realizam análises aprofundadas sobre o desempenho de contêineres em aspectos de planejamento de arquiteturas e armazenamento de dados, pois em alguns casos possui alta complexidade e alto custo financeiro.

24.1.2 Objetivos

24.1.2.1 Objetivo Geral

O objetivo geral deste trabalho é avaliar o desempenho do contêiner Docker sob a perspectiva de planejamento de arquiteturas e armazenamento de dados.

24.1.2.2 Objetivos Específicos

1. Desenvolver um modelo de Rede de Petri Estocásticas para analisar o desempenho de arquiteturas *Mobile Edge Computing* (MEC).
2. Identificar os fatores que mais impactam na eficiência de um Sistema de Gerenciamento de Banco de Dados (SGBD) sendo executado em contêineres.

24.1.3 Organização do trabalho

O restante desta seção se organiza da seguinte forma: A seção 24.2 contém alguns conceitos necessários para o entendimento deste trabalho; a seção 24.3 apresenta uma avaliação de desempenho baseada em Redes de Petri Estocásticas (SPN); a seção 24.4 apresenta uma avaliação de desempenho baseada em *Design of Experiments* (DOE); e a seção 24.5 apresenta as principais conclusões obtidas acerca dos estudos realizados;

24.2 Background

24.2.1 Computação Móvel na Borda

O surgimento de vários novos aplicativos de computação, como realidade virtual e ambientes inteligentes, tornou-se possível devido à disponibilidade de um grande pool de recursos e serviços em nuvem. No entanto, os aplicativos sensíveis a atraso apresentam alto nível de exigência para as infraestruturas computacionais convencionais. O paradigma da computação em nuvem não consegue atender aos requisitos de baixa latência, reconhecimento de local e suporte a mobilidade. Nesse contexto, o Mobile Edge Computing (MEC) foi introduzido para aproximar os serviços e recursos em nuvem aos usuários, aproveitando os recursos disponíveis nas redes de borda (AHMED; REHMANIB, 2017; WANG et al., 2017).

A MEC suporta diferentes opções de implementação, pois os servidores MEC podem ser localizados em diferentes locais dentro da rede de acesso de rádio, dependendo dos requisitos técnicos e de negócios (HU et al., 2015a). A MEC vem se tornando viável para lidar com a grande geração de dados que ocorre na borda da rede e obter *insights* que auxiliem na tomada de decisões em tempo real, além de fornecer flexibilidade para usuários que desejam conservação de energia em baixa latência ou vice-versa no processamento de dados visuais (TRINH; YAO, 2017). Arquiteturas MEC tendem a ser mais responsivas do que as arquiteturas de Mobile Cloud Computing (MCC). Existem disparidades entre arquiteturas MEC e MCC em termos de servidor de computação, distância até os usuários finais e latência típica, etc. Tais características justificam a criação de novos modelos analíticos específicos para arquiteturas MEC.

24.2.2 Redes de Petri Estocásticas

Redes de Petri (*Petri Net - PN*) são uma família de formalismos baseados em estado, apropriada para modelar diversos tipos de sistemas que possuam mecanismos de concorrência, assincronicidade, distribuição, determinísticos, ou estocásticos. Sua representação gráfica permite a visualização de atividades concorrentes e dinâmicas do sistema. Enquanto suas características matemáticas tornam possíveis criar equações de estado, equações algébricas ou outros modelos matemáticos que regem o comportamento do sistema (MURATA, 1989a).

A representação gráfica das PNs são formadas por: lugares (Figura ?? (a)) que correspondem às variáveis de estado; transições (Figura ?? (b)) que representam as ações e eventos do sistema; arcos (Figura ?? (c)) que presentam os fluxos de marcas pelo sistema; e marcas (*tokens*) ?? (d)), cujo conjunto representa o estado do sistema em um instante. A realização de uma ação ou evento (disparo de uma transição) no sistema está ligado a pré-condições, deve existir uma relação entre os lugares e a transição para que esta possa ou não realizar a ação (disparar). Após o disparo alguns lugares terão suas informações alteradas (Marcas), ou seja, levará a uma pós-condição.



Figura 232 – Elementos de uma Rede de Petri

Rede de Petri é uma técnica de modelagem que permite a representação de sistemas, utilizando como alicerce uma forte base matemática. Essa técnica possui a particularidade de permitir modelar sistemas paralelos, concorrentes, assíncronos e não-determinísticos. As Redes de Petri Estocásticas SPN são uma extensão das redes de Petri. As SPNs permitiram ligar as PNs a avaliação de desempenho, uma área normalmente baseada na abordagem de modelagem estocástica (MOLLOY, 1981).

Nas SPNs, as atividades que têm tempos associados, são representadas por transições temporizadas, que são representadas por retângulos brancos (Figura ?? (a)). O período de habilitação da transição corresponde ao tempo para realizar a atividade, e o disparo ao fim da atividade. Outra adição posterior das PNs, são os arcos inibidores (Figura ?? (b)), essa adição permite testar se um lugar não possui *tokens*. Com a presença do

arco a transição estará habilitada se a quantidade de *tokens* no lugar p associado ao arco for menor que o peso do arco n , ou seja $M(p) < n$ (MARSAN et al., 1994).



Figura 233 – Elementos adicionais em uma SPN

Tanto a modelagem visual, como a obtenção das métricas por análise numérica e análise estacionária podem ser realizadas por ferramentas de apoio como a ferramenta Mercury (SILVA et al., 2015). O presente trabalho utilizou tal ferramenta para modelar e executar as análises numéricas.

24.2.3 Design of Experiments

O Design of Experiments (DoE - Projeto de Experimentos) é uma metodologia estatística que foi aplicada no presente trabalho. O DoE foi amplamente utilizado com sucesso em vários campos, tais como otimização (GUNST, 1996), química (LAZIC, 2006) e engenharia de software (KUHN; REILLY, 2002). Existe uma quantidade considerável de livros sobre tal metodologia (SIEBERTZ; BEBBER; HOCHKIRCHEN, 2017; SELTMAN, 2012; ANTONY, 2006); Portanto, a metodologia geral aqui será descrita brevemente. Em cada parte da aplicação, será exemplificado cada caso metodológico para o estudo de caso.

O conceito geral de DoE foi criado por uma série de experimentos reais simulados em um sistema ou modelo de sistema sob observação. Em cada experimento, um ou vários parâmetros de projeto são alterados e o impacto no comportamento do sistema é avaliado. Quais parâmetros são alterados e como eles são alterados é definido usando um plano de experimento. O objetivo é obter o máximo de informações possíveis usando a menor quantidade de experimentos. O comportamento do sistema com base nas mudanças de parâmetros é observado usando conjuntos de saídas. No contexto do DoE, as saídas podem ser referenciadas como “indicadores de performance”, os parâmetros de projeto como “fatores” e os valores das configurações de “níveis”.

Para analisar o impacto de cada fator no sistema e a interação com os outros fatores, várias combinações de fatores precisam ser testadas e, portanto, exige um experimento. Devido à grande quantidade de combinações possíveis, este é um esforço muitas vezes inviável em termos de tempo e custos. O DoE oferece uma coleção de métodos - referidos como planos experimentais ou tabelas de design - para reduzir a quantidade de experimentos necessários para encontrar informações precisas com o menor número de experimentos possível. O tipo de plano usado depende do objetivo do experimento. Encontrar a tabela de projeto ideal, por exemplo, a menor quantidade de experimentos necessários para descrever o comportamento dos sistemas corretamente, tem sido objeto de intensa pesquisa. Usando um conjunto de métodos estatísticos de avaliação nos dados resultantes do experimento, o impacto, efeitos e interações dos fatores em relação aos indicadores de performance escolhidos é avaliado. Os resultados do experimento podem ser usados para formular um modelo substituto matemático, também chamado de metamodelo em disciplinas de engenharia (MILLER et al., 2014; THOMAS et al., 2014). Neste trabalho foi aplicada a metodologia DoE para avaliação da arquitetura proposta. Três tipos de gráficos são adotados usualmente em estudos com DoE: Gráfico de Pareto, Gráfico de Efeitos Principais e Gráfico de Intereração (ESPEJO, 2006).

24.2.4 Gráfico de Pareto

O gráfico de Pareto permite detectar qual o efeito da interação de fatores é mais importante para o processo ou estudo de otimização de projeto com o qual se deve lidar. Exibe os valores absolutos dos efeitos e desenha uma linha de referência no gráfico. Qualquer efeito que ultrapasse essa linha de referência é potencialmente importante. Um gráfico de Pareto é construído como na Figura 234, por exemplo. O gráfico mostra que os fatores B (tool geometry) e C (cutting angle) e a interação AC possuem maior impacto.

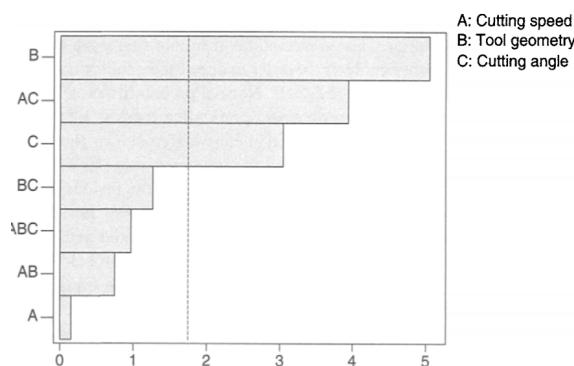


Figura 234 – Gráfico de Pareto com Efeitos Padronizados

24.2.5 Gráfico de Efeitos Principais

O “Gráfico de Efeitos Principais” é o gráfico dos valores médios de resposta em cada nível de um parâmetro de projeto ou variável de processo. Pode-se usar esse gráfico para comparar a força relativa dos efeitos de vários fatores. O sinal e a magnitude de um efeito principal nos diriam o seguinte:

- O sinal de um efeito principal nos diz a direção do efeito, ou seja, se o valor médio da resposta aumenta ou diminui.
- A magnitude nos diz a força do efeito.

Se o efeito do parâmetro de projeto ou processo é positivo, isso implica que a resposta média é maior em nível alto do que em nível baixo da configuração do parâmetro. Em contraste, se o efeito for negativo, isso significa que a resposta média em nível baixo de configuração do parâmetro é maior que em nível alto. A Figura 235 ilustra o efeito principal da temperatura na resistência à tração de uma amostra de aço. Como pode ser visto na Figura, a resistência aumenta quando as configurações da temperatura variam para o nível baixo (ou seja, 1 para 1). O efeito de um parâmetro de projeto ou processo (ou fator) pode ser calculado matematicamente usando a simples Equação 24.1:

$$E_f = \bar{F}_{(+1)} - \bar{F}_{(-1)} \quad (24.1)$$

Quando $\bar{F}_{(+1)}$ significa a resposta média na configuração de alto nível de um fator, e $\bar{F}_{(-1)}$ significa a resposta média na configuração de baixo nível de um fator.

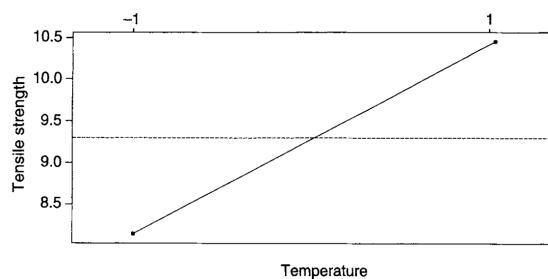


Figura 235 – Gráfico de Efeitos Principais da Temperatura na Resistência à Tração

24.2.6 Gráfico de Interação

O processo de “Interações”, identifica efeitos importantes e determina sua magnitude, logo, as interações entre os efeitos são cruciais. As interações ocorrem quando o efeito de um fator depende do nível de outro

fator. Uma medida de design sempre aborda vários fatores. Entender como esses fatores interagem em que magnitude permite para escolher a melhor combinação de medidas, revelando combinações de fatores com efeito cumulativo ou degradante. A interação entre os fatores A e B pode ser calculada usando a Equação 24.2.

$$I_{A,B} = \frac{1}{2}(E_{A,B(+1)} - E_{A,B(-1)}) \quad (24.2)$$

O $E_{A,B(+1)}$ é o efeito do fator 'A' no nível alto do fator 'B' e $E_{A,B(-1)}$ é o efeito do fator 'A' no nível baixo do fator 'B'.

Para determinar se dois parâmetros de processo estão interagindo ou não, pode-se usar uma ferramenta gráfica simples, porém poderosa, chamada de gráficos de interação. Se as linhas no gráfico de interação forem paralelas, não haverá interação entre os parâmetros do processo. Isso implica que a mudança na resposta média do fator 'A' não depende dos níveis do fator 'B'. Por outro lado, se as linhas não são paralelas, existe uma interação entre os fatores. Quanto maior o grau de afastamento de ser paralelo, mais forte o efeito de interação. Para interação sinérgica, as linhas no gráfico não se cruzam. Por exemplo, a Figura 236(a) é um exemplo de interação sinérgica.

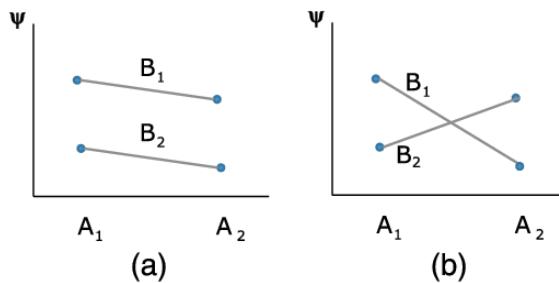


Figura 236 – Gráfico de Interações Sinérgica e Antagônica

Na interação antagônica, as linhas no gráfico se cruzam. Isso pode ser ilustrado na Figura 236(b). Neste caso, a mudança na resposta média para o fator A no nível 'A₁' é alta em comparação ao nível 'A₂'. As mudanças nos níveis do fator 'A' para a resposta média, indica uma dependência do fator 'A' em relação aos níveis do fator 'B'.

24.3 Avaliação de Desempenho de Computação Móvel na Borda Usando Redes de Petri Estocásticas

De acordo com o relatório apresentado pela Cisco Visual Networking Index, o número de dispositivos móveis chegará a 11,6 bilhões em 2020 (JUNG, 2016). A tendência de aumento no uso de dispositivos móveis é fundamentalmente impulsionada pelo aumento de usuários móveis e pelo desenvolvimento de aplicativos cada vez mais interativos e utilitários (KITANOV; MONTEIRO; JANEVSKI, 2016) (BECK et al., 2014). O crescimento do uso de smartphones e tablets teve um efeito extremo sobre as redes móveis trazendo grandes desafios para as companhias de telecomunicações (CAU et al., 2016).

As redes celulares devem suportar a todo custo a baixa capacidade de armazenamento, o alto consumo de energia, a baixa largura de banda e altas latências (ORSINI; BADE; LAMERSDORF, 2015). Com o objetivo de propor arquiteturas cada vez mais otimizadas para este contexto, surgiu a computação móvel na borda, do inglês *Mobile Edge Computing* (MEC). O principal objetivo da MEC é implantar os recursos ainda mais perto dos usuários — na borda da rede. Assim, a computação e armazenamento são executados mais próximos do local de origem, sempre na casa dos milissegundos (JARARWEH et al., 2016b). Por ser uma área ainda recente e com grandes desafios, algumas lacunas de pesquisa ainda devem ser exploradas, como é o caso da avaliação de desempenho de arquiteturas MEC.

Esta seção apresenta um modelo SPN para modelar uma arquitetura MEC onde os recursos de um único servidor são paralelizados com microsserviços sendo executados em múltiplos contêineres. O modelo

proposto neste trabalho permite avaliar o *trade-off* entre tempo médio de resposta (MRT) e utilização de recursos. Em resumo, as principais contribuições deste estudo são:

- um modelo analítico, que é uma ferramenta útil para administradores de sistemas avaliarem o desempenho de arquiteturas MEC, antes mesmo que sejam implantadas;
- um conjunto de análises numéricas com dados reais que provê um guia prático para análise de desempenho em arquiteturas MEC.

24.3.1 Trabalhos Relacionados

Alguns trabalhos relacionados propõem arquiteturas MEC em **cooperação com outras camadas**. No entanto, esses trabalhos não exploram diferentes configurações de recursos computacionais. (TONG; LI; GAO, 2016) propõem implantar servidores de nuvem na borda da rede e posicionar os servidores de forma hierárquica distribuídos geograficamente, de modo a utilizar a nuvem para atender às cargas de pico de requisições advindos da MEC. (CHEN et al., 2016) foca em um algoritmo de otimização no qual calcula onde executará a carga de trabalho (MEC ou nuvem). Os autores utilizam como validação aplicações genéricas que necessitam de alto nível de processamento. (LI; WU, 2018) foca na capacidade da MEC e da nuvem em processar grandes cargas de dados. O objetivo principal do trabalho é tentar diminuir a latência entre as duas camadas atribuindo pesos às complexidades das tarefas a serem executadas remotamente.

Alguns trabalhos focam no problema do **consumo de energia de dispositivos móveis** com auxílio de uma arquitetura MEC. (KE; S; L, 2016) estudou mecanismos de *offloading* investigando uma arquitetura MEC e redes heterogêneas 5G. Os autores formularam um problema de otimização para minimizar o consumo de energia observando tempo de processamento e transferência de dados. (TRINH; YAO, 2017) fez um estudo sobre o potencial da MEC para mitigar a limitação de bateria em dispositivos IoT. Os autores utilizaram uma aplicação de reconhecimento facial para demonstrar a viabilidade das políticas de decisão de *offloading*. (MAO; ZHANG; LETAIEF, 2016) também propôs um algoritmo de otimização de bateria em dispositivos móveis. O algoritmo incluiu frequências do ciclo da CPU do servidor MEC e taxa de latência. Uma vantagem desse algoritmo é que as decisões dependem apenas das informações instantâneas do lado do servidor sem exigir informações de distribuição da solicitação de tarefa. Diferentemente desses trabalhos, este estudo não analisa diretamente o gasto energético dos dispositivos clientes, no entanto a métrica MRT aqui adotada está proporcionalmente relacionada ao gasto energético dos dispositivos clientes. Quanto mais tempo a requisição demorar para executar maior será o consumo energético.

Os trabalhos mais relacionados ao presente trabalho tratam sobre o **planejamento da infraestrutura MEC**. (PREMSANKAR; TALEB, 2018) realizou uma avaliação experimental real com uma aplicação de um jogo eletrônico de alta interatividade, porém com a limitação de haver apenas um cliente móvel. O trabalho de (JARARWEH et al., 2016a) desenvolveu um simulador de uma arquitetura que incluía uma camada de Cloudlet e uma camada MEC visando aumentar a área de cobertura para os usuários móveis, onde os usuários podem realizar requisições com custos mínimos em termos de gasto energético e tempo de execução. Uma limitação é que os autores adotam apenas o número de requisições como parâmetro de entrada e também o fato de usar um simulador próprio sem explicações detalhadas. (LIU et al., 2016) adotaram um modelo de cadeias de Markov como processo de decisão sobre onde as tarefas devem ser executadas (localmente ou no servidor MEC). O modelo leva em consideração o estado de enfileiramento do *buffer* de tarefas, o estado de execução do dispositivo móvel e o estado da rede. No entanto os autores não consideram o servidor MEC com múltiplos nós paralelos, e assim, não aproveitando substancialmente do potencial do paralelismo do servidor. (BADRI et al., 2017) também utilizou cadeias de Markov, porém, com o objetivo de decidir onde executar requisições em múltiplas torres de comunicação MEC. O algoritmo leva em conta a movimentação dos usuários, o custo de comunicação entre usuários e servidores, o custo de execução em cada servidor e o custo de realocações. Este trabalho não foca em múltiplos servidores e sim no planejamento de uma infraestrutura mínima com um único servidor MEC com paralelismo através de múltiplos contêineres.

Diferente da nossa proposta, os trabalhos acima citados não exploram nível de utilização de recursos e apenas alguns consideram MRT. Todos os trabalhos são limitados em termos de parametrização das

avaliações, com no máximo observando três parâmetros de configuração da arquitetura. Nenhum dos trabalhos (com exceção de (PREMSANKAR; TALEB, 2018)) abordou aplicações com alto nível de interação. Ademais, nenhum dos trabalhos adotou Redes de Petri Estocásticas.

24.3.2 Arquitetura Base

A Figura 237 ilustra a arquitetura que propomos modelar e analisar seu desempenho. Tal arquitetura base é bastante adotada para descrever a infraestrutura MEC em diversos trabalhos (TRINH; YAO, 2017; YUANZHE; SHANGGUANG, 2018; LI; WU, 2018) com um servidor MEC visando processar fluxos de dados recebidos. Esses dados são gerados e enviados por aplicativos executados em dispositivos móveis, por exemplo, dados de aplicações de monitoramento da saúde do usuário, dados para a renderização de jogos, etc. No contexto deste trabalho focamos em aplicações com alto nível de interatividade com o usuário, incluindo periféricos como smartphones ou tablets.

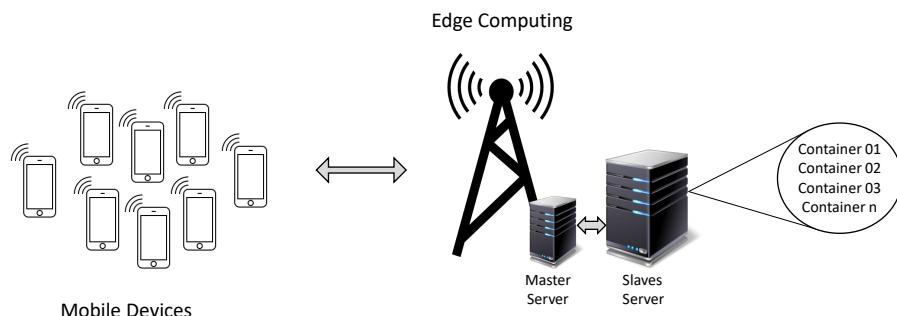


Figura 237 – Arquitetura MEC Base Adotada na Avaliação de Desempenho

Mais detalhadamente, na camada da *edge computing* temos dois servidores, um servidor *master* e um servidor com os nós *slaves* (que farão o processamento em si). Os nós *slaves* são microsserviços que são executados em contêineres. Neste trabalho, cada contêiner é configurado para executar em um núcleo do servidor. Portanto, se existirem 16 núcleos, serão executados 16 contêineres. O uso de contêineres na área de MEC ainda não é muito comum. O uso de contêineres permite maior flexibilidade para escalar o poder computacional da arquitetura de acordo com o volume de tarefas e também restrições de tempo de resposta de aplicações.

O servidor master é responsável por receber as requisições dos dispositivos móveis e distribuí-los entre os nós slaves. A princípio, o servidor master executa o serviço de gerenciamento com o serviço sendo executado em modo bare metal (sem virtualização) e com uso de threads. Porém, nada impede de também virtualizar o serviço do servidor master. Para gerenciar os contêineres consideramos que um orquestrador (ex.: Kubernetes¹ ou Swarm²) de contêineres deve ser utilizado visando maior confiabilidade e elasticidade dos recursos.

Como ilustra a Figura 238, existem 3 intervalos de tempo no fluxo de comunicação da arquitetura: (i) Arrival Delay (AD): intervalo de tempo entre as gerações das requisições; (ii) Distribution Delay (DD): intervalo de tempo para os trabalhos serem distribuídos ou encaminhados aos nós; e (iii) Processing Delay (PD): tempo de processamento das requisições pelos nós.

24.3.3 Proposta de Modelo SPN

Nessa seção, descrevemos a nossa proposta de modelo baseado em redes de Petri estocásticas (SPN) para representar a arquitetura que integra módulos na borda da rede apresentada na seção anterior. Ressaltamos

¹ Kubernetes: <https://kubernetes.io/>

² Swarm: <https://docs.docker.com/engine/swarm/>

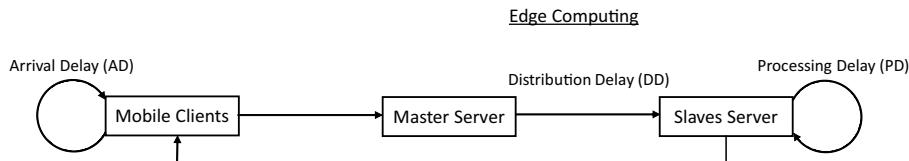


Figura 238 – Intervalos de Tempo na Comunicação dos Componentes da Arquitetura

que o objetivo desse modelo é possibilitar a checagem do desempenho de mudanças no sistema, antes mesmo que elas sejam implantadas. A Figura 244 apresenta o modelo SPN, composto de duas macro partes:

1. *Admission* que trata da geração das requisições;
2. *Edge* composto pelo servidor master e o servidor com nós slaves. O servidor master recebe dados e os distribui entre os nós slaves, que por fim retornam os resultados aos clientes;

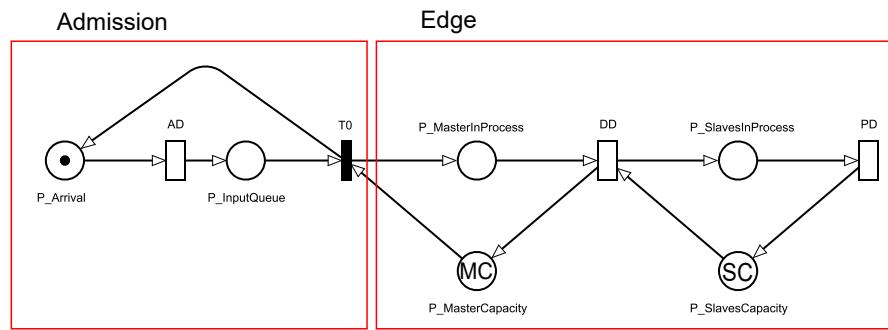


Figura 239 – Modelo SPN para uma Arquitetura Edge Computing

Os componentes são representados por elementos gráficos do tipo lugares (círculos), transição (barra preenchida), transições temporizadas (barras vazias) e marcações de lugares (círculos preenchidos). A Tabela 164 apresenta todos os elementos do modelo extensivamente. As transições temporizadas são parametrizadas com distribuições de probabilidade. O administrador do sistema deve informar essas distribuições de acordo com a literatura ou realizando medições e caracterizações do sistema.

Tabela 164 – Descrição dos Elementos do Modelo

Tipo	Elemento	Descrição
Lugares	P_Arrival	Espera por novas requisições
	P_InputQueue	Espera pela disponibilidade da fila
	P_MasterInProcess	Trabalhos na fila do servidor master
Transições Temporizadas	P_MasterCapacity	Capacidade do servidor master
	P_SlavesInProcess	Requisições na fila dos nós slaves da borda
	P_SlavesCapacity	Capacidade da borda
Marcações dos Lugares	AD	Tempo de chegada entre requisições (<i>Single Server</i>)
	DD	Tempo para o servidor master distribuir as requisições entre os nós slaves (<i>Infinite Server</i>)
	PD	Tempo gasto para processar o trabalho no nó slave(<i>Infinite Server</i>)
Marcações dos Lugares	MC	Capacidade máxima da fila de entrada do master
	SC	Capacidade de nós slaves disponíveis para processamento

Dada a visão geral do modelo, descrevemos agora o fluxo de processamento de dados entre seus componentes. A sub-rede *Admission* é composta por dois lugares **P_Arrival** e **P_InputQueue**, que representam a espera entre gerações de requisições e a aceitação dessas requisições na fila, respectivamente. Os tokens em **P_Arrival** e **P_InputQueue** representam qualquer tipo de requisição que envolva entrada de dados. Os tempos entre chegadas de requisições é atribuído à transição **AD**. Consideramos que os tempos entre disparos são exponencialmente distribuídos, essa suposição pode ser modificada, alterando essa distribuição. A transição **AD** não leva em conta as perdas provenientes da rede.

Quando **T0** é disparado, a sub-rede *Edge* é alcançada. A quantidade de tokens em **P_MasterInProcess** representa o enfileiramento de requisições na Edge. A marcação **MC** em **P_MasterCapacity** indica a quantidade de espaço de armazenamento temporário do servidor master, enfileirando as requisições. O enfileiramento ocorre quando não há a capacidade disponível para servir a requisição recém-chegada. Assim, caso haja espaço de armazenamento suficiente, um token é retirado de **P_InputQueue** e de **P_MasterCapacity**, alocando um token a **P_MasterInProcess**. Quando isto acontece, o lugar da entrada na subrede *Submission* (**P_Arrival**) é então habilitada, permitindo um novo disparo.

O disparo de **DD** representa o início da distribuição de requisições aos nós slaves. Estes disparos são condicionados à quantidade de nós disponíveis para processamento em **P_SlavesCapacity** (com marcação **SC**). A marcação **SC** indica a quantidade de nós disponíveis na borda da rede. À medida que as requisições vão sendo consumidas pelo lugar **P_SlavesInProcess**, os tokens são retirados do **P_SlavesCapacity**. Este fluxo significa que cada trabalho será alocado em um recurso à medida que for chegando.

PD representa o tempo gasto pelo nó slave para processar uma requisição. Quando **PD** é disparada, um token é retirado de **P_SlavesInProcess** e um token é retornado a **P_SlavesCapacity**. A transição **AD** possui semântica *single server* de uma distribuição exponencial, pois estamos considerando taxas de chegada exponencialmente distribuídas. Todas as outras transições possuem a semântica *infinite server*, então, cada trabalho é processado independentemente. É importante notar que o tempo de processamento depende muito da capacidade computacional de cada nó. Neste trabalho consideramos que todos os nós de cada camada possuem a mesma capacidade computacional.

O modelo proposto permite avaliar um número muito grande de cenários, pois o avaliador deve configurar 5 parâmetros (vide Tabela 164). Os parâmetros incluem: as 3 transições temporizadas, e as 2 marcações de lugares relacionados a recursos ou carga de trabalho. Qualquer alteração em um destes parâmetros pode impactar significativamente no tempo médio de resposta do sistema e no nível de utilização dos recursos. A variação das possibilidades de cenários considerando um grande número de fatores é o que torna este modelo a principal contribuição deste trabalho.

24.3.4 Métricas de Desempenho

Nesta seção, definimos métricas para avaliar a arquitetura de borda com base no modelo proposto. O tempo médio de resposta (**MRT**) pode ser obtido a partir da Lei de Little (LITTLE, 1961). A Lei de Little relaciona o número médio de requisições em progresso em um sistema (*RequestsInProgress*), a taxa de chegada de novas requisições (*ARR*) e o tempo médio de resposta (*MRT*). A taxa de chegada é o inverso do tempo de chegada. Considerando a transição para tempo entre gerações do modelo, temos que $ARR = \frac{1}{AD}$. Vale ressaltar que a Lei de Little requer um sistema estável, ou seja, que possua uma taxa de requisições menor que a taxa de processamento dos servidores. No modelo proposto pressupomos que a taxa de chegada real não necessariamente é a taxa de chegada efetiva, pois alguns trabalhos podem se perder, ou por termos filas finitas, serem descartados. Então, como recomendado pelo autor Jain (1990), a probabilidade de descartes representado pela variável *N_Discard* é subtraída. Portanto, a equação correspondente à Lei de Little para *MRT* utilizada no modelo proposto é expressa na Equação 24.3.

$$MRT = \frac{RequestsInProgress}{ARR \times (1 - N.Discard)} \quad (24.3)$$

A Equação 24.4 define *RequestsInProgress*. Para calcular o número de requisições em progresso no sistema, precisa-se somar a quantidade de tokens em cada um dos lugares que representam uma requisição em andamento. Na Equação 24.4, *Esp(Lugar)* representa a esperança estatística de existir tokens em “Lugar”,

onde $Esp(Lugar) = (\sum_{i=1}^n P(m(Lugar) = i) \times i)$. Em outras palavras, $Esp(Lugar)$ indica qual a quantidade média esperada de tokens que ocupam aquele Lugar.

$$\text{RequestsInProcess} = Esp(P_MasterInProcess) + Esp(P_SlavesInProcess) \quad (24.4)$$

A equação 24.5 define $N_Discard$. Para calcular o descarte é necessário existir token na fila de entrada ($P_ArrivalQueue$) e não restar mais nenhum recurso disponível dentro das sub-redes. $P(Lugar = n)$ calcula a probabilidade de existirem n tokens em “Lugar”.

$$N_Discard = P((P_InputQueue = 1) \wedge (P_MasterCapacity = 0) \wedge (P_SlavesCapacity = 0)) \quad (24.5)$$

Por fim, além do MRT nós calculamos também a probabilidade de utilização dos recursos. A utilização do master é dada pela Equação 24.6, e a utilização dos nós slaves é calculada pela Equação 24.7. A utilização é obtida dividindo o número de tokens do lugar correspondente aos elementos em uso dividido pela capacidade total daqueles recursos.

$$U_Master = \frac{Esp(P_MasterInProcess)}{MC} \quad (24.6)$$

$$U_Slaves = \frac{Esp(P_SlavesInProcess)}{SC} \quad (24.7)$$

24.3.5 Análises Numéricas

Esta seção apresenta duas análises numéricas a partir da resolução do modelo SPN proposto visando analisar MRT e Utilização. No trabalho de (PREMSANKAR; TALEB, 2018) (que iremos chamar de “trabalho de referência”) os autores avaliaram uma arquitetura MEC com um único dispositivo móvel como cliente através de experimentos em um ambiente real com execução de serviços em contêineres. O trabalho de referência avaliou um jogo 3D chamado Neverball onde o jogador deve inclinar o piso para controlar a bola fazendo com que ela colete moedas e alcance um ponto de saída antes que o tempo acabe. Extraímos do trabalho de referência os parâmetros de entrada do modelo. Portanto, este estudo evolui o trabalho dos autores pois nós agora analisamos numericamente cenários com múltiplos usuários considerando o jogo com uma resolução de 800x600 pixels. O parâmetro adotado do trabalho de referência correspondendo ao tempo de processamento (PD) de uma requisição foi de 24ms. Adotamos o tempo de 5ms para a distribuição das requisições (transição DD). Neste estudo estabelecemos que o servidor master tem uma restrição de atender no máximo 40 requisições paralelamente, ou seja setamos a marcação MC com valor 40.

O modelo permite uma ampla variedade de parametrizações. Na presente análise variamos dois parâmetros, o intervalo de tempo entre chegadas de requisições (AD) e a capacidade de recursos do servidor com nós slaves (SC). O valor de AD foi variado entre 1ms até 10ms com acréscimos de 0.5ms. A variável SC foi configurada com três possibilidades (8, 16 e 32), correspondendo a opções sobre a quantidade de núcleos de um servidor. Obviamente todos estes parâmetros poderiam ser variados de outras formas, por exemplo, o número de nós slaves poderiam não estar atrelados a número de núcleos, podendo adotar por exemplo SC na casa dos milhares. Adotando os parâmetros acima mencionados, a seguir apresentamos os resultados considerando as métricas MRT, descarte, utilização do servidor master, e utilização do servidor com nós slaves.

A Figura 240 apresenta os resultados para o tempo médio de resposta. A princípio espera-se que quanto maior o intervalo de tempo entre chegadas (AD) menor seja o valor de MRT, pois o sistema estará mais ocioso com menos chegadas de requisições. Espera-se também que quanto maior for a quantidade de recursos slaves (SC) menor será o valor de MRT pois se aumentará a disponibilidade de recursos, aumentando-se o nível de paralelismo. Estes dois comportamentos são facilmente observados nos resultados quando os valores de descartes de trabalhos são desprezíveis no modelo (que podem ser observados na Figura 243), para esses intervalos o MRT decresce até o tempo mínimo para executar trabalhos sem enfileiramento no sistema.

No entanto, para os casos que o descarte é presente no sistema, observamos o aumento do MRT até um pico para que ele decresça, este comportamento é decorrente do valor limitante de recursos no sistema, onde parte dos trabalhos entrantes são descartados quando não há mais recursos, limitando a variação

do MRT ao tempo entre chegadas, como pode ser deduzido da lei de Little ([JAIN, 1990](#)), portanto o tempo médio entre saídas irá aumentar junto com o tempo médio entre chegadas até o pico, para $SC = 8$ foi em $AD = 1.5\text{ms}$ e para $SC = 16$ foi de $AD = 3.0\text{ms}$. Ao atingir esses tempos entre chegadas, a quantidade de trabalhos dentro do sistema começa a reduzir drasticamente, reduzindo o MRT mesmo com o aumento do AD. É importante ressaltar que consideramos no MRT a taxa de chegada efetiva, isto é, ajustando o seu valor com a probabilidade de descarte.

O valor de MRT para $SC = 32$ é extremamente baixo, mesmo para um intervalo entre chegadas de 1ms. Comparando $SC = 16$ e $SC = 32$, temos que a partir de 2.5ms os MRTs se igualam, e a partir de 5.5ms a configuração $SC = 8$ também apresenta o mesmo resultado médio. Portanto, caso o contexto real apresentasse um AD de 5.5ms, um servidor de 8 núcleos atingiria o mesmo desempenho do que servidores mais potentes, portanto este trabalho pode auxiliar no dimensionamento adequado da escolha de servidores, identificando a melhor opção de desempenho e custo para uma carga de trabalho esperada.

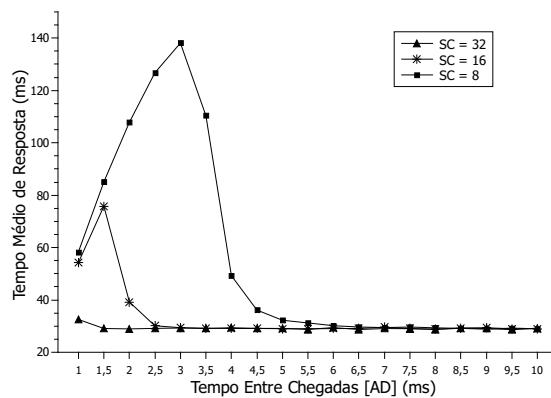


Figura 240 – Tempo Médio de Resposta (MRT)

A Figura 241 apresenta o nível de utilização do servidor master. O servidor master é o primeiro componente que a requisição atinge ao entrar na camada MEC. Para as configurações $SC = 8$ e $SC = 16$, o nível de utilização gira em torno de 100% nos menores valores de AD, posteriormente este valor cai. Para $SC = 32$, mesmo com $AD = 1.0\text{ms}$, o valor de utilização chega a apenas 20%. A partir de $AD = 5.5\text{ms}$ as três configurações possuem valores semelhantes e próximos a 0%. O administrador do sistema deve considerar se deseja níveis altos ou baixos de ociosidade para seu servidor master. A Figura 242 mostra o nível de utilização do servidor com nós slaves. Quanto maior o número de recursos, menor o nível de utilização dos nós slaves. À medida que aumenta-se AD, o nível de utilização decai de forma suave nos três casos. No entanto, esta queda somente inicia-se em $AD = 3.0\text{ms}$ para $SC = 8$ e em $AD = 1.5\text{ms}$ para $SC = 16$. Até estes pontos, o nível de utilização gira em torno de 82%, o que causa o comportamento do MRT explicado anteriormente.

A Figura 243 apresenta a probabilidade de descarte de novas requisições. Para $SC = 32$ a probabilidade de descarte é igual a zero. Portanto, caso seja possível adquirir um servidor com 32 núcleos não haverá descarte independente do intervalo entre chegadas de requisições. Para $SC = 8$ e $SC = 16$ somente a partir de $AD = 2.0\text{ms}$ e $AD = 4.0\text{ms}$ as probabilidades de descarte tendem a zero. Estes intervalos iniciais com descarte estão diretamente relacionados ao alto nível de utilização apresentados por ambos os servidores, impactando diretamente no tempo médio de resposta do sistema. Portanto, qualquer análise estocástica realizada com o modelo proposto deve observar as quatro métricas em questão para obter uma visão completa do comportamento do sistema. Também é possível identificar os limites de funcionamento do sistema, ou seja, até quantos trabalhos podem ser perdidos sem que comprometa a utilidade do sistema.

24.3.6 Resumo da seção

A Computação Móvel na Borda (MEC) surgiu como uma alternativa para diminuir a latência da rede, fazendo com que o processamento do fluxo de dados ocorresse mais próximo dos usuários mobile. No entanto, a

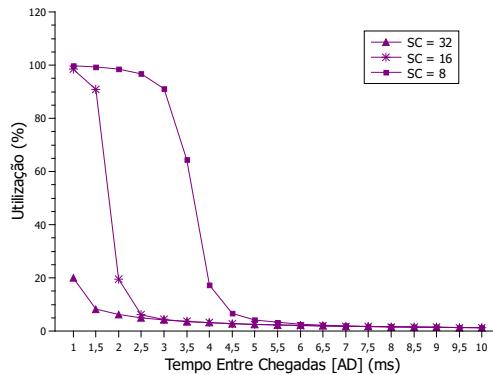


Figura 241 – Utilização do Servidor Master

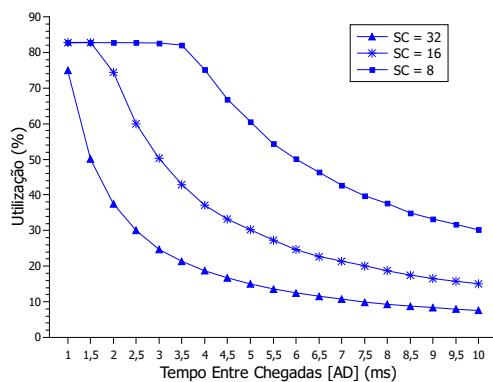


Figura 242 – Utilização do Servidor Slave

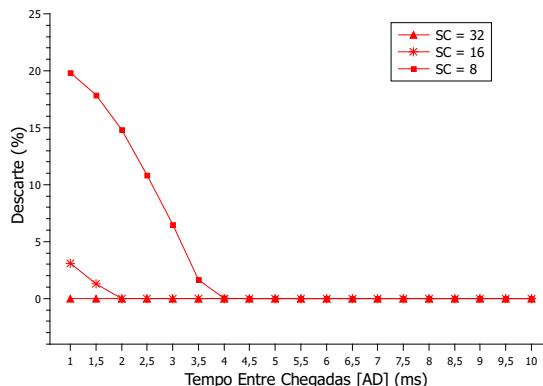


Figura 243 – Probabilidade de Descarte

configuração dos servidores pode influenciar diretamente no desempenho da arquitetura MEC. Esta seção propôs um modelo de Rede de Petri Estocástica (SPN) para modelar tal cenário e analisar seu desempenho, considerando diversos parâmetros que podem afetar diretamente no Tempo Médio de Resposta (MRT) e no Nível de Utilização. A seção apresenta também análises numéricas com valores de entrada reais que servem como um guia prático para auxiliar administradores de infraestruturas computacionais a adequar suas arquiteturas, encontrando o *trade-off* entre o MRT e o nível de utilização. A análise numérica permitiu ver

que a taxa entre chegadas possui grande impacto sobre o sistema. Para valores muito pequenos observou-se níveis de descarte, o que pode comprometer o resultado das outras métricas, principalmente do MRT.

24.4 Uma Avaliação de Desempenho de Contêineres Docker Executando Diferentes SGBDs Relacionais

A plataforma Docker vem sendo bastante utilizada em Internet das Coisas (Internet of Things - IoT). Alguns trabalhos nessa área utilizam bancos de dados para armazenar informações, como ([JIANG et al., 2014](#)) e ([GUPTA et al., 2016](#)). Visto isso, é importante realizar avaliações de desempenho de contêineres junto a bancos de dados, como uma forma de auxiliar na melhoria da qualidade de serviço de aplicações. Alguns trabalhos analisaram contêineres Docker com banco de dados relacionais, mas não realizaram uma análise de sensibilidade profunda, tornando necessária uma avaliação com diversos fatores e níveis de forma conjunta, para extrair informações mais precisas.

Esta seção apresenta uma análise sobre o uso de contêineres Docker executando banco de dados relacionais. Para tanto, foi realizada uma avaliação de desempenho baseada em análise de sensibilidade sobre o contêiner Docker em comparação com diferentes bancos de dados. Foram considerados os seguintes fatores: SGBD utilizado, número de tabelas e número de tuplas. E como variáveis dependentes, foram adotadas as seguintes: Tempo de Execução para processamento dos dados e Nível de Falha de requisições. Assim, esta seção tem como principal contribuição o provimento de indicativos de qual banco de dados relacional deve ser adotado junto ao Docker visando um melhor desempenho.

24.4.1 Trabalhos Relacionados

Esta seção apresenta alguns trabalhos relacionados ao presente estudo, ou seja, avaliaram o desempenho do contêiner Docker em conjunto com algum SGBD.

Os autores em ([MORABITO, 2016](#)) avaliaram o uso de contêineres Docker em ambientes restritos. Os autores avaliaram por meio de diferentes ferramentas de benchmark, o desempenho do Docker quando executado em um computador de placa única, no caso o Raspberry Pi (RPi2). Os resultados mostram um impacto quase insignificante da camada de virtualização de contêiner em termos de desempenho, se comparado à execução nativa.

Os autores em ([BARIK et al., 2016](#)) avaliaram o desempenho de VMs e contêineres simultaneamente em computadores tradicionais. Os autores mostram que contêineres têm algumas deficiências nas áreas de segurança e isolamento, pois proporciona um isolamento de segurança mais baixo do que a virtualização.

Os autores em ([VELÁSQUEZ; MUÑOZ-ARCENTALES; RODRIGUEZ, 2018](#)) apresentam uma análise dos diferentes sistemas de banco de dados para armazenar as informações coletadas por uma rede de sensores sem fio. Diferentes testes são apresentados para avaliar SGBDs relacionais (MySQL, PostgreSQL) e No-SQL (MongoDB, Couch, Neo4J). Os SGBDs foram implantados em contêineres Docker. O objetivo do trabalho foi fornecer informações necessárias para a correta escolha da base de dados para esse tipo de sistema. Os resultados indicaram que o melhor SGBD analisado para a ingestão de dados por segundo é o PostgreSQL, mas, para a quantidade de armazenamento total, o MongoDB obteve melhor desempenho.

Os autores em ([SHIRINBAB; LUNDBERG; CASALICCHIO, 2017](#)) apresentam uma extensa comparação de desempenho entre o contêiner VMware e o Docker, enquanto executa o Apache Cassandra. O Apache Cassandra lidera o ranking de SGBD distribuído NoSQL quando se trata de plataformas de Big Data. Como linha de base para comparações, os autores avaliaram o Cassandra focando na infraestrutura física. O estudo mostrou que o Docker tinha uma sobrecarga menor em comparação com o VMware ao executar o Cassandra. O desempenho do Cassandra na infraestrutura com Docker era tão bom quanto no ambiente sem virtualização alguma.

Todos os trabalhos acima destacados utilizaram a ferramenta Docker como hospedeiro para executar alguma instância de SGBD. Pôde-se verificar várias conclusões comparativas entre bancos de dados relacionais e até não relacionais. No entanto, nenhum dos trabalhos observados realizaram uma análise de sensibilidade sobre fatores que interferem no desempenho do Docker junto ao SGBD. A Tabela 165 resume as comparações.

Tabela 165 – Trabalhos Relacionados

Trabalho Relacionado	Métricas	SGBD	Hardware	Análise de Sensibilidade
(MORABITO, 2016)	CPU, Memória, Tipo de Transação	MySQL	Raspberry	Não
(BARIK et al., 2016)	Segurança	MySQL	Computador Desktop	Não
(SHIRINBAB; LUND-BERG; CASALICCHIO, 2017)	CPU, Tipo de Transação	Apache cassandra	Computador Desktop	Não
(VELÁSQUEZ; MUÑOZ-ARCENTALES; RODRIGUEZ, 2018)	Segurança	MySQL, PostgreSQL, MongoDB, Couch, Neo4J	Computador Desktop	Não
O presente trabalho	Tempo de Execução, Falhas	MySQL, PostgreSQL, SQLServer	Computador Desktop	Sim

24.4.2 Análise de Sensibilidade do Contêiner Docker e SGBDs Relacionais

Esta seção descreve os experimentos realizados para analisar os fatores com maior influência no desempenho de contêineres com SGBDs relacionais, baseado em duas métricas: tempo de execução e nível de falha na requisição.

24.4.3 Desenho Experimental

O ambiente de experimentos foi composto por dois computadores conectados à mesma rede: o Computador A foi utilizado para hospedar o contêiner Docker, enquanto o Computador B foi utilizado para enviar as requisições de acesso aos dados presentes no contêiner. A Tabela 166 apresenta as configurações das respectivas máquinas.

Tabela 166 – Configurações dos Computadores A e B

	Processador	Memória	Disco Rígido	Sistema Operacional
Computador A	Intel i5	8 GB	1 TB	Linux Ubuntu 18.04 LTS
Computador B	Intel i3	4 GB	500 GB	Linux Ubuntu 16.04 LTS

Para realizar a avaliação de desempenho, duas métricas foram definidas: tempo de execução e nível de falha na requisição. Para a avaliação da métrica de tempo de execução, foi realizado o *Design of Experiments* (DoE), com três fatores estabelecidos: SGBD, número de tabelas e número de tuplas da tabela. O SGBD possui três níveis, que foram escolhidos por serem alguns dos mais populares: MySQL (v. 8.0.12), PostgreSQL (v. 11.0) e SQL Server (v. 2017). Vale ressaltar que as configurações padrão dos SGBDs foram mantidas, e a consulta realizada buscava os dados de todas as colunas das tabelas (ID e NOME). O fator número de tabelas se refere à quantidade de tabelas que serão consultadas durante o experimento e possui dois níveis: 1 e 2. Já o fator número de tuplas é referente à quantidade de tuplas que cada tabela consultada possui e foram determinados três níveis: 1000, 3000 e 5000. A escolha do número de tuplas e tabelas foi baseada em sucessivos testes com níveis de fatores mínimos para garantir um controle sobre o test-bed. Nos experimentos em que são utilizadas duas tabelas, foi realizada a operação de junção, e o número de tuplas das mesmas devem ser iguais. A Tabela 167 apresenta os fatores e níveis escolhidos para a realização do DoE utilizando a métrica de tempo de execução.

Com todos os fatores e níveis definidos, deve-se combinar os fatores e níveis para definir como os experimentos deverão ser realizados. A Tabela 168 apresenta as combinações possíveis entre os fatores e níveis.

Uma outra avaliação de desempenho foi realizada para a análise do nível de falha na requisição, com o objetivo de identificar qual é o banco de dados mais eficaz, ou seja, qual deles possui o menor nível de

Tabela 167 – Fatores e níveis do DoE para a métrica tempo de execução

SGBD	Nível 1	Nível 2	Nível 3
	MySQL	PostgreSQL	SQL Server
Número de Tabelas	1	2	-
Número de Tuplas	1000	3000	5000

Tabela 168 – Combinações de fatores e níveis para a métrica tempo de execução

Nº	SGBD	Número de Tabelas	Número de Tuplas
1	MySQL	1	1000
2	MySQL	1	3000
3	MySQL	1	5000
4	MySQL	2	1000
5	MySQL	2	3000
6	MySQL	2	5000
7	PostgreSQL	1	1000
8	PostgreSQL	1	3000
9	PostgreSQL	1	5000
10	PostgreSQL	2	1000
11	PostgreSQL	2	3000
12	PostgreSQL	2	5000
13	SQL Server	1	1000
14	SQL Server	1	3000
15	SQL Server	1	5000
16	SQL Server	2	1000
17	SQL Server	2	3000
18	SQL Server	2	5000

falha. Para esse experimento foram utilizados os mesmos SGBDs do DoE para a métrica tempo de execução: MySQL, PostgreSQL e SQL Server. Porém, o número de tuplas e número de tabelas foram fixados em 5000 tuplas e uma única tabela, pois de acordo com sucessivos testes com níveis de fatores mínimos realizados para essa avaliação, foi constatado que a alteração desses fatores não influenciam na métrica em questão.

Na métrica tempo de execução é analisada a média do tempo total (em milissegundos) de duração do experimento. Para a análise do nível de falha na requisição é observada a média da porcentagem de erro encontrada durante o experimento, ou seja, a porcentagem de solicitações que não foram atendidas. Os valores para ambas as métricas foram coletados de acordo com os dados disponibilizado pela ferramenta de teste de carga utilizada, o JMeter³. Escolhemos essa ferramenta, pois a mesma foi utilizada em alguns trabalhos relacionados a este, como (GILLAN et al., 2018) e (GAUR; JOSHI; SRIVASTAVA, 2017).

Para o cálculo do tempo de execução, configurou-se o Computador B com o JMeter, para que a mesma enviasse requisições de leitura para o SGBD presente no Docker, instalado no Computador A. Para tanto, o JMeter foi configurado para gerar requisições simultâneas de leitura emulando um grupo de 100 usuários. Já na avaliação da métrica de nível de falha na requisição, o grupo de usuários sofreu variações e os teste foram realizado com 500, 1000, 1500, 2000 e 2500 threads simultâneas. Em ambas as avaliações, foram realizadas 100 réplicas de experimento para cada combinação.

24.4.4 Resultados Obtidos

Esta seção apresenta os resultados obtidos de acordo com os dados coletados no ambiente de testes. A Figura 244 apresenta o gráfico de Pareto para os fatores relacionados a métrica tempo de execução. Quando um fator possui alto impacto nos testes, significa que ao alterar o nível dele são obtidos valores bem distintos. De acordo com os valores do p-value encontrados, os efeitos do fator SGBD possui maior relevância dentre os

³ <https://jmeter.apache.org/>

fatores deste estudo, portanto a escolha do SGBD a ser utilizado é determinante na eficiência do contêiner, com relação ao tempo de execução.

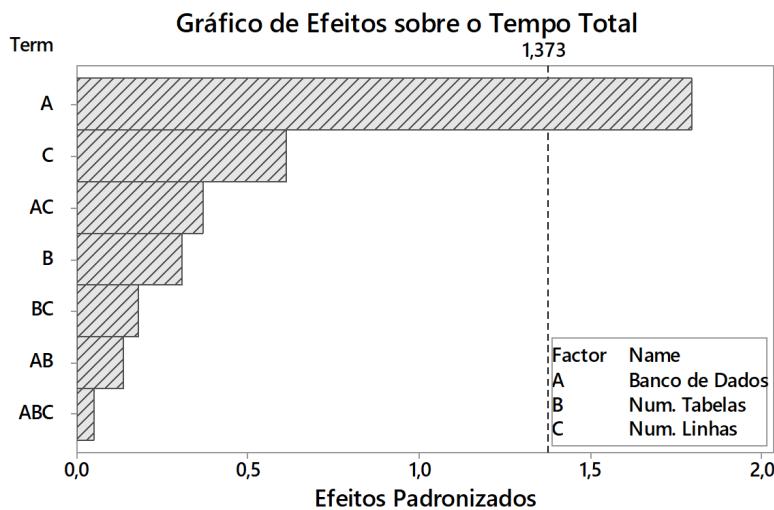


Figura 244 – Influência dos fatores na métrica tempo de execução

A Figura 245 apresenta o gráfico de efeitos principais para a métrica tempo de execução. O gráfico representa o tempo médio para a realização dos testes de cada nível para cada um dos fatores definidos. Nesse gráfico, quanto mais horizontal for a linha, menos influência tem aquele fator, pois significa que os diferentes níveis do fator influenciam no resultado final de forma semelhante.

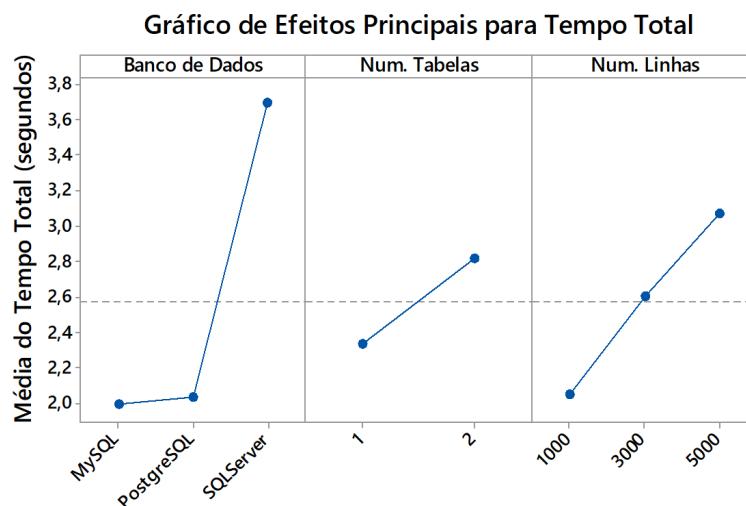


Figura 245 – Efeitos principais para Tempo de Execução

Nesse estudo, através do gráfico de efeitos principais, pode-se concluir que todos os níveis dos fatores definidos interferem na métrica tempo de execução. O SGBD e o número de tuplas são os fatores que possuem maior efeito. Sobre o fator SGBD, o MySQL atingiu a melhor média de tempo de execução (1,993ms), com o PostgreSQL bem próximo (2,031ms), enquanto o SQL Server alcançou uma média de tempo muito maior que ambos (3,692ms). Sendo assim, pode-se dizer que o SGBD mais eficiente foi o MySQL, visto que alcançou a menor média de tempo de execução de leitura.

Com relação ao fator número de tabelas, percebe-se que o fator influencia no tempo médio total de acordo com a sua quantidade. Ao utilizar duas tabelas, o número de dados manipulados é dobrado, o que aumenta o tempo utilizado durante os testes. Por fim, o fator número de tuplas influencia de forma que à medida que a quantidade de tuplas aumenta, o tempo médio de leitura também aumenta. Quanto maior o número de tuplas, mais dados são manipulados, tornando necessário um tempo maior para a realização dos experimentos.

A Figura 246 exibe o gráfico de interações para cada combinação possível entre os fatores. Com esse gráfico, é possível analisar quais níveis possuem maior interferência no resultado final dos experimentos. Na primeira interação nota-se que na relação SGBD e tempo de execução, o número de tabelas influenciou no resultado. Percebe-se que ao manipular duas tabelas é obtido um tempo médio maior que o tempo de manipulação para apenas uma tabela, independente do SGBD. Da mesma forma, utilizando apenas uma tabela, apesar do desempenho ser semelhante ao caso anterior, o tempo médio total é inferior. Na segunda interação, percebe-se que o Número de Tuplas interfere no resultado da relação entre o SGBD e o Tempo de Execução. É possível constatar que o Tempo de Execução aumenta de acordo com o Número de Tuplas, independente do SGBD utilizado. No entanto, a influência do Número de Tuplas que o SQL Server sofreu foi inferior aos demais. Na última interação é analisada a relação entre o Número de Tabelas e o Número de Tuplas. Conclui-se que a influência do Número de Tabelas no Tempo de Execução cresce de acordo com o aumento do Número de Tuplas. Tal resultado acontece pois quanto maior o Número de Tuplas nas tabelas, mais dados serão manipulados, o que exige maior processamento e consequentemente maior quantidade de tempo.

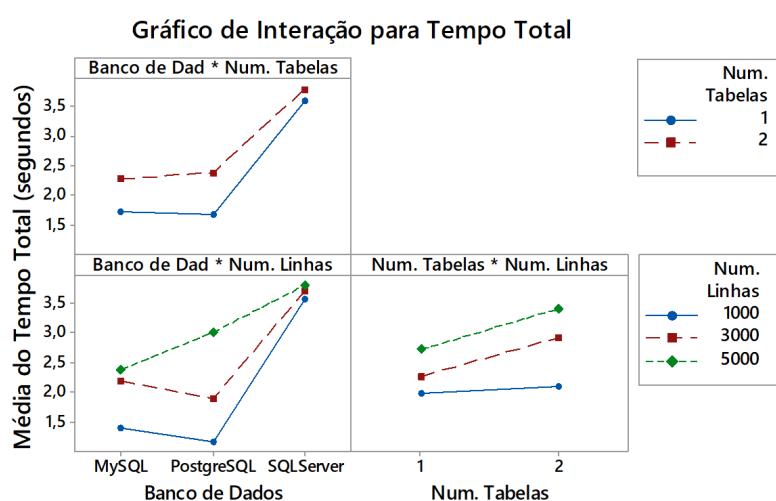


Figura 246 – Interações dos fatores para Tempo de Execução

A Figura 247 apresenta o gráfico de otimização para a métrica tempo de execução. Esse gráfico mostra o efeito de cada fator na resposta do experimento. Os dados na terceira linha, que são exibidos no topo do gráfico e as linhas verticais representam as configurações atuais do fator, ou seja, os níveis que estão sendo analisados. O valor de 'y' e as linhas na horizontal representam o resultado para o nível do fator atual. Analisando o gráfico é perceptível que para obter o menor tempo possível (1,150ms), a melhor configuração para os fatores é o Banco de Dados do SGBD PostgreSQL, utilizando uma única tabela de 1000 tuplas.

A Figura 248 apresenta o gráfico de barras que exibe a porcentagem do nível de falha na requisição de cada SGBD, de acordo com a quantidade de threads simultâneas. Analisando o gráfico, percebe-se que o SQL Server obteve a menor porcentagem de falha. Já o MySQL e o PostgreSQL obtiveram porcentagem de falha semelhantes entre eles, porém muito superior ao SQL Server. Acreditamos que a justificativa para esse resultado seja o fato de que por configuração padrão, o número máximo de conexões do MySQL e PostgreSQL é de 100 usuários simultâneos, enquanto o SQL Server esse número é dinâmico, podendo chegar a 32.767.



Figura 247 – Gráfico de otimização para tempo de execução

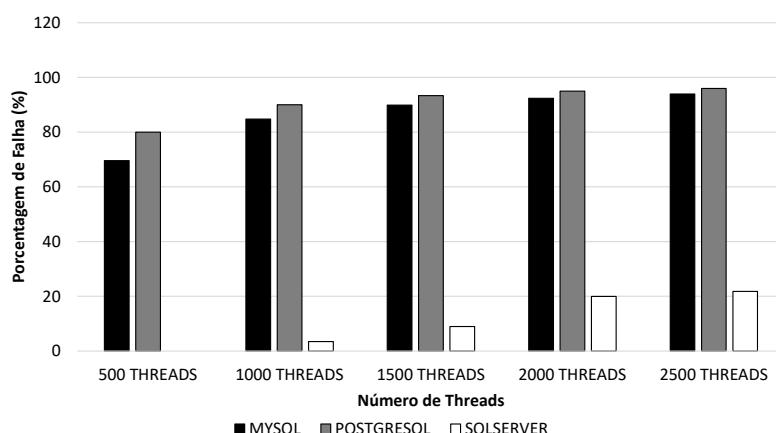


Figura 248 – Porcentagem do nível de falha na requisição

A Figura 249 exibe o gráfico de barras que representa o tempo total dos testes para nível de falha na requisição, os valores para essa métrica foram coletados considerando o tempo médio das requisições bem sucedidas. Os SGBDs MySQL e PostgreSQL alcançaram os menores tempos de testes, e assim como na porcentagem do nível de falha na requisição, os resultados foram próximos um do outro. Nota-se também que o SQL Server utilizou uma quantidade de tempo muito superior aos demais, mas após observar as Figuras 248 e 249, percebe-se que apesar disso foi o SGBD que proporcionou menor número de falhas. O MySQL e o PostgreSQL mesmo com o bom desempenho relacionado ao tempo total dos testes, apresentaram grande porcentagem de falhas.

A Figura 250 mostra as combinações que obtiveram melhor desempenho para as métricas analisadas: tempo de execução e nível de falha na requisição. As combinações foram feitas com os melhores níveis de cada fator, de acordo com as métricas utilizadas.

Sabendo quais são as melhores combinações, foram definidos dois cenários com a finalidade de identificar qual combinação obtém melhor desempenho, de acordo com as necessidades de cada um:

- **Cenário 1:** uma aplicação de pequeno porte precisa acessar o Banco de Dados para selecionar usuários cadastrados. Nesse caso, para obter o melhor desempenho é importante que o tempo utilizado seja o menor possível. Sendo assim, é recomendado o PostgreSQL, pois é mais eficiente com pequeno Número de Tuplas na tabela.
- **Cenário 2:** uma aplicação de porte maior e com um número de usuários superior que o cenário 1 precisa acessar o Banco de Dados para selecionar usuários cadastrados. Nesse caso, o mais importante é garantir a consistência dos dados, evitando falha na requisição dos mesmos durante a consulta.

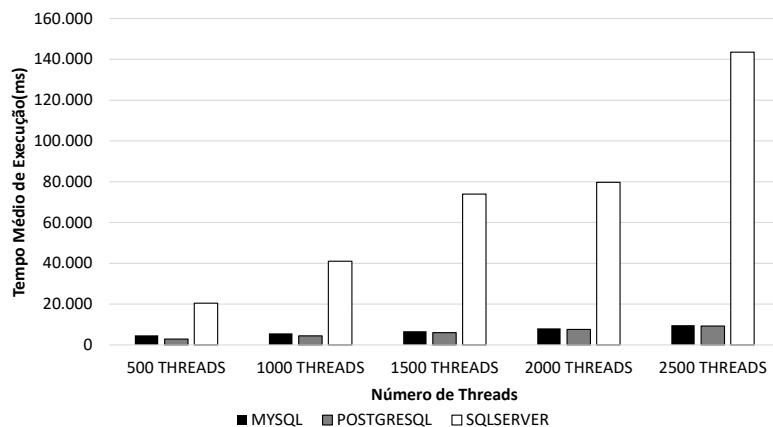


Figura 249 – Tempo dos testes para nível de falha na requisição



Figura 250 – Melhores combinações obtidas

Assim, é recomendada a utilização do SQL Server, pois independente de levar mais tempo para realizar a tarefa, é o SGBD que possui menor nível de falha na requisição.

24.4.5 Resumo da seção

As máquinas virtuais trouxeram benefícios como a possibilidade de prover tolerância a falha e balanceamento de carga. Porém, apesar das vantagens, as VMs causam perda de eficiência. Como alternativa as VMs, os contêineres se apresentam como uma solução mais leve. Este trabalho realizou uma avaliação de desempenho baseada em análise de sensibilidade com o objetivo de identificar os fatores que mais impactam na eficiência de um SGBD utilizando Docker como contêiner. Nas avaliações executadas foram observados alguns fatores, são eles: SGBD, número de tabelas e número de tuplas. De acordo com o que foi analisado,

nota-se que as configurações que proporcionam melhor desempenho é relativo à métrica escolhida. Quando se trata de menor tempo de execução, a melhor combinação encontrada dentre os fatores e níveis analisados foi o uso do PostgreSQL utilizando uma única tabela com 1000 tuplas, e a pior combinação foi o uso do SQL Server com 2 tabelas e 5000 tuplas. Já para menor nível de falha na requisição, a configuração que possibilitou melhor desempenho foi o SQL Server com 500 threads simultâneas, e a pior combinação foi o PostgreSQL com 2500 threads simultâneas.

24.5 Conclusão

Este trabalho propôs uma avaliação de desempenho do contêiner docker, para isso, foram desenvolvidos um modelo SPN para representar e avaliar o desempenho de uma arquitetura MEC básica, e um DoE para avaliar as melhores combinações de fatores para um banco de dados. O modelo permite estimar o Tempo Médio de Resposta (MRT) e o nível de utilização dos recursos na borda da rede. O avaliador pode configurar até 5 parâmetros de entrada no modelo, o que permite um alto nível de flexibilidade de avaliação. O DoE permite identificar a melhor configuração para consultas em um banco de dados instalado no contêiner docker. Para isso, foram realizadas avaliações de desempenho para as métricas: tempo de execução e nível de falha na requisição. Nessas avaliações foram observados alguns fatores, são eles: SGBD, número de tabelas e número de tuplas.

Na primeira avaliação foi executada uma análise numérica baseando-se em dados reais de um artigo de referência que executou experimentos simplificados. A análise numérica permitiu observar o comportamento das quatro métricas (MRT, descarte, utilização master, e utilização slave) em função da variação do tempo entre chegadas de novas requisições. A análise numérica permitiu concluir que a taxa entre chegadas é um parâmetro de grande impacto sobre o sistema. Para valores muito pequenos (até 3.0ms) observou-se níveis de descarte para servidor slaves com 16 e 32 núcleos, o que pode comprometer principalmente o tempo médio de resposta. Embora tenhamos nos concentrado em um caso de uso de jogo eletrônico, os resultados de nossa análise numérica oferece *insights* sobre o uso da computação de borda para aplicativos diversos em IoT. Claro que para conclusões mais acuradas sobre outras aplicações deve-se alimentar o modelo com as mensurações respectivas da aplicação em questão.

De acordo com o que foi analisado na segunda avaliação, nota-se que as configurações que proporcionam melhor desempenho é relativo à métrica escolhida. Quando se trata de menor tempo de execução, a melhor combinação encontrada dentre os fatores e níveis analisados foi o uso do PostgreSQL utilizando uma única tabela com 1000 tuplas, obtendo o tempo total de 1,150ms, e a pior combinação foi o uso do SQL Server com 2 tabelas e 5000 tuplas, com o tempo de 3,962ms. Já para menor nível de falha na requisição, a configuração que possibilitou melhor desempenho foi o SQL Server com 500 threads simultâneas, pois proporcionou falha de 0% nas requisições, e a pior combinação foi o PostgreSQL com 2500 threads simultâneas, pois resultou em 96% de falha.

Como trabalhos futuros, pretende-se estender o modelo para incluir requisitos de disponibilidade, mensurar gasto energético e explorar alocação entre múltiplas torres MEC. Pretende-se ainda analisar outros SGBDs, bem como utilizar outros fatores e níveis para analisar mais profundamente o impacto dos mesmos na disponibilidade de um Banco de Dados.

Computação veicular em nuvem: modelagem de disponibilidade e análise de sensibilidade

Autores: Gabriel Araújo e Francisco Airton Silva

Resumo

As *Vehicle ad hoc networks* (VANETs) surgiram para tornar o tráfego mais eficiente e inteligente. As *Road Side Units* (RSUs) podem atuar como sensores e como provedoras de informações de rota para veículos. RSUs têm recursos de processamento, armazenamento e comunicação. No entanto, RSUs podem sofrer com solicitações de pico, demandas de dados não funcionais e indisponibilidade. Para superar essa deficiência, a computação em nuvem pode atuar como um recurso adicional, processando parte das solicitações, denominada *vehicular cloud computing* (VCC). Este artigo usa *Stochastic Petri Net* (SPNs) e *Reliability Block Diagram* (RBD) para avaliar a disponibilidade e confiabilidade de uma arquitetura VCC com várias RSUs. Duas análises de sensibilidade foram realizadas, que identificaram os componentes do modelo que têm o impacto mais significativo. Além de um modelo básico, modelos estendidos com maior redundância também foram propostos. O modelo base obteve disponibilidade = 97,68%, e o modelo estendido obteve disponibilidade = 99,19%. Portanto, os modelos visam ajudar os administradores de rede a planejarem arquiteturas VANET mais otimizadas, reduzindo falhas.

25.1 Introdução

As *vehicular ad hoc network* (VANET) são um tipo especial de rede desenvolvida a partir da *mobile ad hoc network* (MANET) (TALIB et al., 2018). As VANETs tratam-se de redes sem fio com o objetivo de prover comunicação entre os veículos automotores, além de oferecer maior conforto e segurança a todos os envolvidos no trânsito. Comunicações providas pelas VANETs podem ser de três tipos: V2V (*Vehicle to Vehicle*), V2I (*Vehicle to Infrastructure*) e híbrida. A comunicação V2V refere-se a uma comunicação direta entre dois veículos. A V2V também suporta mensagens orientadas a eventos em caso de emergência e executa aplicativos como jogos, *download* de arquivos, etc (CHOUDHARY et al., 2015). A V2I trata-se de uma comunicação entre um veículo e infraestruturas fixas nas margens das estradas. Estas infraestruturas são chamadas de RSU (*Road Side Unit*). Na comunicação V2I os veículos também podem obter informações da RSU como serviços baseados em localização (CHOUDHARY et al., 2015). Por fim, a comunicação híbrida combina os dois tipos anteriores.

As VANETs podem ser aplicadas a diversas áreas relacionadas ao controle de trânsito, como segurança da rede (HUSSAIN; HUSSAIN; ZEADALLY, 2019), controle de tráfego (JIAN et al., 2019), gerenciamento de vagas de estacionamento (ZHANG et al., 2016), entre outras. No entanto, as arquiteturas VANETs enfrentam diversos problemas para lidar com as grandes quantidades de dados de tráfego gerados por veículos e RSUs. Para resolver tais problemas é necessário que as VANETs tenham acesso a recursos de processamento e armazenamento robustos. A computação em nuvem pode ser uma alternativa viável para auxiliar as arquiteturas VANETs, fornecendo recursos de alta capacidade. Apesar de ser um tema promissor existem alguns desafios ao integrar VANETs e computação em nuvem. Existe a dificuldade em lidar com a alta mobilidade dos veículos e a possível longa distância aos *data centers*, diminuindo o tempo de resposta. Tais desafios fizeram surgir a linha de pesquisa denominada *vehicular cloud computing* (VCC) (GU; ZENG; GUO, 2013; WAN et al., 2014b; WAN et al., 2014a; ISMAEL; IBRAHIM; AL-KHALIL, 2020). VCC foi aplicado em contextos bem específicos como: rastreamento de rotas (HUSSAIN et al., 2013), modelo de disseminação de dados (FAROOQ; PASHA; KHAN, 2014) e controle de tráfego (JOSHI; JAIN; AGARWAL, 2015).

Dentre outras funções, o controle de tráfego impacta na segurança física das pessoas evitando acidentes. A disponibilidade das arquiteturas VCC é, portanto, muito importante para evitar falhas. As RSUs são uma das alternativas para auxiliar o controle de tráfego no contexto de VCC. As RSU podem não suportar a demanda, além de serem equipamentos de alto custo. Uma alternativa de baixo custo para avaliar arquiteturas computacionais são os modelos analíticos (SILVA et al., 2017; SILVA et al., 2015; ČEPIN, 2011). Alguns trabalhos anteriores utilizaram modelos de rede de Petri estocástica (SPN) para avaliar arquiteturas VANETs (ZHANG et al., 2016; JUNIOR et al., 2017; MALIK; PANDEY, 2018), mas nenhum com foco em arquiteturas VCC. Este trabalho propõe modelos SPN e diagramas de blocos de confiabilidade (RBD) para avaliar a dependabilidade (disponibilidade e confiabilidade) de arquiteturas VCC. Em resumo, as principais contribuições deste trabalho são:

- Uma modelagem hierárquica com RBD e SPN para avaliar a confiabilidade de uma arquitetura VCC. Três modelos de disponibilidade e um modelo de confiabilidade foram propostas. Os modelos permitem um sistema administrador para projetar um sistema VCC baseado em teorias probabilísticas. É possível prever como muita carga de trabalho específica de recursos será necessária.
- Estudos de caso baseados nos modelos propostos. O estudos de caso podem orientar outros pesquisadores no uso dos modelos funcionando como um guia e dando alguns percepções iniciais.
- Análise de sensibilidade sob o modelo SPN componentes. A análise identificou a maioria impactando parâmetros na disponibilidade do sistema. Tal um processo permite compreender profundamente o modelo e otimizá-lo.

25.1.1 Objetivos

O objetivo geral deste trabalho é desenvolver modelos analíticos para avaliar a disponibilidade e confiabilidade de uma arquitetura VCC com múltiplas RSUs. Os objetivos específicos deste trabalho são:

- Desenvolver uma modelagem hierárquica com RBD e SPN para avaliar a dependabilidade de uma arquitetura VCC;
- Realizar estudos de caso a partir dos modelos propostos;
- Fazer análises de sensibilidade sob os componentes do modelo SPN para identificar os parâmetros mais impactantes na disponibilidade do sistema; e
- Auxiliar administradores de rede a planejar arquiteturas VANETs mais otimizadas, diminuindo risco de falhas.

25.1.2 Organização

O resto deste trabalho está organizado como segue: O capítulo 25.2 apresenta alguns conceitos para a compreensão deste trabalho. O capítulo 25.3 mostra alguns trabalhos relacionados explorando diferenças específicas dispostas em uma tabela. O capítulo 25.4 apresenta a visão geral da arquitetura avaliada. Os capítulos 25.5, 25.6 e 25.7 apresentam três modelos de disponibilidade de SPN com as respectivas descrições da arquitetura avaliada. O capítulo 25.10 detalha o modelo de SPN de confiabilidade. O capítulo 25.8 detalha um estudo de caso no qual foram comparados os três modelos de disponibilidade. Por fim, o capítulo 25.11 traça algumas conclusões e trabalhos futuros.

25.2 Referencial Teórico

Esta seção descreve conceitos fundamentais para o entendimento deste trabalho. As seções incluem conceitos sobre diagramas de blocos de confiabilidade, redes de Petri estocásticas e análise de sensibilidade.

25.2.1 Diagrama de Blocos de Confiabilidade

O RBD é um modelo combinatório frequentemente utilizado para representar o funcionamento de um sistema e o funcionamento de seus componentes a fim de estimar sua confiabilidade. Os componentes são geralmente representados por blocos que podem ser arranjados em série, paralelo, ponte, k -out-of- n , ou mesmo uma combinação de abordagens anteriores (KUO; ZUO, 2003).

A Figura 251 ilustra dois exemplos de modelos RBD; o primeiro (Figura 251(a)) é um exemplo de série e o segundo é um exemplo paralelo (Figura 251(b)). Na composição da série, se um único componente falhar, todo o sistema ficará indisponível. Portanto, para um sistema composto por n componentes sua confiabilidade (ou disponibilidade) $P_p(t)$ é obtida pela Equação 25.1.

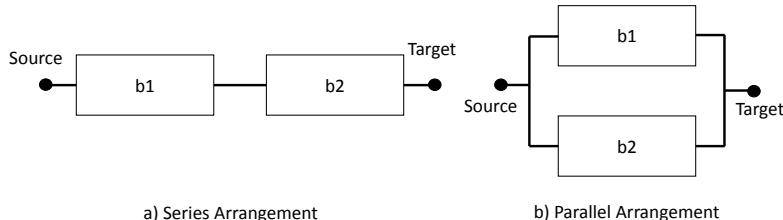


Figura 251 – Configurações de RBD: (a) série e (b) paralelo

$$P_s(t) = \prod_{i=1}^n P_i(t) \quad (25.1)$$

onde $P_i(t)$ é a confiabilidade ou a disponibilidade do bloco b_i .

Para o arranjo paralelo, se pelo menos um componente estiver funcionando, todo o sistema estará operacional. Em um sistema composto por componentes n sua confiabilidade (ou disponibilidade) $P_p(t)$ é obtida através da Equação 25.2.

$$P_p(t) = 1 - \prod_{i=1}^n (1 - P_i(t)) \quad (25.2)$$

onde $P_i(t)$ é a confiabilidade ou a disponibilidade do bloco b_i . Para outros exemplos e equações de forma fechada, consulte (KUO; ZUO, 2003).

25.2.2 Rede de Petri Estocástica

Redes de Petri (MURATA, 1989b) são uma família de formalismos muito adequada para modelar vários tipos de sistemas, uma vez que a simultaneidade, a sincronização e os mecanismos de comunicação são representados naturalmente. Este trabalho pretende usar uma extensão particular, as SPNs (MARSAN, 1995), porque permite a associação de atrasos estocásticos a transições temporizadas, podendo o respectivo espaço de estado ser convertido em CTMC (TRIVEDI, 2002). Os modelos SPN apresentam uma base matemática sólida e são adequados para representar e analisar sistemas paralelos com componentes heterogêneos e que apresentam aspectos de simultaneidade e sincronização. Portanto, esse formalismo representa uma escolha importante para modelar sistemas de computação em nuvem. Em SPNs, os locais são representados por círculos, enquanto as transições são representadas como retângulos preenchidos (transições imediatas) ou retângulos vazios (transições cronometradas).

Arcos (arestas direcionadas) conectam lugares a transições e vice-versa. Os tokens (pequenos círculos preenchidos) podem residir em locais, que denotam o estado (ou seja, marcação) de um SPN. Um arco inibidor é um tipo de arco especial que representa um pequeno círculo branco em uma borda, em vez de uma seta, e geralmente são usados para desativar as transições se houver tokens presentes em um local. O comportamento de um SPN é definido em termos de um fluxo de tokens, no sentido de que os tokens são criados e destruídos de acordo com os disparos de transição. As transições imediatas representam atividades instantâneas e têm maior prioridade de disparo do que as transições temporizadas. Além disso, tais transições podem conter uma condição de guarda, e um usuário pode especificar uma prioridade de disparo diferente, entre outras transições imediatas. Os SPNs também permitem a adoção de técnicas de simulação para obtenção de métricas do sistema como alternativa à geração de um CTMC.

25.2.2.1 Exemplo de Modelagem SPN

Um processo simples de carregamento/descarregamento de celular pode ser modelado por meio do SPN. Inicialmente, a bateria do celular está descarregada e leva algum tempo para iniciar o processo de carregamento (duração média $1/\lambda$). Em seguida, o celular é conectado a um carregador e o processo de carregamento é iniciado. A duração do processo de cobrança é considerada em média $1/r$. Após o processo de carregamento, o carregador é removido e o processo de descarga é iniciado (duração média $1/\mu$). Podemos modelar o comportamento do processo de carga/descarga usando o SPN representado na Figura 252.

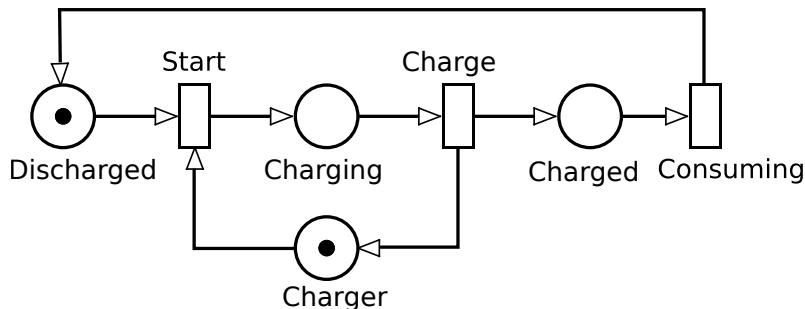


Figura 252 – SPN representando um processo de carregamento/descarregamento de um telefone celular.

- lugar *Discharged* representa o estado local do celular quando ele é descarregado;
- lugar *Charging* corresponde ao estado do celular quando está em processo de carregamento;
- lugar *Charged* é o estado que modela a situação quando o celular está carregado;
- lugar *Charger* representa o estado local do carregador quando não está em uso;
- transição *Start* modela a ação iniciando o processo de carregamento do celular. A taxa dessa transição é λ ;

- transição *Charge* representa o próprio processo de carregamento do celular. A taxa dessa transição é r ;
- transição *Consuming* representa o processo de descarga do celular. A taxa dessa transição é μ .

Nos modelos SPN de avaliação de disponibilidade, existe o conceito de componentes ativos ou inativos. A Figura 253 mostra um pequeno modelo de disponibilidade com dois componentes (A e B). Ambos têm parâmetros de tempo médio para falha (MTTF) e reparo (MTTR). O componente A, por exemplo, está ativo quando tem um token no lugar A_U (para cima) e inativo quando tem um token no lugar A_D (para baixo). Neste exemplo, para o componente B estar ativo, A também deve estar ativo. O arco de inibição garante que, se o componente A mudar de estado superior para estado inferior, a transição T0 será disparada e o componente B também estará em estado inferior.

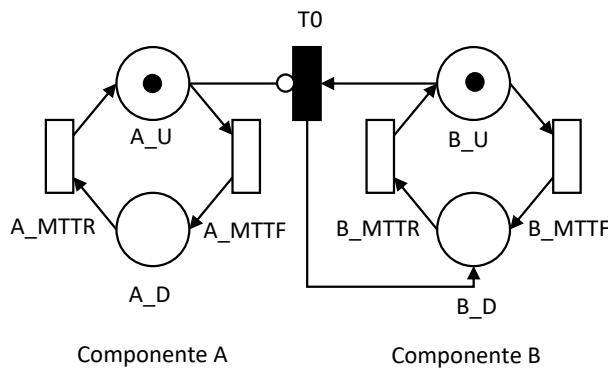


Figura 253 – Exemplo de dependência entre componentes que representam estados de disponibilidade.

25.2.3 Análise de Sensibilidade

Uma análise de sensibilidade é uma medida do efeito local de um dado de entrada em relação aos dados de saída. O objetivo desta análise é destacar os componentes de maior e menor impacto em uma métrica avaliada. Posteriormente, técnicas podem ser adotadas para melhorar alguns aspectos do sistema, como disponibilidade ou desempenho (CAMPOLONGO; TARANTOLA; SALTELLI, 1999). Uma análise de sensibilidade pode ajudar os administradores de sistema a otimizar arquiteturas de sistema com despesas mínimas.

25.2.3.1 Diferença Percentual

Existem muitas maneiras de realizar a análise de sensibilidade, como análise de regressão, análise de perturbação, variação uma a uma, simulação de Monte Carlo, análise diferencial paramétrica, análise de correlação e diferença percentual. A distinção de qual método utilizar é uma etapa difícil, sendo necessário estabelecer quais recursos computacionais estão disponíveis, e as características dos problemas abordados (CAMPOLONGO et al., 2004) (PIANOSI et al., 2016). O método da diferença percentual visa ser um dos escolhidos para realizar a análise de sensibilidade deste trabalho, que é uma abordagem que não utiliza um domínio contínuo para os valores de entrada dos parâmetros. Este processo é baseado no cálculo da diferença percentual, variando um parâmetro de entrada com um valor mínimo e máximo (HOFFMAN; GARDNER, 1983).

A Equação 25.3 mostra como o índice de sensibilidade é calculado pelo método da diferença percentual. A expressão $\max \{Y(\theta)\}$ e $\min \{Y(\theta)\}$ são, respectivamente, os valores de saída máximo e mínimo, medidos pela variação do parâmetro θ sob um intervalo de n valores possíveis (JÚNIOR, 2016).

$$S_\theta\{Y\} = \frac{\max \{Y(\theta)\} - \min \{Y(\theta)\}}{\max \{Y(\theta)\}} \quad (25.3)$$

Tabela 169 – Comparação de trabalhos a partir dos modelos, métricas, múltiplas RSUs e análise de sensibilidade.

Trabalhos	Tipo de Modelo	Métricas	Múltiplas RSUs	Análise de Sensibilidade
(MALIK; PANDEY, 2018)	Petri Net	Desempenho	Não	Não
(ZHANG et al., 2016)	Petri Net	Desempenho	Não	Não
(JUNIOR et al., 2017)	Petri Net	Desempenho	Não	Não
(SHAH; ILHAN; TURELLI, 2019)	Markov Chain	Desempenho	Não	Não
(LI et al., 2017a)	Markov Chain	Desempenho	Não	Não
(YAO et al., 2017)	Markov Chain	Desempenho	Sim	Não
(BRAHMI; DJAHEL; GHAMRI-DOUDANE, 2012)	Markov Chain	Desempenho	Não	Não
(BITAM; MELLOUK, 2013)	Markov Chain	Desempenho	Não	Não
(ZHENG; WU, 2015)	Markov Chain	Desempenho	Não	Não
Este Trabalho	Petri Net e RBD	Disponibilidade, confiabilidade e tempo de inatividade	Sim	Sim

25.2.3.2 Projeto de Experimentos

O Projeto de Experimentos, ou DoE (*Design of Experiments*), corresponde a uma coleção de técnicas estatísticas que permite aprofundar o conhecimento a respeito do produto ou processo em estudo (KLEIJNEN, 1995). Pode ser definido também por uma série de testes em que o pesquisador altera o conjunto de variáveis ou fatores de entrada com o intuito de observar e identificar os motivos para alterações na resposta de saída.

Os parâmetros a serem alterados são definidos utilizando um plano de experimento. O objetivo é gerar a maior quantidade de informação com menos experimentos possíveis. O comportamento do sistema com base nas mudanças de parâmetros pode ser observado usando conjuntos de saídas. Este trabalho também visa aplicar a metodologia DoE para realizar a análise de sensibilidade. Na literatura existem três categorias de gráficos adotados usualmente para experimentos com DoE:

- Gráfico de Pareto é representado por barras em ordem decrescente. Quanto maior a barra maior o impacto. Cada barra representará a influência de cada fator sobre a variável dependente.
- Gráficos de efeitos principais são utilizados para examinar as diferenças entre as médias de nível para um ou mais fatores, representa graficamente a resposta média para cada nível de fator ligado por uma linha. Pode ser aplicado na utilização de comparação entre a força relativa dos efeitos de vários fatores. O sinal e magnitude de efeito principal podem expressar o valor médio de resposta. A magnitude expressará a força do efeito, quanto maior a inclinação da linha, maior será a magnitude do efeito principal. É necessário considerar que a linha horizontal significa que não tem nenhum efeito principal, ou seja, cada nível afetará a resposta da mesma maneira.
- Gráficos de interação são responsáveis por identificar interações entre fatores. Uma interação ocorre quando a influência de um dado fator no resultado é alterada (amplificada ou reduzida) pela diferença de nível de outro fator. Se as linhas no gráfico são paralelas, significa que haverá interação entre os fatores. Caso, não sejam paralelas, significa que existe interação entre os fatores. Quanto maior o grau de afastamento em paralelo, mais forte o efeito de tal interação. Em uma interação sinérgica, as linhas no gráfico não se cruzaram. Nas interações antagônicas as linhas no gráfico irão se cruzar.

25.3 Trabalhos Relacionados

Esta seção apresenta alguns trabalhos relacionados. A Tabela 169 resume um comparativo dos trabalhos relacionados sob quatro aspectos: Tipo de Modelo, Métricas, Múltiplas RSUs e Análise de Sensibilidade.

O primeiro critério de comparação focou em **Tipo de Modelo** utilizado pelos trabalhos. A maioria dos trabalhos utilizou as cadeias de Markov como modelo analítico para avaliar suas propostas. (BENATIA et al., 2013) propõe uma cadeia de Markov para permitir simulações mais precisas de cenários em VANETs. (BRAHMI; DJAHEL; GHAMRI-DOUDANE, 2012) propõe uma cadeia de Markov tornar mensagens V2V mais seguras. (JUNIOR et al., 2017; MALIK; PANDEY, 2018; ZHANG et al., 2016) utilizaram as redes de Petri para avaliação de propostas como infraestruturas VANETs, autenticação de mensagens V2V e V2I, e estacionamento mais eficiente. No entanto, nenhum dos artigos utilizou modelos RBDs hierárquicos em conjunto com seus modelos analíticos.

Sobre as **Métricas** utilizadas pelos artigos, observa-se que todos optaram pela avaliação de desempenho. Alguns artigos apresentam propostas inovadoras e pretendem por meio da avaliação de desempenho, fazer a validação de seus modelos. Outros artigos como (SHAH; ILHAN; TURELI, 2019; ZHENG; WU, 2015; LI et al., 2017a) se propuseram a fazer uma avaliação de desempenho no controle de acesso ao meio (MAC) no padrão IEEE 802.11p (IEEE, 2010) no contexto das VANETs. Contudo, nenhum dos trabalhos explorou as métricas de disponibilidade, confiabilidade ou tempo de inatividade.

Outro aspecto refere-se às **Múltiplas RSUs** que dizem respeito a utilização da várias RSUs em conjunto, para o funcionamento correto da proposta. Em (YAO et al., 2017) é apresentado uma estratégia de roteamento preditivo que explora a regularidade dos comportamentos de movimentação de veículos para aumentar o desempenho da transmissão. Este é o único trabalho que faz uso de múltiplas RSUs. Entretanto, múltiplas RSUs não são o foco, são usadas apenas para transportar nós em casos específicos, onde as RSUs estejam implantadas e a comunicação V2V não seja possível.

Por fim, o critério **Análise de Sensibilidade** trata-se de uma análise realizada nos componentes do sistema, variando os MTTF e MTTR de cada um para descobrir qual componente tem maior impacto na disponibilidade do sistema. Com o resultado desta análise, é possível replicar os componentes mais sensíveis utilizando modelos redundantes e assim melhorar a disponibilidade e consequentemente o tempo de inatividade do sistema. Nenhum dos trabalhos relacionados realizou uma análise de sensibilidade. Este trabalho explora as métricas de disponibilidade, tempo de inatividade e confiabilidade a partir de um modelo de SPN hierárquicos com modelos RBDs. Este trabalho também utiliza múltiplas RSUs que trabalham cooperativamente auxiliadas pela computação em nuvem, que é utilizada para compensar nas limitações de armazenamento e processamento. Além disso, também foi realizado duas análises de sensibilidade para gerar os modelos SPN redundantes para melhorar a disponibilidade do sistema.

25.4 Visão Geral da Arquitetura Base Avaliada

A Figura 254 apresenta uma visão geral do cenário avaliado, que inclui um grupo de RSUs que se comunicam com os veículos e a nuvem. Presume-se que cada veículo tenha um dispositivo OBU (Unidade de Bordo) com capacidade de rastreamento GPS. RSUs têm capacidade limitada de armazenamento e processamento, portanto, uma nuvem privada é usada para fazer o processamento de informações adicionais. A princípio, todas as RSUs estão conectadas à nuvem por meio de uma conexão sem fio (5G, por exemplo). No entanto, RSUs podem ser conectadas via, por exemplo, IEEE 802.11p, Lora ou 5G C-V2X. A escolha de qualquer uma dessas redes tem um impacto direto na disponibilidade do sistema. A arquitetura ilustra um grupo de quatro RSUs, mas o cenário é totalmente escalonável para qualquer número de RSUs. Cada RSU possui uma área de convergência, que se comunica com os veículos que a alcançam. As RSUs serão posicionadas ao longo da estrada. Os veículos encontrarão RSUs para apoiar enquanto viajam pela estrada.

Quando um veículo entra na área de convergência de uma RSU, ele solicita informações de tráfego sobre a região. O tráfego é maior nos horários de pico, então a taxa de chegada de novas solicitações geralmente é alinhada com a hora do dia ou o número e a velocidade dos veículos na estrada (SHAAMILI; RANJITH; SUPRIYA, 2018). A solicitação de veículo para um RSU conterá dados do próprio veículo, como rota previamente traçada, velocidade, condições do veículo, etc. A RSU irá então encaminhar a solicitação do veículo para a nuvem, que está recebendo informações das regiões das outras RSUs. Por fim, a nuvem agrupará e processará os dados, retornando uma resposta ao RSU solicitante, que então encaminhará ao veículo solicitante. Após o veículo receber a resposta da RSU, a OBU calculará a velocidade recomendada,

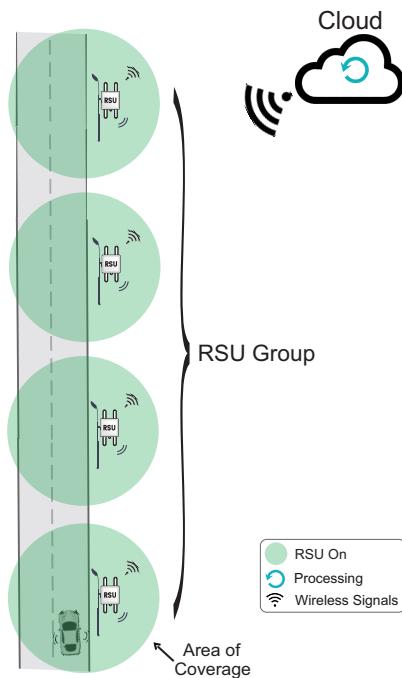


Figura 254 – Visão geral do cenário avaliado.

por exemplo. Além disso, a RSU pode alertar o motorista sobre rotas mais eficientes e possíveis acidentes. Em seguida, o motorista pode optar por manter ou alterar a velocidade atual do veículo, cruzando assim os cruzamentos com o menor tempo de espera possível.

A figura 255 apresenta a arquitetura da nuvem dividida em camadas. Esta organização em camadas visa destacar que os componentes obedecem a uma relação de dependência, onde os componentes superiores só estarão disponíveis se os inferiores estiverem funcionando. A nuvem é composta pelos componentes *frontend* e *nó*. O *front-end* é responsável pelo gerenciamento de dados. É dividido nos seguintes componentes: *hardware* (HW), *operating system* (OS), *platform manager* (PM), *cluster manager* (CM), *block-based storage* (BBS), e *filebased storage* (FBS). O *nó* fornece máquinas virtuais e serviços hospedados aos usuários, sendo composto por *hardware* (HW), *operating system* (OS), *hypervisor* (HP), *instance manager* (IM), e *virtual machine* (VM). A tabela 170 descreve alguns dos componentes baseados em arquiteturas de nuvem também usados por (MELO et al., 2017b).

Tabela 170 – Descrição dos componentes da nuvem.

Componente	Descrição
Platform Manager	Define e monitora qual cluster fornecerá novas instâncias
Cluster Manager	Define qual máquina física pertence a um cluster
Block-Based Storage	Fornece armazenamento em bloco para máquinas virtuais
File-Based Storage	Armazenamento que permite aos usuários trocarem informações sobre máquinas virtuais
Instance Manager	Mantém o ciclo de vida das máquinas virtuais

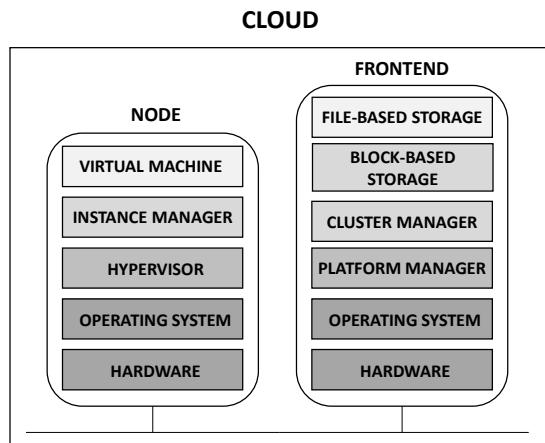


Figura 255 – Arquitetura em camadas da nuvem dividida entre os componentes *frontend* e *node*.

25.5 Modelo Um - Base

A Figura 256 apresenta os modelos SPN e RBD para o cenário detalhado no capítulo anterior. A Figura 256(a) mostra o modelo básico do SPN. O modelo básico consiste nos componentes: RSUN GROUP, NETWORK e CLOUD. Cada um desses componentes possui um MTTF (tempo médio de falha) e um MTTR (tempo médio de reparo). O número de RSUs ativas é dado pelo número de *tokens* em RSUN_U (RSUs ativos), então a marcação N indica quantos RSUs estão em operação. Da mesma forma, o número de *tokens* em RSUN_D indica quantos RSUs estão inativos. A mudança entre os estados ativo e inativo em cada RSU é causada pelo disparo das transições: RSUN_MTTF e RSUN_MTTR. Cada RSU faz parte de um grupo de compartilhamento de informações. Portanto, para que o RSUN GROUP esteja disponível, o número de *tokens* em RSUN_U deve ser igual ao valor de marcação N. O avaliador pode expandir a arquitetura proposta para qualquer número de RSUs, apenas alterando o número de *tokens* usando a marcação N.

A NETWORK estará funcionando quando tiver um token na localização NETWORK_U (rede ativa). A rede não estará funcionando quando houver um token no local NETWORK_D (rede inativa). As mudanças ativas e inativas da rede são acionadas pelas transições: NETWORK_MTTF e NETWORK_MTTR. Semelhante à rede, a CLOUD está funcionando quando o local CLOUD_U tem um token. A nuvem não está funcionando quando o local CLOUD_D tem um token. As mudanças ativas e inativas na nuvem são acionadas pelas transições: CLOUD_MTTF e CLOUD_MTTR.

O modelo base é um modelo hierárquico formado por SPN e RBD. Os modelos de RBDs foram escolhidos para representar a disponibilidade da nuvem porque não há alta dependência entre os componentes e porque técnicas como reparo automatizado ou migração de máquinas virtuais não são consideradas neste trabalho. Consideramos que os valores de MTTF e MTTR seguem uma distribuição exponencial. Os parâmetros do componente nuvem do modelo SPN são obtidos a partir dos resultados gerados pelo modelo RBD da nuvem.

A figura 256(b) mostra o modelo RBD da CLOUD. O modelo RBD da nuvem é composto por FRONTEND e NODE. Esses dois componentes também possuem modelos de RBDs que geram parâmetros para o modelo de RBD da nuvem, compreendendo uma hierarquia ainda maior. A Figura 256(d) mostra o modelo RBD do *frontend*, que consiste em *hardware* (HW), *operating system* (OS), *platform manager* (PM), *cluster manager* (CM), *block-based storage* (BBS) e *file-based storage* (FBS). A Figura 256(c) mostra o modelo RBD do nó, formado por *hardware* (HW), *operating system* (OS), *hypervisor* (HP), *instance manager* (IM) e *virtual machine* (VM). Portanto, a proposta básica é uma combinação de um modelo SPN e três modelos RBD.

Para este modelo, duas métricas de interesse foram exploradas: disponibilidade e tempo de inatividade. A equação de disponibilidade representa a soma das probabilidades do número de tokens em cada estado. Dessa forma, P representa probabilidade e # representa o número de tokens em um determinado local. O tempo de inatividade (D) pode ser obtido por $D = (1 - A) \times 8760$, onde A representa a disponibilidade do

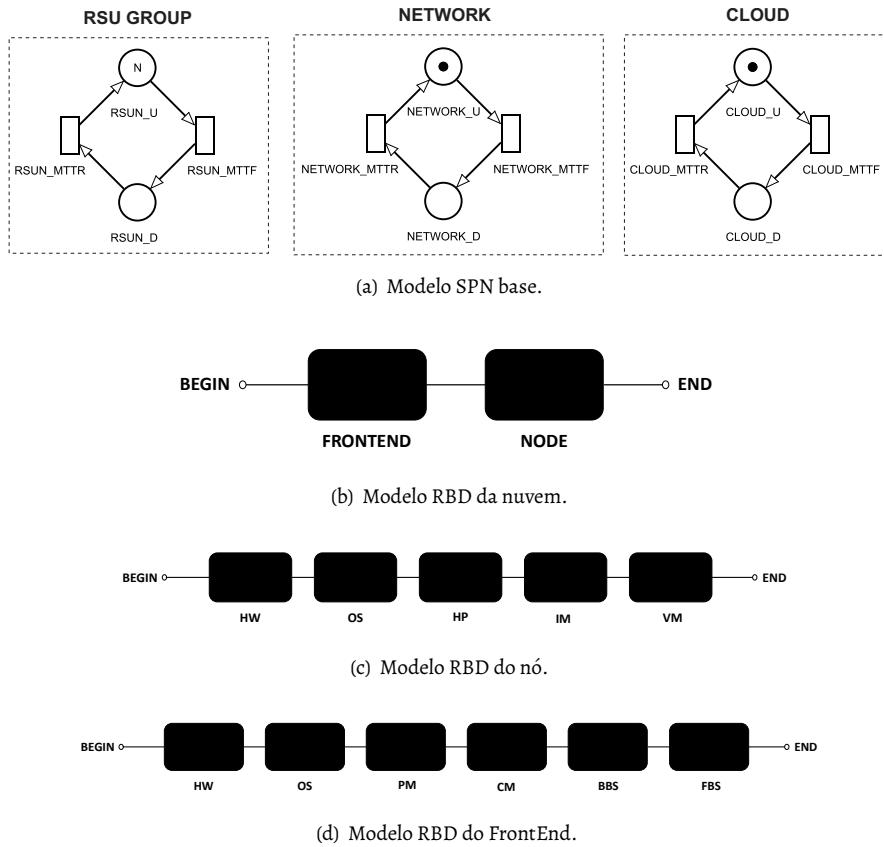


Figura 256 – Modelo SPN base e modelos RBD da nuvem.

Tabela 171 – Índices de sensibilidade do modelo base.

Modelo Base		
Variável	Índice	Servidor
RSUN_MTTF	$2,13 \times 10^{-2}$	Infinite
RSUN_MTTR	$1,59 \times 10^{-2}$	Single
CLOUD_MTTF	$9,53 \times 10^{-3}$	Infinite
CLOUD_MTTR	$7,18 \times 10^{-3}$	Single
NETWORK_MTTF	$1,91 \times 10^{-3}$	Infinite
NETWORK_MTTR	$1,44 \times 10^{-4}$	Single

sistema e 8760 o número de horas no ano. Para a métrica de disponibilidade, o sistema está totalmente funcional quando o grupo de RSUs, a rede e a nuvem estão ativos ao mesmo tempo. A equação 25.4 representa a condição descrita acima.

$$\mathbf{A} = P\{(\#RSUN_U = N) \text{AND} (\#CLOUD_U > 0) \text{AND} (\#NETWORK_U > 0)\} \quad (25.4)$$

25.5.1 Análise de Sensibilidade Usando Diferença Percentual

A análise de sensibilidade foi realizada a partir dos tempos de transição de todos os componentes do modelo base. O MTTF e o MTTR foram variados em cinco valores. O intervalo está entre 50% abaixo (valor mínimo) e 50% acima (valor máximo) do valor base. A tabela 171 apresenta os índices de sensibilidade das transições do modelo base em ordem decrescente, ou seja, do maior para o menor impacto na disponibilidade do sistema.

O resultado dessa análise indicou o RSU GROUP como o componente mais sensível, por isso foi esco-

Tabela 172 – Fatores e níveis para calcular a métrica de disponibilidade.

Fatores	Nível 1	Nível 2
RSUN_MTTF	250	750
RSUN_MTTR	1	3
NETWORK_MTTF	4161	12483
NETWORK_MTTR	6	18
CLOUD_MTTF	62.946	188.839
CLOUD_MTTR	0.4569	1.37069

Tabela 173 – Combinação de fatores e níveis de DoE para a métrica de disponibilidade.

Fatores (h)	Combinações							
	#1	#2	#3	#4	#5	#6	#7	#8
RSUN_MTTF	250	750	250	750	250	750	250	750
RSUN_MTTF	1	1	3	3	1	1	3	3
NETWORK_MTTF	4161	4161	4161	4161	12483	12483	12483	12483
NETWORK_MTTR	18	6	6	18	18	6	6	18
CLOUD_MTTF	188.839	62.946	188.839	62.946	62.946	188.839	62.946	188.839
CLOUD_MTTR	1.37069	1.37069	0.4569	0.4569	0.4569	0.4569	1.37069	1.37069

lhido como um componente a ser replicado nos modelos estendidos com *Cold Standby* e *Warm Standby*. Para enriquecer os resultados da análise de sensibilidade, a seção seguinte apresenta outra análise baseada na metodologia DoE (ROCHA et al., 2019), destacando além do impacto dos fatores e a interação entre eles.

25.5.2 Análise de Sensibilidade Usando DoE

Uma análise de sensibilidade com base no DoE foi realizada nos componentes do sistema, variando o MTTF e o MTTR de todos os componentes em 50% a mais e 50% a menos dos parâmetros de base. Os parâmetros básicos do grupo RSU e da rede foram extraídos de (AUDU et al., 2017) e (ANDRADE; NOGUEIRA, 2018), respectivamente. Os parâmetros da nuvem foram obtidos hierarquicamente a partir da execução do modelo RBD da nuvem. Os valores de *frontend* e nó, usados para executar o modelo RBD da nuvem, foram extraídos de (MELO et al., 2017b). A tabela 172 descreve os fatores e níveis DoE usados para medir a disponibilidade. Os valores mínimo (nível 1) e máximo (nível 2) para cada transição cronometrada são aplicados para gerar os gráficos.

A tabela 173 exibe os fatores relacionados às transições cronometradas do grupo RSU, rede e componentes da nuvem. A tabela foi gerada com oito combinações (fração 1/8). As combinações de cada fator variaram em dois níveis, calculando a métrica de disponibilidade.

A figura 257 apresenta o gráfico de Pareto, que fornece uma análise do fator de impacto mais significativo. RSUN_MTTR e RSUN_MTTF são os fatores de maior impacto, seguidos por NETWORK_MTTR. O componente RSU é o fator de impacto mais significativo, como na análise anterior, considerando MTTF e MTTR. Percebemos que o componente NETWORK, em seu tempo médio de reparo, tem um impacto substancial.

A Figura 258 mostra o gráfico de efeitos principais entre os parâmetros MTTF e MTTR do grupo de RSUs, rede e componentes da nuvem. O eixo y representa a média das disponibilidades obtidas para cada nível. Os fatores RSUN_MTTR e RSUN_MTTF tiveram o impacto mais significativo, com linhas mais inclinadas. O fator RSUN_MTTF obteve uma disponibilidade de 95,72% em seu primeiro nível (250 h). No segundo nível (750 h), a disponibilidade era de 97,80%. O RSUN_MTTR tem uma disponibilidade semelhante, mas inversamente, 97,80% no nível 1 (1 h) e 95,72% no nível 2 (3 h). A variação dos fatores RSUN_MTTR e RSUN_MTTF resultou em uma mudança considerável na disponibilidade do sistema. No fator NETWORK_MTTF, a variação foi de apenas 96,66% no nível 1 (4.161 h) e 96,86% no nível 2 (12483 h), variação mínima para disponibilidade. Os outros fatores variaram um pouco mais que NETWORK_MTTF.

A figura 259 apresenta a interação dos fatores. As médias de disponibilidade variaram de 0,94 a 0,99.

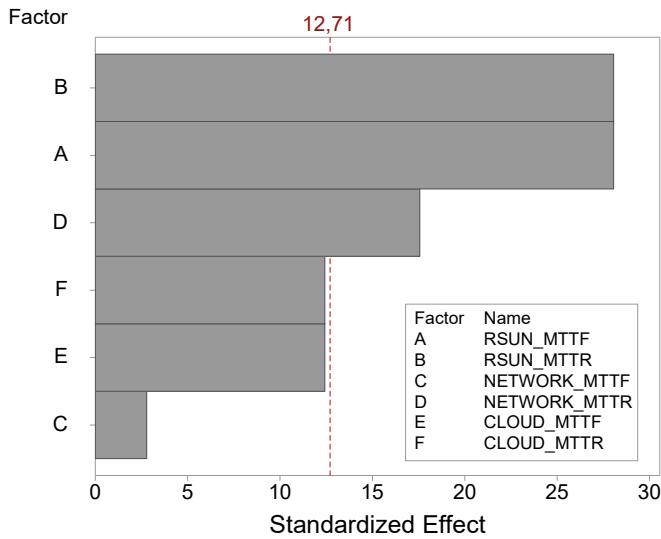


Figura 257 – Impacto dos fatores do sistema na disponibilidade.

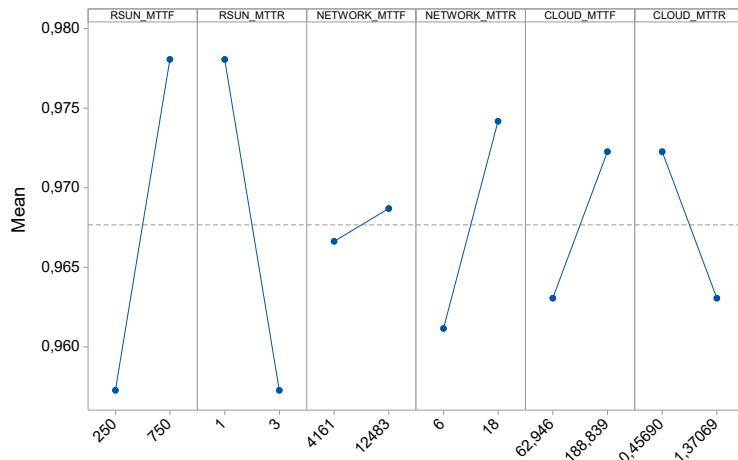


Figura 258 – Efeitos principais dos fatores do sistema.

Linhos paralelos significam que não há interação. Quanto menos paralelos forem as linhas, maior será a força da interação. As interações mais significativas incluíram os fatores de Rede. Por exemplo, NETWORK_MTTF e CLOUD tiveram altas interações.

25.6 Modelo Dois - Estendido com *Cold Standby*

A Figura 260 apresenta uma visão geral do cenário avaliado para a proposta estendida *Cold Standby* (BRIŠ, 2013). Este cenário é uma extensão do cenário de base anterior, com redundância para melhorar a disponibilidade. Nesta proposta, dois grupos de RSUs: RSU Group 1 e RSU Group 2. RSU Group 1 é o primeiro grupo e funciona como cenário base, estando posicionado em um lado da estrada. RSU Group 2 é o segundo grupo e tem o mesmo funcionamento que RSU Group 1, estando posicionado no lado oposto da estrada. A estrada é dividida em seções de estradas. Portanto, duas RSUs são colocadas em cada seção da estrada, uma de cada lado. Cada trecho de estrada deve ter pelo menos uma RSU ativa para que a estrada seja considerada totalmente coberta, ou seja, esteja disponível.

Nesta proposta *Cold Standby*, consideramos que inicialmente, apenas RSU Group 1 estará funcio-

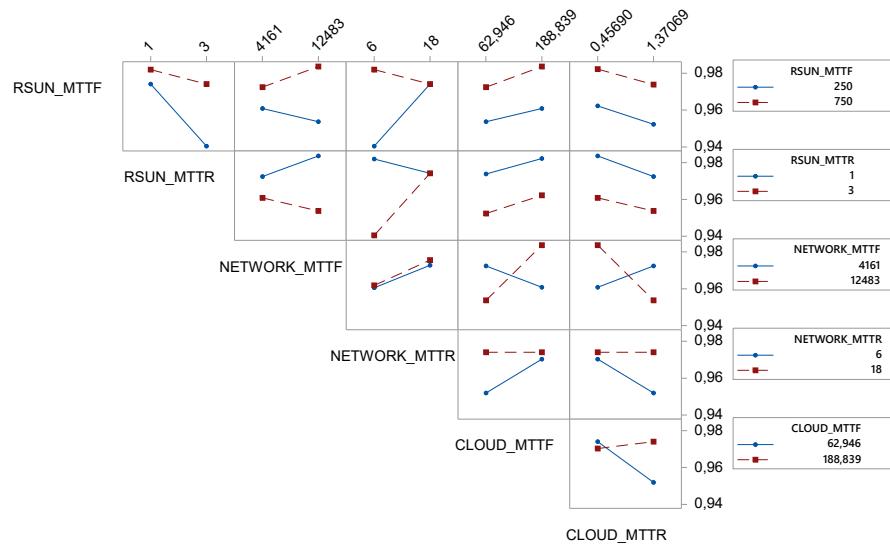


Figura 259 – Interação entre fatores de disponibilidade.

nando, enquanto o segundo grupo será desligado. No entanto, quando uma das RSUs do grupo 1 falhar, a RSU do lado oposto (grupo 2) será acionada, para que o trecho da estrada continue a receber suporte e o sistema permaneça disponível. Quando a RSU do grupo 1 for reparada, a RSU oposta será desconectada. Se as duas RSUs que cobrem a mesma seção da estrada falharem, esta seção da estrada tornará o sistema indisponível.

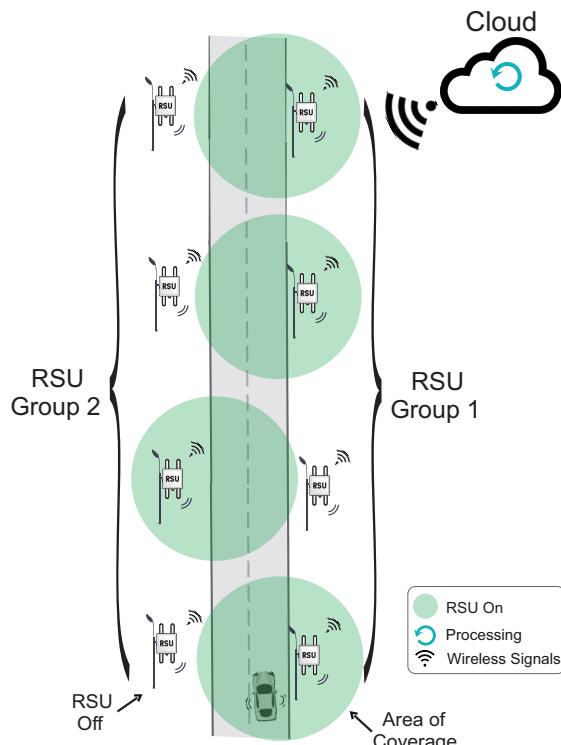


Figura 260 – Visão geral do cenário avaliado. Arquitetura estendida com redundância, com RSUs secundárias em modo desligado do outro lado da estrada.

A Figura 261 estende o modelo de arquitetura do cenário base com redundância *Cold Standby* (BRIŠ, 2013). Esta proposta visa melhorar a disponibilidade do sistema em relação ao modelo básico. Para esta arquitetura, a redundância foi aplicada apenas ao componente RSU GROUP, que se mostrou o componente mais relevante por meio de análises de sensibilidade.

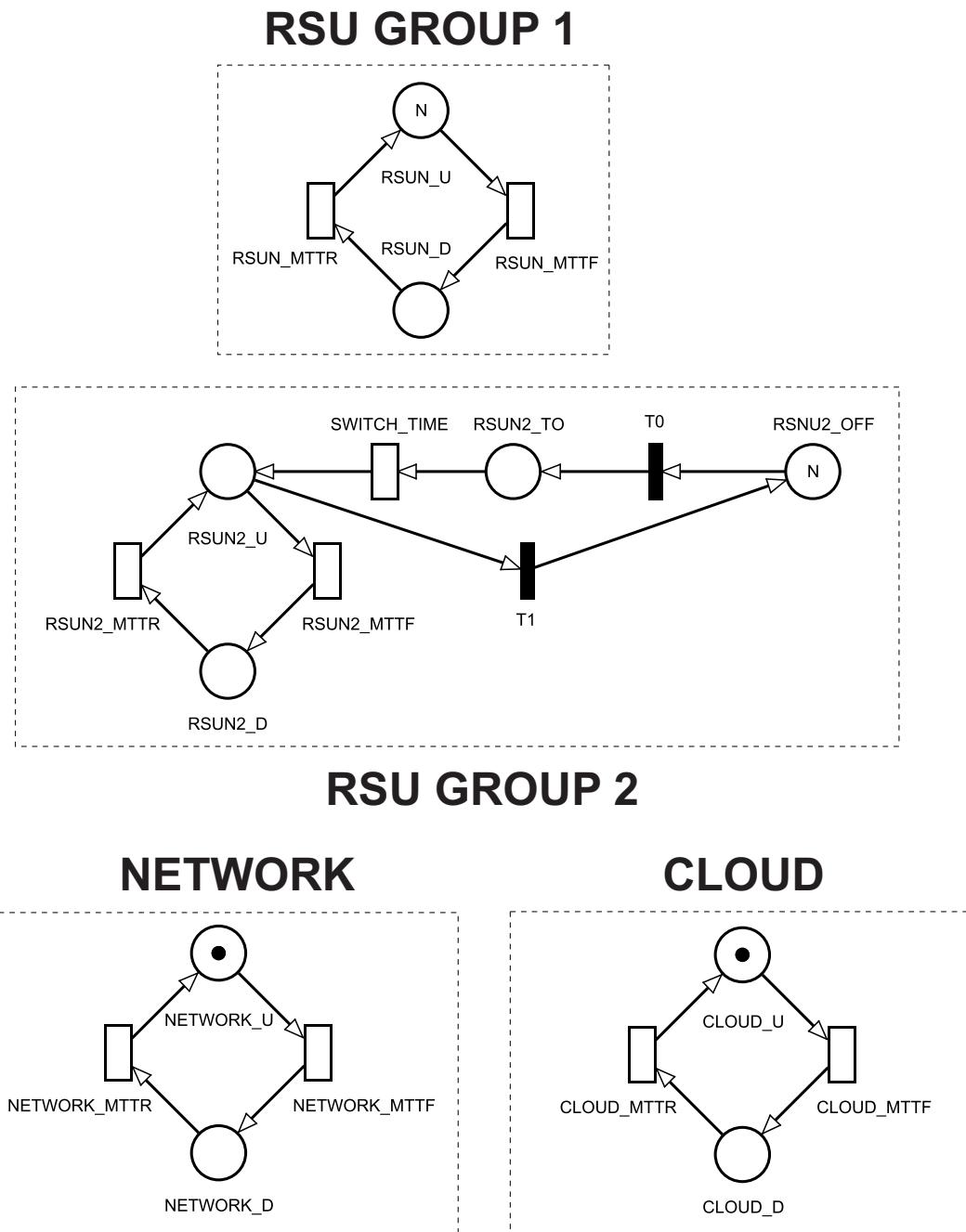


Figura 261 – Modelo SPN estendido com Cold Standby.

Neste modelo, um novo componente chamado RSU GROUP 2 foi adicionado, com a mesma estrutura, operação e parâmetros de entrada como RSU GROUP 1. Uma nova transição cronometrada chamada SWITCH_TIME foi adicionada, além de duas novas transições imediatas, T0 e T1. Dois novos locais também

foram adicionados, RSUN2_TO (RSU Turning On) e RSUN2_OFF. Os componentes do RSU GROUP 2 estão “frios”, ou seja, desabilitados, enquanto os componentes do RSU GROUP 1 estarão ativos. A transição T0 imediata é disparada quando uma RSU do RSU GROUP 1 falha, ou seja, um *token* em RSUN_U é removido. O disparo T0 envia um *token* armazenado em RSUN2_OFF para RSUN2_T0, colocando o *token* redundante em um estado “ligado”. Quando um token chega em RSUN2_T0, a transição cronometrada SWITCH_TIME pode ser disparada ativando RSU GROUP 2. Portanto, o tempo desta transição também corresponde ao tempo que o RSU GROUP 2 levará para inicializar.

O comportamento de T0 colocar um *token* em um estado de “ligado” faz a transição Switch_Time funcionar como um *K-Server*. Assim, em vez dos tokens redundantes em RSUN2_OFF competindo para serem disparados, apenas o número de tokens em RSUN_D é habilitado. Este comportamento reflete a operação dos trechos da estrada no modelo redundante. Mesmo se eles tiverem outros RSUs prontos para serem ativados (tokens em RSUN2_OFF), apenas o RSU redundante daquela seção específica pode ser acionado (o RSU no lado oposto da estrada).

Vejamos um exemplo para um melhor entendimento. Considere que existe um *token* em RSUN_D. T0 e Switch_Time habilitarão o *token* redundante para substituí-lo, mas se o *token* redundante também falhar, o sistema ficará indisponível. T0 não pode permitir que mais *tokens* redundantes sejam vinculados, já que os *tokens* restantes pertencem a outras seções da estrada.

Quando a transição RSUN_MTTR é disparada, um *token* em RSU GROUP 1 será reparado e a transição imediata T1 enviará um *token* em RSUN2_U de volta para RSUN2_OFF, indicando que uma RSU redundante foi desconectada. No entanto, a transição T0 só é disparada se a soma dos tokens em RSUN2_U, RSUN2_D e RSUN2_T0 for maior que o número de *tokens* em RSUN_D. Dessa forma, a quantidade de *tokens* é controlada. Para que o sistema esteja disponível, a soma dos *tokens* em RSUN_U e RSUN2_U deve ser igual ao número de *tokens* definidos na marca N. Se a soma dos *tokens* em RSUN_U e RSUN2_U for menor que o valor em N, o sistema ficará indisponível.

Antes de apresentar a métrica de disponibilidade, a Figura 262 apresenta um exemplo de como funciona o esquema de redundância para calcular a disponibilidade do cenário avaliado. Conforme mencionado anteriormente, a estrada é dividida em seções. Se as duas RSUs na seção estiverem em modo de falha, o sistema ficará indisponível. Considerando a figura (com quatro seções), a soma da quantidade de RSUs ativas em ambos os lados deve ser igual a quatro para que o sistema esteja disponível. A conexão de rede e a nuvem também devem estar disponíveis. Nesse sentido, a Equação 25.5 define a disponibilidade do modelo estendido com *Cold Standby*.

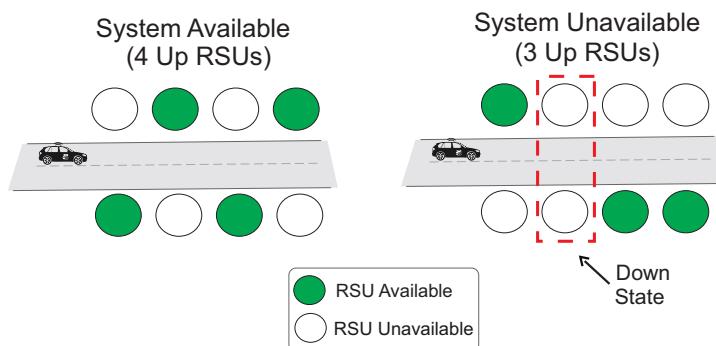


Figura 262 – Esquema de redundância para o cenário avaliado.

$$\begin{aligned} \mathbf{A} = P\{ & ((\#RSUN_U + \#RSUN2_U) = N) \\ & \text{AND}(\#\text{NETWORK_U} > 0) \text{AND}(\#\text{CLOUD_U} > 0) \} \end{aligned} \quad (25.5)$$

Vale ressaltar que as transições MTTF (tanto do modelo base quanto estendido) são da semântica do servidor infinito e as transições do MTTR são da semântica do servidor único. A tabela 174 mostra as

Tabela 174 – Condições de guarda para o modelo SPN estendido com *Cold Standby*.

Transição	Expressão	Descrição
T0	((#RSUN2..D+RSUN2..U+RSUN2..TO)<#RSUN..D)	Quando a soma dos tokens em RSUN2 é menor que o número de tokens desativados em RSUN.
T1	((#RSUN2..D+RSUN2..U+RSUN2..TO)>#RSUN..D)	Quando a soma dos tokens em RSUN2 é maior que o número de tokens desativados em RSUN.

condições de guarda usadas para a operação do modelo estendido com *Cold Standby*. As condições de guarda são usadas em duas transições.

25.7 Modelo Três - Estendido com *Warm Standby*

A Figura 263 apresenta uma visão geral do cenário avaliado para a proposta estendida com *Warm Standby* (SOUSA et al., 2014; YUAN; MENG, 2011). Como no cenário anterior, este cenário é uma extensão do cenário básico usando redundância. Esta proposta também tem dois grupos de RSUs: RSU Group 1 e RSU Group 2. Ambos os grupos estão posicionados e operados de forma semelhante ao cenário anterior. No entanto, RSU Group 2 não será completamente desconectado.

Inicialmente, RSU Group 1 estará ativo e RSU Group 2 estará parcialmente ativo. Ou seja, ficará ocioso. Como o RSU Group 2 não estará totalmente ativo, seu cálculo não é considerado. Quando uma RSU no grupo 1 falha, a RSU no lado oposto (grupo 2) será totalmente ativa para cobrir a respectiva seção. No entanto, o tempo para inicializar o suporte quente por um dispositivo sobressalente é muito menor do que o suporte frio do dispositivo. Além disso, RSUs parcialmente conectadas demoram muito mais para falhar do que RSUs totalmente ativas. Quando o RSU Group 1 for reparado, ele será ligado novamente. A RSU oposta será parcialmente desligada. Se as duas RSUs que cobrem a mesma seção falharem, o sistema ficará indisponível. Portanto, o sistema estará disponível quando o número total de RSUs que compõem um grupo estiver ativo.

A Figura 264 mostra o modelo estendido com *Warm Standby* (SOUSA et al., 2014; YUAN; MENG, 2011). Este modelo busca melhorar ainda mais a disponibilidade do sistema em relação ao modelo base. O RSU GROUP é o componente mais sensível, como mencionado anteriormente, e portanto redundância também foi aplicada somente a ele. Neste novo modelo, o componente RSU GROUP 2 foi adicionado como um componente redundante. Porém, RSU GROUP 2 possui dois componentes que representam dois estados, um operacional (OP) e um inativo (OC).

Consideramos o MTTF de RSU GROUP 2 (RSUN2_MTTR_OC) sendo 50% maior que o operacional baseado no trabalho de (GUIMARÃES; MACIEL; JR, 2013; SOUSA et al., 2014). Como no modelo anterior, uma nova transição cronometrada chamada SWITCH_TIME também foi adicionada, além de duas transições imediatas chamadas T0 e T1. Um novo local chamado RSUN2_TO_OP também foi adicionado. RSU GROUP 2 estará rodando ao mesmo tempo que RSU GRUPO 1. No entanto, inicialmente, apenas o componente ocioso estará ativo. Desta forma, nenhum serviço estará rodando em RSU GROUP 2, então o cálculo não é considerado.

A transição T0 é disparada quando um RSU GROUP 1 RSU falha, ou seja, um token em RSUN_U é removido. O disparo T0 envia um token de RSUN2_U_OC para RSUN2_T0_OP. Após o token redundante ser enviado para RSUN2_T0_OP, o SWITCH_TIME pode ser disparado, ativando o RSU GROUP 2 operacional. Portanto, o tempo desta transição também corresponde ao tempo que o RSU GROUP 2 levará para inicializar. Como no modelo anterior, a transição Switch_Time funciona como um K-Server para refletir a operação das seções no modelo. Desta forma, apenas o número de tokens em RSUN_D será habilitado. Se

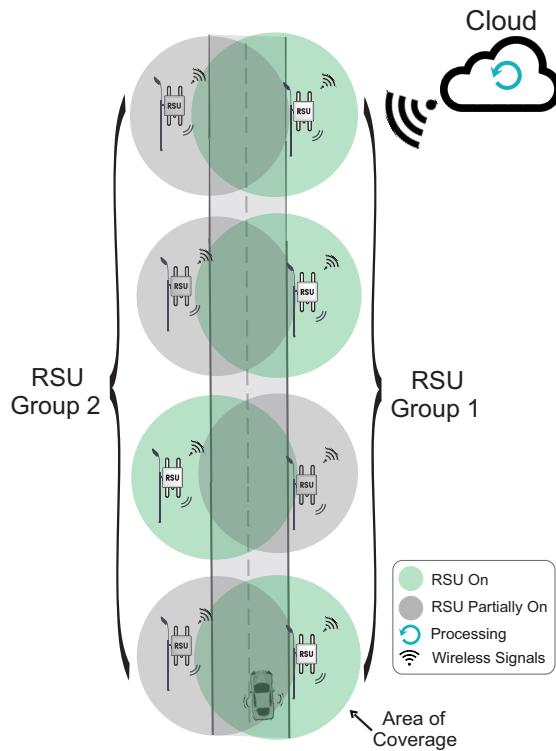


Figura 263 – Arquitetura estendida redundante, com RSUs secundárias parcialmente ativadas do outro lado da estrada.

o *token* principal e o redundante falharem simultaneamente, o sistema ficará indisponível, pois os *tokens* restantes em RSUN2_U_OC pertencem a outras seções da estrada. Para que o sistema esteja disponível, a soma das fichas em RSUN_U e RSUN2_U_OP deve ser igual ao número de fichas definido na marca N.

Quando a transição RSUN_MTTTR é disparada, um *token* em RSU_GROUP_1 será reparado, e um *token* em RSUN2_U_OP será enviado de volta para RSUN2_U_OC. No entanto, a transição T1 só é disparada se a soma dos *tokens* em RSUN_U, RSUN2_U_OP e RSUN2_T0_OP for maior que o número de *tokens* em RSUN_D. Se a soma dos *tokens* em RSUN_U e RSUN2_U_OP for menor que o valor em N, o sistema ficará indisponível. A tabela 175 detalha as condições de armazenamento usadas neste modelo.

Tabela 175 – Condições de guarda para o modelo SPN estendido com *Warm Standby*.

Transição	Expressão	Descrição
TO	((#RSUN2_D_OP+RSUN2_U_OP+ RSUN2_TO_OP)<#RSUN_D)	Quando a soma dos tokens em RSUN2 operacional é menor do que a quantidade de tokens desativados em RSUN.
T1	((#RSUN2_D_OP+RSUN2_U_OP+ RSUN2_TO_OP)>#RSUN_D)	Quando a soma dos tokens em RSUN2 operacional é maior do que a quantidade de tokens desativados em RSUN.

A disponibilidade do modelo estendido com *Warm Standby* é obtida pela Equação 25.6. Todos os compo-

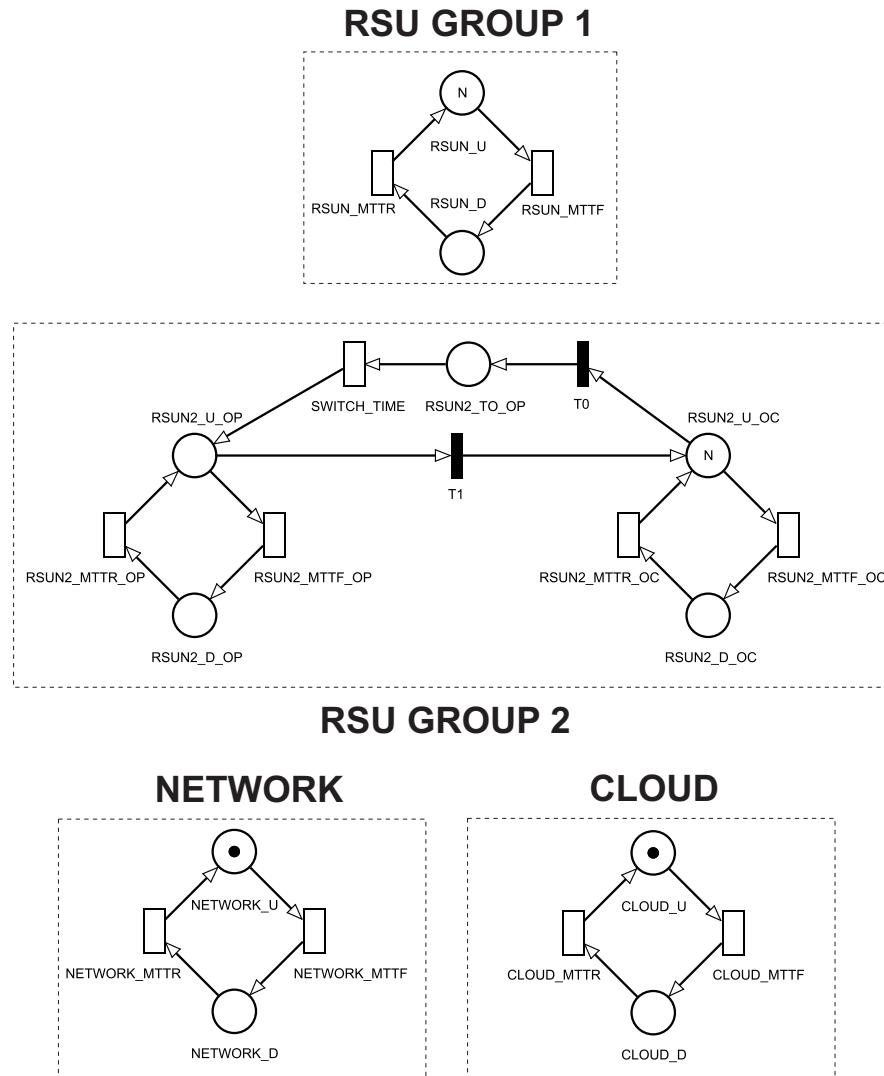


Figura 264 – Modelo SPN estendido com Warm Standby.

nentes do sistema devem estar ativos para que todo o sistema seja considerado disponível.

$$\begin{aligned} \mathbf{A} = P\{ & (\#RSUN_U + \#RSUN2_U_OP) = N \\ & \text{AND}(\#NETWORK_U > 0) \text{AND}(\#CLOUD_U > 0) \} \end{aligned} \quad (25.6)$$

25.8 Estudos de Caso Considerando Todos os Modelos de Disponibilidade

Esta seção apresenta um estudo de caso, incluindo a avaliação da disponibilidade dos modelos apresentados.

25.8.1 Parâmetros de Entrada

A tabela 176 apresenta os parâmetros de entrada para os modelos SPN. O grupo RSU e os tempos de falha e reparo da rede foram obtidos em (AUDU et al., 2017) e (ANDRADE; NOGUEIRA, 2018), respectivamente. O MTTF e o MTTR da nuvem foram obtidos hierarquicamente. Todos os valores de MTTR e MTTF para os componentes do frontend e do nó foram extraídos de (MELO et al., 2017b).

A tabela 176 mostra os parâmetros de entrada dos RBDs. O SWITCH_TIME foi configurado com 0,0833333 horas (ARAUJO et al., 2019). Na transição SWITCH_TIME do modelo estendido com *Cold Standby*, o tempo

Tabela 176 – Parâmetros de entrada de modelos SPN.

Componente	MTTF (horas)	MTTR (horas)
RSU	500	2
Network	83220	12
Cloud	125,89284516597696	0,9137943301091298
Frontend	180,72123967079426	0,9689053599591678
Node	414,9585492984169	0,7830543099654388

Tabela 177 – Parâmetros de entrada de modelos RBD.

Componente	MTTF (horas)	MTTR (horas)
Frontend	180,72123967079426	0,9689053599591678
Node	414,9585492984169	0,7830543099654388
Hardware	8760	1,6666700000000003
Operating System	2893	0,25
Hipervisor	2990	1
Instance Manager	788,4	1
Virtual Machine	2880	0,0191667
Platform Manager	788,4	1
Cluster Manager	788,4	1
Block-Based Manager	788,4	1
File-Based Manager	788,4	1

adoitado foi de 0,66666664 horas, oito vezes maior que o do modelo estendido com *Warm Standby*. Os dispositivos redundantes em *Cold Standby* estão totalmente desligados e demoram mais para ligar quando comparados ao *Warm Standby*, que já está parcialmente conectado. Finalmente, os valores MTTF e MTTR dos componentes do RSU GROUP 2 são iguais aos do RSU GROUP 1 para fins de simplificação. A única exceção é o MTTF do grupo RSU ocioso, que é 50% maior que o modelo básico, com um valor igual a 750h.

25.9 Resultados

A Figura 265 mostra a disponibilidade e o tempo de inatividade que foram calculados por análise estacionária usando a ferramenta *Mercury* (SILVA et al., 2015). Os cenários A, B e C na Figura 265(a) referem-se à disponibilidade dos três modelos SPN propostos: Base (97,68%), *Cold Standby* (98,86%) e *Warm Standby* (99,19%). Os cenários A, B e C, na Figura 265(b), referem-se ao tempo de inatividade dos três modelos SPN propostos: Base (202,94 h/a), *Cold Standby* (99,29 h/a) e *Warm Standby*, respectivamente (70,20 h/a). Portanto, está claro que a redundância melhorou a disponibilidade e o tempo de inatividade em comparação com o modelo básico.

25.10 Modelo Quatro - Modelo de Confiabilidade

A Figura 266 apresenta o modelo SPN de confiabilidade para a arquitetura VCC do cenário base. Este modelo é composto pelo RSU GROUP, NETWORK e CLOUD e possuem os mesmos parâmetros de entrada dos modelos de disponibilidade. O número de *tokens* em RSUN_U dá o número de RSUs ativos, então a marcação N indica quantos RSUs estão funcionando. Da mesma forma, o número de *tokens* em RSUN_D indica quantos RSUs estão inativos. A mudança do status ativo para inativo em cada RSU é acionada pelo gatilho de transição: RSUN_MTTR. Cada RSU pode operar de forma independente. Desde que o número de *tokens* em RSUN_U seja igual ao valor da marcação N, este componente estará disponível.

A rede estará funcionando quando tiver um *token* no local NETWORK_U. NETWORK_D indica uma rede indisponível. Tal como acontece com a rede, a nuvem está funcionando quando o local CLOUD_U tem um *token*. CLOUD_D com um *token* indica que a nuvem está inativa. Este modelo é uma variação do modelo SPN básico removendo as transições MTTR de todos os componentes. Depois que os componentes falham, eles

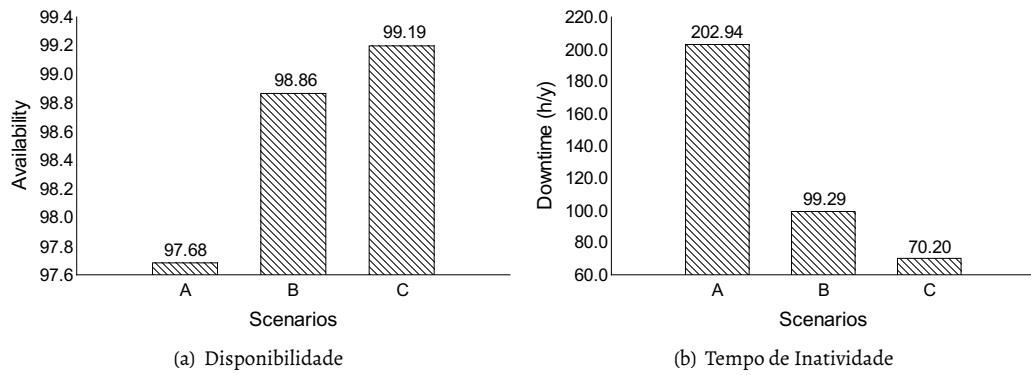


Figura 265 – Comparação dos níveis de disponibilidade e indisponibilidade em relação aos três modelos propostos.

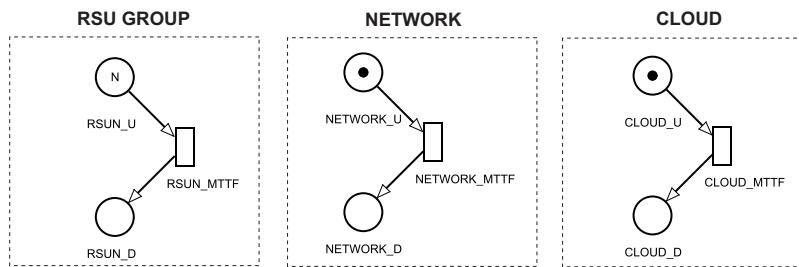


Figura 266 – Modelo SPN de confiabilidade para arquitetura VCC.

não podem ser reparados. Todos os parâmetros de entrada são iguais aos do modelo básico. Este modelo tem como objetivo mostrar o nível de confiança do sistema para continuar rodando ao longo do tempo. A equação 25.7 define a confiabilidade (R) do modelo.

$$R = 1 - P\{(\#RSUN_D > 0) \text{OR} (\#CLOUD_D > 0) \text{OR} (\#NETWORK_D > 0)\} \quad (25.7)$$

A confiabilidade também foi avaliada com foco no componente de maior impacto, o grupo RSU. A figura 267 mostra os níveis de confiabilidade do modelo, variando o MTTF do grupo RSU em três intervalos. O MTTF do grupo RSU foi variado entre o valor base menos 50% (A), valor base (B) e valor base mais 50% (C). A confiabilidade é baseada na probabilidade de um sistema estar operando por um determinado período. Portanto, tomando como exemplo o tempo de 350 horas, temos que a confiabilidade do sistema permanece em operação para o cenário A é 0,0002, o cenário B é 0,0036 e o cenário C é 0,0091. O cenário A não conseguiu completar 300 horas de confiabilidade, mostrando o impacto da redução de menos 50% no valor base do MTTF do grupo RSU. Os cenários B e C conseguiram completar 350 horas de confiabilidade. No entanto, o cenário C teve um nível de confiabilidade mais alto do que o B devido ao valor de MTTF mais alto. Portanto, quanto maior o tempo de falha, maior a confiabilidade, pois o sistema permanecerá em operação por mais tempo.

25.11 Conclusão

Este trabalho propôs modelos SPN e RBD para representar e avaliar a confiabilidade de um cenário de VCC com várias RSUs, auxiliadas pela computação em nuvem. Duas análises de sensibilidade foram realizadas para identificar o componente com maior impacto na disponibilidade do sistema. Os resultados da análise de sensibilidade indicaram o componente RSU GROUP como o mais impactante. A partir dessa análise, o modelo básico foi estendido em dois modelos redundantes, replicando o componente RSU GROUP com base

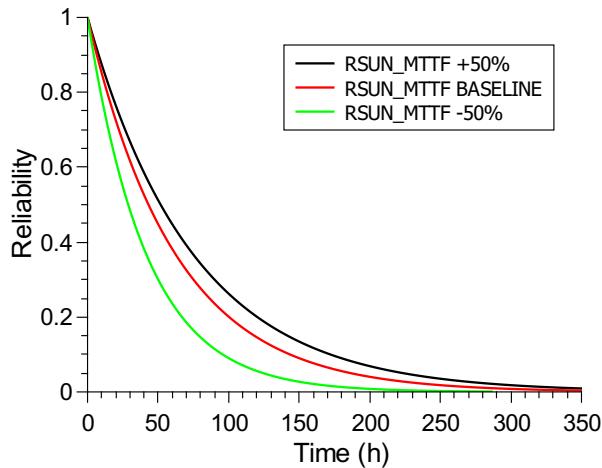


Figura 267 – Níveis de confiabilidade variando o componente RSUN: RSUN_MTTF -50%, RSUN_MTTF base e RSUN_MTTF +50%.

nas redundâncias *Cold Standby* e *Warm Standby* para melhorar a disponibilidade do sistema. Os resultados mostraram que replicar o componente mais sensível gera aumentos significativos na disponibilidade do sistema, aumentando de 97,68% no modelo base para 98,86% no modelo *Cold Standby* e 99,19% no modelo *Warm Standby*. O tempo de inatividade dos três modelos foi: base (202,94 h/a), *Cold Standby* (99,29 h/a) e *Warm Standby* (70,20 h/a). Houve uma queda substancial de 65,4% no tempo de inatividade do modelo básico para o modelo estendido com *Warm Standby* por ano. Além disso, foi realizada uma análise de confiabilidade. Como trabalhos futuros, pretendemos combinar os modelos de desempenho e disponibilidade, explorando questões de performabilidade no contexto de VCC.

Envelhecimento da plataforma Docker: uma avaliação sistemática de desempenho e previsão do consumo de recursos

Autores: Lucas Santos e Francisco Airton Silva

Resumo

O envelhecimento do software é um fenômeno sofisticado frequentemente manifestado pelo desempenho gradual e degradação de recursos em aplicativos e plataformas baseados em software. Trabalhos anteriores mostraram um progresso significativo na modelagem e avaliação, bem como análises experimentais de fenômenos de envelhecimento de software em sistemas de software, como sistemas de servidor virtualizado, mas sem consideração adequada para plataformas de software em uma plataforma como serviço (PaaS), como Docker. A plataforma Docker caracterizada por contêineres baseados em virtualização no nível do sistema operacional está sujeita ao envelhecimento do software sem exceções, exigindo, portanto, mais estudos. Nesse sentido, este Capítulo apresenta um estudo abrangente sobre envelhecimento e rejuvenescimento de software, juntamente com predição de comportamento em uma plataforma Docker direcionada. Um conjunto de experimentos de envelhecimento de software é realizado continuamente por trinta dias para investigar o acúmulo e a variação de falhas relacionadas ao envelhecimento nas configurações de hardware. Além disso, a metodologia Stress-Wait-Rejuvenation (SWARE) é adotada em experimentos de envelhecimento e rejuvenescimento de software para detectar sintomas de envelhecimento e investigar a eficácia das técnicas de rejuvenescimento de software em um único experimento. Desta forma, pode-se assimilar o consumo de recursos da plataforma de software sob a carga de trabalho. A observação mostra que o consumo de recursos permanece em um nível alto mesmo após a interrupção da carga de trabalho estressante. Outros resultados experimentais indicam graves problemas de envelhecimento da plataforma Docker desde seus estágios iniciais de execução. No entanto, as técnicas de rejuvenescimento - como simplesmente reiniciar o sistema operacional - podem reduzir drasticamente o fenômeno de consumo de recursos. Os dados experimentais são coletados e alimentados para vários modelos de previsão de dados com base em séries temporais e um algoritmo de aprendizado de máquina de memória de curto prazo longo (LSTM) para prever fenômenos de envelhecimento na plataforma Docker direcionada e comparar

modelos de previsão de envelhecimento. Os resultados da previsão implicam que o modelo LSTM supera na previsão do envelhecimento com uma precisão muito maior do que os modelos avaliados restantes. As descobertas deste estudo podem ajudar a facilitar a administração e manutenção de sistemas PaaS baseados em Docker com uma consideração adequada para encontrar fenômenos de envelhecimento de software e adotar técnicas de rejuvenescimento de software apropriadas em plataformas Docker.

26.1 Introdução

O envelhecimento do software existe inherentemente nas plataformas de software e expõe a degradação dos recursos computacionais e do desempenho, que está diretamente relacionado à execução do software ao longo do tempo (GROTTKE; MATIAS; TRIVEDI, 2008). O envelhecimento do software geralmente se manifesta pelo esgotamento dos recursos do sistema, corrupção de dados e acúmulo de erros. Além disso, o vazamento de memória é um dos efeitos comuns do envelhecimento do software (LIU; TAN; WANG, 2019; LIU; MENG, 2019). Vários trabalhos anteriores mostraram efeitos de envelhecimento em várias plataformas complexas, como servidores web (GROTTKE et al., 2006), sistemas operacionais (COTRONEO et al., 2010) e middle-wares (CARROZZA et al., 2010). O rejuvenescimento de software é uma contramedida proativa projetada para reduzir a probabilidade de falhas futuras causadas pelos efeitos do envelhecimento. A ideia subjacente do rejuvenescimento do software é pausar ou interromper a execução do software para investigar as falhas relacionadas ao envelhecimento e, em seguida, remover os erros acumulados ou os efeitos do envelhecimento acumulado e atualizar seu estado interno (HUANG et al., 1995). O rejuvenescimento de software pode ser acionado com base em uma variedade de medidas, por exemplo, técnicas de rejuvenescimento baseadas no tempo, que são acionadas regularmente após um intervalo de rejuvenescimento específico (POLINSKY et al., 2020; NGUYEN; KIM; PARK, 2014; BAI et al., 2020) ou técnicas de rejuvenescimento baseadas em eventos que são disparados se uma ação ou restrição, como um limite de consumo de recursos, for violado (ARAUJO et al., 2011b; CHEN et al., 2006).

Na era da computação em nuvem em geral e da virtualização, os aplicativos são distribuídos e hospedados em plataformas virtualizadas para otimizar o consumo de recursos de hardware. A adoção de máquinas virtuais, hipervisores e diferentes arquiteturas virtualizadas transformou as infraestruturas de computação na última década. No entanto, devido aos problemas de alta sobrecarga de recursos do paradigma de computação baseado em máquinas virtuais, a computação baseada em contêiner está surgindo como uma solução promissora para futuras infraestruturas virtualizadas (GILLANI; LEE, 2020). As plataformas baseadas em contêiner (por exemplo, Docker¹) são uma alternativa às soluções de virtualização tradicionais. No entanto, os estudos de envelhecimento relacionados ao uso de contêiners ainda são escassos e desafiadores, e precisam de uma investigação mais aprofundada. Pelo que investigamos na literatura, em (TORQUATO; VIEIRA, 2019) foi apresentado um dos primeiros trabalhos sobre envelhecimento e rejuvenescimento de software em plataformas Docker. Nossa estudo estende este trabalho em duas direções principais. (i) É uma questão de curiosidade como os efeitos do envelhecimento do software poderiam ser em diferentes configurações de hardware e, portanto, foram realizados novos experimentos para investigar os efeitos do envelhecimento do software em diferentes configurações de hardware. (ii) Com dados coletados experimentalmente sobre os efeitos do envelhecimento, como os comportamentos do envelhecimento poderiam ser previstos, foram adotados diferentes modelos para prever os comportamentos do envelhecimento do software.

Este Capítulo apresenta análises sistematicamente experimentais de uma plataforma Docker em relação aos aspectos de envelhecimento e rejuvenescimento de software. A Figura 268 mostra a metodologia de pesquisa deste estudo em três etapas principais. Etapa 1: realizamos a investigação e avaliação dos fenômenos de envelhecimento ao longo de um período específico sem interrupção, observando o consumo de recursos de RAM, Swap e CPU. Como uma extensão do trabalho (TORQUATO; VIEIRA, 2019), agora são avaliados três aspectos correlacionados em configurações distintas de computador (4 GB, 8 GB e 16 GB de RAM). Etapa 2: posteriormente, adotamos o Stress-Wait-Rejuvenation (SWARE) mostrado em (TORQUATO et al., 2018)

¹ <https://www.docker.com/>

para rejuvenescer a plataforma após um período de estagnação da carga de trabalho. Etapa 3: finalmente, são apresentados vários modelos de previsão usando os dados coletados com uma comparação.

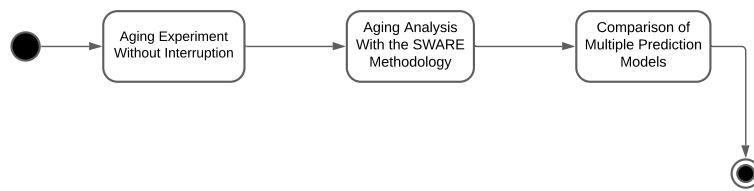


Figura 268 – Uma metodologia sistemática adotada no estudo dos fenômenos de envelhecimento do software da plataforma Docker.

26.2 Objetivos

26.2.1 Objetivo Geral

O objetivo geral deste Capítulo é apresentar experimentos e análises abrangentes sobre como os fenômenos de envelhecimento do software se manifestam e como o rejuvenescimento do software pode neutralizar o fenômeno do envelhecimento na plataforma Docker.

26.2.2 Objetivos Específicos

- Demonstrar através de experimentos o consumo crescente de memória RAM e outros recursos computacionais devido ao envelhecimento na plataforma Docker.
- Verificar se existe eficácia na ação de rejuvenescimento de software no objetivo de trazer de volta o sistema a um estado estável e livre do acúmulo de erros na plataforma Docker.
- Comparar diferentes técnicas de predição de dados que ajudem prever o avanço do envelhecimento na plataforma Docker.

26.3 Organização do Trabalho

O restante desse Capítulo está organizada da seguinte forma. Na Seção 26.4, alguns conceitos para a compreensão deste Capítulo. Na Seção 26.5, são discutidos e apresentados trabalhos relacionados selecionados na literatura. Na Seção 26.6, é apresentado o primeiro estudo sobre envelhecimento de longo prazo sem interrupção. Na Seção 26.7, o segundo estudo sobre estagnação da carga de trabalho e subsequente rejuvenescimento do sistema é apresentado. Na Seção 26.8, é mostrada a implementação e comparação de vários modelos de predição com séries temporais e um algoritmo de aprendizado de máquina. Na Seção 26.10, são apresentadas as conclusões importantes deste estudo.

26.4 Referencial Teórico

Essa seção apresenta os conceitos necessários para a compreensão do artigo. A primeira parte trata do envelhecimento e do rejuvenescimento do software. A segunda parte fornece alguns detalhes sobre a arquitetura da plataforma Docker. A terceira e quarta partes abordam modelos de previsão baseados em séries temporais.

26.4.1 Envelhecimento e rejuvenescimento de software

O fenômeno conhecido como envelhecimento de software pode ser definido como um acúmulo sucessivo de erros ou como a degradação dos recursos computacionais disponíveis no sistema. O envelhecimento

pode ocorrer devido a falhas acumuladas, geralmente na execução de longa duração, que geralmente é causada por erros de programação durante o desenvolvimento do software (AVIZIENIS et al., 2004). A persistência de adversidades causadas pelo envelhecimento do software tende a comprometer a disponibilidade e confiabilidade do sistema. As consequências podem afetar a satisfação dos usuários, perdas financeiras ou até mesmo a perda de vidas (GROTTKE; MATIAS; TRIVEDI, 2008). O envelhecimento do software geralmente ocorre de forma lenta e gradual. Normalmente, seus efeitos se tornam perceptíveis depois que o software está em execução por um período indeterminado de tempo. O envelhecimento é comumente causado pelo alto custo computacional necessário para manter softwares sofisticados em operação constante (CASTELLI et al., 2001). Recursos como a CPU tornam-se monopolizados devido ao uso excessivo de processos específicos com prioridade mais alta. A quantidade de memória disponível no sistema ao longo do tempo também tende a se tornar escassa. Durante o desenvolvimento de software, a tendência para o envelhecimento do software pode ser observada. Por exemplo, pode haver erros em um trecho de código que acabarão por levar ao envelhecimento. Um erro de desenvolvimento típico é a não liberação de memória exigida pelo sistema. A parte da memória não utilizada deve ser liberada para uso por outros processos, evitando o acúmulo de memória, ou o chamado vazamento de memória (CASTELLI et al., 2001) (LIU; TAN; WANG, 2019) (LIU; MENG, 2019).

Uma contramedida adotada para contornar o problema do envelhecimento pode ser definida como rejuvenescimento de software. O rejuvenescimento é a aplicação de técnicas preventivas para forçar o sistema afetado a retornar a um estado estável e livre de erros (HUANG et al., 1995). Exemplos de ações executadas como um rejuvenescimento incluem reiniciar um aplicativo específico ou um conjunto completo de aplicativos. Como o rejuvenescimento do software incorre em sobrecarga e tempo de inatividade do sistema, otimizar a política de rejuvenescimento do software é relevante e crucial para a implementação dessa técnica. Por exemplo, procedimentos de rejuvenescimento frequentes podem promover o desempenho do sistema. No entanto, pode aumentar o tempo de conclusão de uma tarefa ou até mesmo falhar a tarefa se for em tempo real (cumprir um prazo de uma tarefa específica); assim, otimizando o agendamento de rejuvenescimento para maximizar a probabilidade de conclusão bem-sucedida da tarefa é significativa. Outras medidas de otimização de desempenho incluem disponibilidade do sistema, tempo médio de conclusão da tarefas, probabilidade de perda da transação e tempo médio de resposta (LEVITIN; XING; HUANG, 2019).

26.4.2 Plataforma Docker

Nos últimos anos, o uso de containers tem sido apontado como uma alternativa viável para substituição de máquinas virtuais (VMs) tradicionais, pois os containers demandam menos recursos computacionais. Os contêineres são uma alternativa de baixo custo à virtualização e também oferecem desempenho superior às máquinas virtuais (GILLANI; LEE, 2020), (BAYSER; CERQUEIRA, 2017). A plataforma Docker² é um sistema de código aberto e um dos mais conhecidos gerenciadores de contêineres da atualidade, além de ser uma das ferramentas pioneiras na área.

A arquitetura Docker permite a criação de ambientes virtualizados de forma independente e isolada. Ao contrário de uma VM típica, um contêiner Docker não precisa de um hipervisor ou de um sistema operacional completo. Os contêineres do Docker compartilham o kernel com o sistema operacional host, aumentando o desempenho e diminuindo o custo dos recursos computacionais. O Docker é dividido em três camadas que se comunicam entre si: **Cliente, Host e Registro**. A camada Cliente permite que os usuários interajam e gerenciem containers instanciados por meio de uma arquitetura cliente / servidor. A comunicação ocorre diretamente com o Host, onde estão os containers. O Host pode ser uma máquina física ou virtual e é o ambiente onde os aplicativos serão executados com contêineres Docker. O ambiente Host é formado pelos seguintes componentes: Daemon Docker, imagens, contêineres, rede e volumes. O Registro é o serviço que fornece os repositórios para baixar imagens Docker (CHUNG et al., 2016).

A Figura 269 compara a implantação do aplicativo usando um hipervisor e um contêiner. Como mostra a figura, a implementação baseada em hipervisor é ideal quando os aplicativos na mesma nuvem exigem

² <https://www.docker.com/>

sistemas operacionais ou versões de sistema operacional diferentes (por exemplo, RHEL Linux, Debian Linux, Ubuntu Linux, Windows 2000, Windows 2008, Windows 2012). A abstração deve estar no nível da VM para fornecer essa capacidade de executar diferentes versões do sistema operacional (BERNSTEIN, 2014).

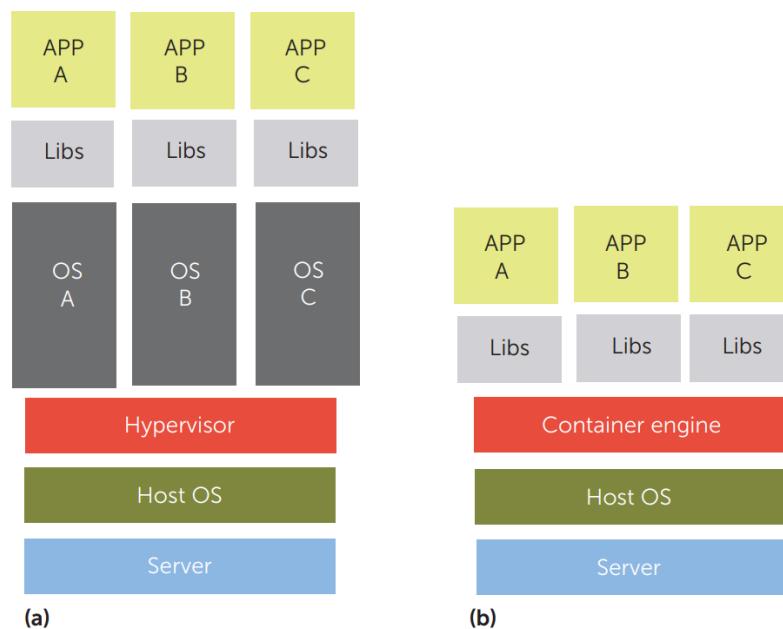


Figura 269 – Comparação de (a) hipervisor e (b) implantações baseadas em contêiner. Uma implantação baseada em hipervisor é ideal quando os aplicativos na mesma nuvem exigem diferentes sistemas operacionais ou diferentes versões de sistema operacional; em sistemas baseados em contêiner, os aplicativos compartilham um sistema operacional, portanto, essas implantações podem ser significativamente menores em tamanho.

26.4.3 Modelo de previsão com séries temporais

Uma série temporal é um conjunto de dados indexados sucessivamente em um intervalo de tempo (SHUMWAY; STOFFER, 2017). As séries temporais são utilizadas nos mais diversos campos de estudo, como estatística, economia, matemática, meteorologia, agricultura, etc. (CRYER; KELLET, 1991). É possível reconhecer padrões e tendências de previsão apenas analisando séries de dados matematicamente. A partir de um modelo matemático, é possível gerar uma nova sequência de dados.

As séries temporais são geralmente caracterizadas pelos seguintes fatores: sazonalidade, tendência e ruído. Sazonalidade é o padrão que os dados seguem ao longo de algum tempo. A tendência está relacionada ao comportamento que os dados da série temporal tendem a seguir: crescente, decrescente ou constante (ANDERSON, 2011). O ruído se refere a dados aleatórios que não fazem parte da tendência ou sazonalidade. Neste trabalho, quatro modelos clássicos foram usados para prever dados em séries temporais: ARIMA, Drift, Holt e Holt-Winters. A escolha de tais modelos baseou-se em um estudo dos principais preditores usados no envelhecimento do software. O levantamento dos modelos de previsão utilizados no estado da arte é apresentado no seção de trabalhos relacionados (seção 26.5).

- O modelo **ARIMA** é um dos modelos mais utilizados para previsão de dados em séries temporais, o modelo é descrito pela seguinte equação:

$$y' = c\phi_1y'_{t-1} + \dots + \phi_py'_{t-p} + \theta_1e_{t-1} + \dots + \theta_qe_{t-q} + e_t \quad (26.1)$$

, onde y'_t é uma série diferencial que pode ser integrada mais de uma vez. ϕ é o coeficiente de regressão. θ é o coeficiente da média móvel, e é o erro. O modelo ARIMA é composto por três valores (p, d, q), onde p é o grau autoregressivo, d significa quantas vezes será integrado, q é o parâmetro de média móvel (PEARCE, 1987).

- O modelo **Drift** é descrito pela seguinte equação (HYNDMAN; ATHANASOPOULOS, 2018):

$$yT + \frac{h}{T-1} \sum_{t=2}^T (yt - yt-1) = yT + h\left(\frac{yT - y1}{T-1}\right) \quad (26.2)$$

, onde T é a duração da série temporal. $y1$ é o primeiro valor de observação, yT é o último valor de observação e h é o horizonte de previsão.

- O modelo **Holt** consiste em dois componentes: suavização e o componente de captura de tendência. O modelo é descrito pela seguinte equação (HYNDMAN; ATHANASOPOULOS, 2018):

$$\hat{y}_T = l_t + hb_t \quad (26.3)$$

, onde l_t é atualmente o componente de suavização t . b_t é o componente de tendência no momento t . α é o parâmetro de suavização, $0 \leq \alpha \leq 1$. β é o parâmetro de suavização para tendência, $0 \leq \beta \leq 1$. y é o valor da previsão. O b_t é uma média ponderada da tendência que é estimada no momento t , com base em l_{t-1} e b_{t-1} , que fazem parte da tendência estimada no passado.

- O modelo **Holt-Winters** é uma extensão do método de Holt para capturar sazonalidade adicionando um novo componente (KALEKAR, 2004). A seguinte equação descreve o modelo Holt-Winters:

$$\hat{y}_t + h|t = l_t + hb_t + s_{t-m+h_m^+} \quad (26.4)$$

, onde h_m^+ significa $[b(h-1) \bmod m] + 1$, o que garante que a sazonalidade use as observações mais recentes. O l_t é o fator de suavização e mostra a média ponderada entre a sazonalidade e a previsão não sazonal no momento t . α é o coeficiente de suavização, β é o coeficiente de tendência e s é o coeficiente de sazonalidade (HYNDMAN; ATHANASOPOULOS, 2018). O b_t é a tendência e é o mesmo que o método Holt. S_t é sazonalidade e mostra a média ponderada no índice sazonal atual $y_t - l_{t-1} - b_{t-1}$, onde γ é o coeficiente de suavização (KALEKAR, 2004).

26.4.4 Modelo de previsão com aprendizado de máquina

O aprendizado de máquina é essencialmente uma abordagem à inteligência artificial que usa qualquer sistema que seja adepto do aprendizado como os humanos, ou seja, com base na experiência. Como um cérebro humano, ele visa reconhecer os padrões e se atualizar para aplicá-los em decisões futuras. Assim, ele desafia as antigas tradições de alimentar dados nos sistemas por meio da programação e aprende por meio de exemplos. As decisões em Aprendizado de Máquina são orientadas por dados ao invés de algoritmos (VIRMANI et al., 2020).

O aprendizado de máquina tem sido amplamente usado em mineração de dados e permite que o sistema aprenda padrões e modelos estruturais úteis a partir de dados de treinamento (XIE et al., 2018). O ML geralmente consiste em duas fases principais: a fase de treinamento e a fase de teste. A fase de treinamento é onde os métodos de aprendizado de máquina são aplicados para aprender o modelo do sistema a partir dos dados de treinamento. A fase de teste é onde o sistema pode obter o resultado estimado para cada nova entrada usando o modelo treinado.

Os algoritmos de ML são normalmente classificados em três categorias: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço. Na aprendizagem supervisionada, exemplos de entradas e saídas desejadas são apresentados ao computador. O objetivo da aprendizagem supervisionada é

aprender uma regra geral que mapeia as saídas das entradas. No aprendizado não supervisionado, nenhum tipo de saída é apresentada ao algoritmo de aprendizado, permitindo que ele encontre um padrão nas entradas fornecidas. Finalmente, a aprendizagem por reforço faz com que um programa de computador interaja com um ambiente dinâmico (BELL, 2020).

Neste trabalho, foi adotado o algoritmo ML denominado Long Short Term Memory (LSTM). O algoritmo LSTM é um tipo de rede neural recorrente, que provou ter sucesso em uma série de problemas diferentes (SAK; SENIOR; BEAUFAYS, 2014), (SAINATH et al., 2015), (MALHOTRA et al., 2015), (CHENG; DONG; LAPATA, 2016). No entanto, até onde se sabe, o algoritmo LSTM não foi aplicado para prever o envelhecimento do software. Os algoritmos LSTM são adequados para classificar, processar e prever com base em dados de série temporal, pois pode haver atrasos de duração desconhecida entre eventos importantes em uma série temporal. Esses aspectos permitem utilizar o LSTM como o algoritmo principal para previsão de envelhecimento do software.

26.5 Trabalhos Relacionados

Tabela 178 – Trabalhos Relacionados

Trabalhos	Plataforma	Métricas	Duração	Rejuv.	Múltiplas Configurações	Método de Previsão
(TORQUATO; VIEIRRA, 2019)	Docker	Memory RAM, Swap, CPU	26 dias	Sim	Não	Sem Previsão Aplicada
(YAKHCHI et al., 2015)	Computador Desktop	CPU, Process, Memória	Não Informado	Não	Não	MLP, LR, M5, RT, SMO, LMS, GP
(MAGALHÃES; SILVA, 2010)	Servidor de Aplicação	Memória RAM, CPU	140 Minutos	Não	Não	MLP, Naive Bayes, J48
(GROTTKE et al., 2006)	Apache Web Server	Tempo de Resposta, Memória RAM, Swap	25 dias	Não	Não	Mann-Kendall test
(ARAUJO et al., 2011a)	Eucalyptus Cloud Platform	Memória RAM, Swap, Memory Fragmentation	10 dias	Sim	Não	Linear Regression
(LI; VAIDYANATHAN; TRIVEDI, 2002)	Apache Web Server	Memória RAM, Swap, Disk, CPU , Processos, Tempo de Resposta	7 dias, 25 dias, 14 dias	Não	Não	Linear Regression, ARIMA
(ARAUJO et al., 2011b)	Eucalyptus Cloud Platform	Memória RAM, Swap, CPU, Disco, Processos	3 dias	Sim	Não	The Linear Trend Model, Quadratic Trend Model, Growth Curve Model, S-Curve Trend Model
(MELO et al., 2017a)	OpenStack Cloud Platform	Memória RAM, Swap, CPU, Disco, Processos	7 dias	Não	Não	Linear Trend Model, Growth Curve Model, Quadratic Trend Model, S-Curve Trend Model
(MENG et al., 2016)	I2EE Application Server	Memória heap da JVM, CPU, Tempo de resposta, Taxa de Transferência	19,7 horas	Sim	Não	Sem Previsão aplicadas
(PEREIRA et al., 2018)	Aplicação Android, Eucalyptus Cloud Platform, Computador Desktop	Memory Swap	35 minutos, 30 ddias, 70 minutos	Não	Sim	ARIMA, Drift, Holt, Holt-Winters, Linear Regression, Simple Exponential Smoothing
Este Trabalho	Docker	Memória RAM, Swap, CPU	30 dias	Sim	Sim	ARIMA, Drift, Holt, Holt-Winters, LSTM

Essa seção apresenta alguns trabalhos relacionados. A Tabela 178 resume uma comparação de trabalhos relacionados considerando seis aspectos: Plataforma, Métricas, Duração, Rejuvenescimento, Múltiplas Configurações e o Método de Predição.

Os primeiros critérios de comparação focaram na **plataforma** avaliada. Os trabalhos relacionados utilizaram várias plataformas. No entanto, algumas plataformas foram utilizadas por mais de um artigo, como o Apache (LI; VAIDYANATHAN; TRIVEDI, 2002), (GROTTKE et al., 2006) e Eucalyptus (ARAUJO et al., 2011a), (ARAUJO et al., 2011b). As tecnologias relacionadas à computação em nuvem têm sido amplamente estudadas no envelhecimento de software, podendo-se notar que além do Eucalyptus, a plataforma OpenStack também foi citada. No entanto, apenas o trabalho de Torquato et al. (TORQUATO; VIEIRA, 2019) realizou experimentos com foco na plataforma Docker. Como mencionado anteriormente, este artigo é uma extensão do trabalho de Torquato.

As **metrics** mais estudadas sobre envelhecimento foram: RAM, Swap memory e CPU — essas mesmas métricas adotadas neste trabalho. Outras métricas também foram estudadas, mas com menor frequência, como o tempo de resposta (LI; VAIDYANATHAN; TRIVEDI, 2002), (GROTTKE et al., 2006), (MENG et al., 2016). O tempo de resposta é essencial para ser avaliado quando aspectos como Qualidade da Experiência negativa e indisponibilidade causam problemas aos usuários. Também foi observada a utilização da métrica **processos**, que se refere aos processos secundários observados durante o monitoramento do sistema (LI; VAIDYANATHAN; TRIVEDI, 2002), (ARAUJO et al., 2011a), (MELO et al., 2017a). Os processos avaliados nestes três trabalhos referem-se aos processos relacionados ao software específico, que no caso de uma plataforma em nuvem; por exemplo, tem vários processos relacionados ao serviço. Em relação ao Docker, tanto o presente trabalho quanto o trabalho de (TORQUATO; VIEIRA, 2019) observaram apenas o processo “dockerd”.

A **duração** dos experimentos também é uma característica importante. Dependendo do tempo de observação, os padrões de envelhecimento podem ou não ser identificados. O menor período foi realizado pelo trabalho de (ARAUJO et al., 2011a) com apenas três dias de observação. O período mais prolongado foi aplicado pelo experimento proposto neste trabalho. No entanto, os trabalhos de (TORQUATO; VIEIRA, 2019) e (GROTTKE et al., 2006) foram muito próximos, com 26 e 25 dias respectivamente. Se o pesquisador está interessado em encontrar o ponto de esgotamento de recursos, é fundamental levar semanas ou mesmo vários meses. A carga de trabalho, claro, é um ponto igualmente relevante.

Software **rejuvenescimento** é uma ação fundamental para trazer um sistema que esgotou seus recursos de volta a um estado estável. Alguns estudos optaram por analisar os efeitos do envelhecimento do software e a eficácia do rejuvenescimento. Apenas três estudos não exploraram o rejuvenescimento: (LI; VAIDYANATHAN; TRIVEDI, 2002), (GROTTKE et al., 2006) e (MELO et al., 2017a).

Outro aspecto foi **múltiplas configurações de hardware**. Em ambientes de teste, deve-se observar as configurações de hardware e como elas tendem a influenciar o avanço do envelhecimento do software. Este trabalho se destaca de todos os demais analisados. Essa proposta de trabalho é a única a realizar um estudo experimental de envelhecimento e rejuvenescimento em uma plataforma utilizando mais de uma configuração de hardware em um mesmo experimento. Consideramos este recurso uma contribuição porque em nosso trabalho anterior (TORQUATO; VIEIRA, 2019) o uso de apenas uma configuração de hardware se limita a tirar certas conclusões.

O último critério a ser observado foi o **métodos de predição** adotados. É possível perceber que alguns métodos foram utilizados em mais de um trabalho, como o modelo ARIMA (utilizado por (LI; VAIDYANATHAN; TRIVEDI, 2002) e (PEREIRA et al., 2018)) ou a Regressão Linear (utilizado por (ARAUJO et al., 2011a) e (LI; VAIDYANATHAN; TRIVEDI, 2002)). Alguns artigos adotaram modelos de predição clássicos e também modelos de aprendizado de máquina (MAGALHÃES; SILVA, 2010) (YAKHCHI et al., 2015). Nenhum dos trabalhos relacionados usou o modelo LSMT.

26.6 Experimentos de envelhecimento do Docker sem interrupção

Essa seção apresenta um experimento inicial para investigar os efeitos do envelhecimento na plataforma Docker por muitos dias e sem interrupção.

26.6.1 Metodologia

Para o ambiente de teste, três computadores foram usados para obter uma gama mais ampla de dados. As configurações de hardware de cada computador são mostradas na Tabela 179. O sistema operacional Ubuntu Server 18.04.4 LTS foi instalado em todos os computadores. O experimento consistiu em submeter os três computadores a uma carga de trabalho constante usando a plataforma Docker. Ao mesmo tempo, esses recursos eram monitorados. Todo o experimento foi realizado em trinta dias consecutivos e, ao final do experimento, os dados foram coletados e analisados.

Tabela 179 – Configurações de Hardware dos Computadores A, B & C

Computer	Processor	RAM Memory	Swap Memory	Hard Disk
Computer A	Intel i3 de 3.10 GHz	4GB	2GB	500GB
Computer B	Intel i3 de 3.10 GHz	8GB	4GB	500GB
Computer C	Intel Xeon 3.3 GHz	16GB	4GB	1TB

O Docker versão 17.05.0-ce foi instalado nos três computadores. Em um ciclo constante, um conjunto de containers foi criado e monitorado. Todas as imagens de contêiner continham os softwares Owncloud³ e MySQL⁴. Para simular o uso constante de computadores, foi criada uma carga de trabalho a ser executada igualmente em cada computador. O objetivo era estressar o sistema e acelerar o fenômeno do envelhecimento. A carga de trabalho foi desenvolvida na forma de *shell script*, consistindo em uma repetição sucessiva de operações para inicializar e finalizar os containers Docker com as imagens mencionadas. O Algoritmo ?? apresenta os detalhes do processo de execução.

Portanto, a carga de trabalho consistia em instanciar cinquenta contêineres usando o comando *docker create*. A cada dez segundos, o número de contêineres criados é verificado. Quando o número de contêineres não atingiu cinquenta, dois novos contêineres são instanciados e o processo é repetido até que a condição anterior se torne falsa. Após completar 50 contêineres, todos eles são encerrados com o comando *docker stop* e removidos com o comando *docker rm*. Então, o processo espera cinquenta segundos para ser executado novamente. O sistema é monitorado paralelamente à execução da carga de trabalho; para isso, são usados comandos de outro script de shell. Este script, por sua vez, usa chamadas de sistema Linux padrão, como *free -mh*, *top* e *grep*, para coletar dados relacionados às métricas definidas como indicadores de envelhecimento do software: memória principal, Swap e CPU.

26.6.2 Resultados e análise do experimento

A Figura 270 apresenta os resultados da análise considerando os três computadores (A, B e C). Dois gráficos são apresentados para cada computador, um com RAM + Swap e outro com CPU. Havia sintomas de esgotamento de recursos do sistema, o que sugere envelhecimento do software.

As figuras 270(a) e 270(b) mostram, respectivamente, o uso de RAM + Swap e a porcentagem de uso de CPU do computador A. A figura 270(a) indica que após 200 horas, o uso de RAM atinge um pico de aproximadamente 1.400 MB. Quase ao mesmo tempo, o uso de memória Swap no computador A começa a crescer exponencialmente, enquanto o uso de RAM só diminui após atingir seu pico. Por outro lado, a porcentagem de uso da CPU no nível do usuário (Figura 270(b)) não apresentou variações significativas, permanecendo constante.

As figuras 270(c) e 270(d) mostram o resultado para o computador B. O computador B, devido à sua configuração de hardware mais avançada, demorou mais para ativar a memória Swap. A figura 270(c) revela que o uso de RAM cresce exponencialmente até aproximadamente 300 horas. Seu crescimento diminuiu a partir desse ponto até atingir o pico de aproximadamente 1.800 MB. Após 500 horas, o consumo de memória diminui ligeiramente. A figura 270(c) também mostra que logo após 300 horas, a memória Swap é ativada, mas com crescimento discreto. Em relação à CPU (Figura 270(d)), da mesma forma que o computador A, o computador B não apresentou variações significativas no uso da CPU.

³ Owncloud: <https://owncloud.org/>

⁴ MySQL: <https://www.mysql.com/>

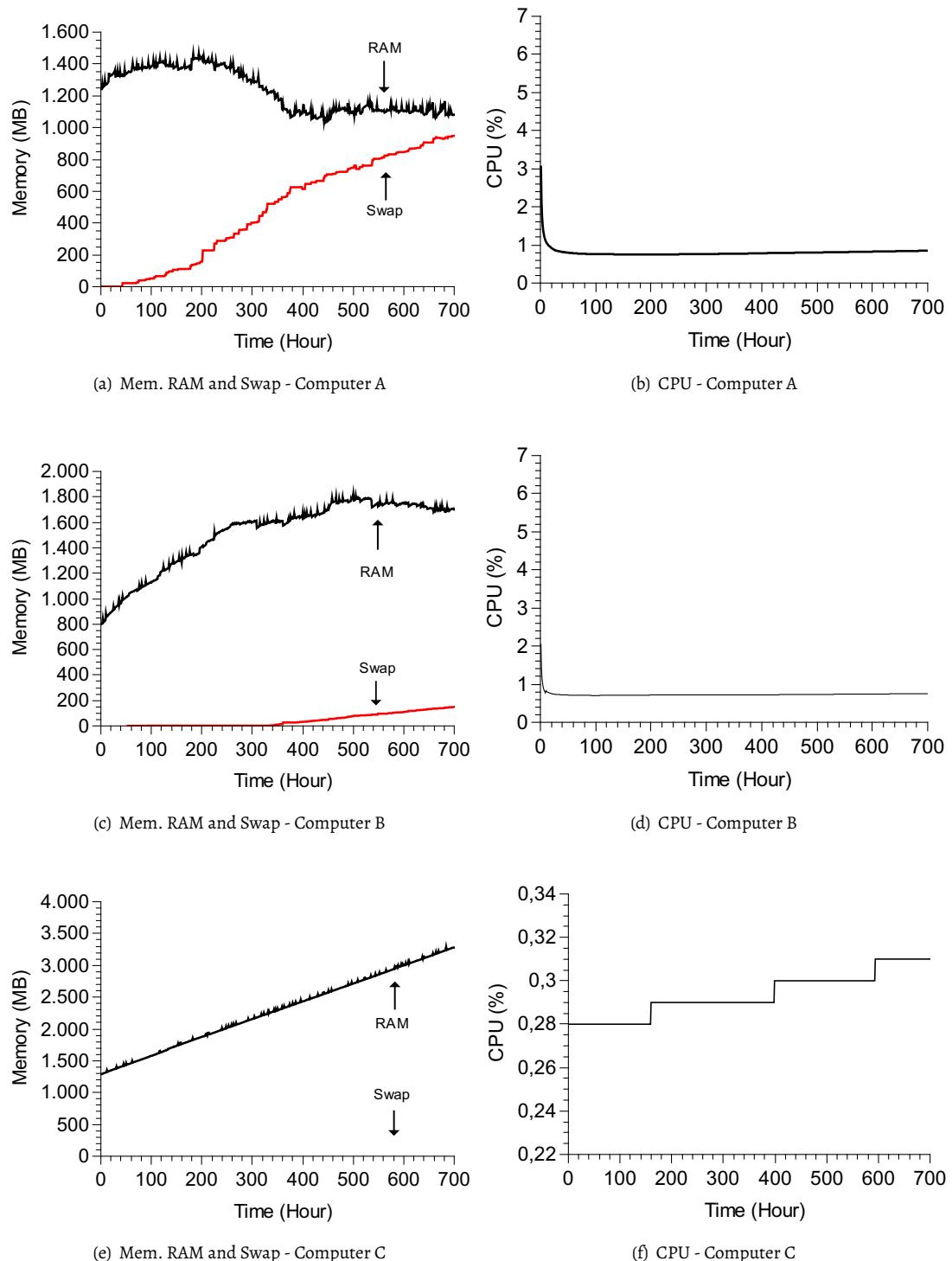


Figura 270 – Utilização geral de recursos dos computadores A, B e C

As figuras 270(e) e 270(f) mostram o resultado para o computador C. O computador C possui o hardware mais avançado entre os três computadores. Portanto, o computador C teve um crescimento constante do consumo de RAM, mas sem acionar Swap. Mesmo sem acionar a troca, esse crescimento na RAM sugere efeitos de envelhecimento. Este comportamento linear é semelhante ao crescimento nos momentos iniciais

dos computadores A e B. A CPU possui valores de consumo desprezíveis (Figura 270(f)), apesar de apresentar alterações.

Esse experimento ininterrupto foi necessário para observar o comportamento do consumo de recursos ao longo de muitos dias. Apesar de apresentar resultados significativos no envelhecimento, o rejuvenescimento não foi aplicado. O próximo seção visa observar esse outro aspecto, que é monitorar o consumo de recursos interrompendo a geração de workload e aplicando o reinício do sistema operacional.

26.7 Experimentos de envelhecimento do Docker com metodologia SWARE

Essa seção apresenta um segundo experimento de envelhecimento de software na plataforma Docker usando a metodologia SWARE. O experimento buscou observar como os recursos se comportam quando uma carga de trabalho sintética fica estagnada após um determinado período. Além disso, uma ação de rejuvenescimento foi realizada em seguida.

O rejuvenescimento pode ser executado em diferentes níveis, desde a reinicialização de um componente individual até a reinicialização de todo o nó no qual o aplicativo está sendo executado. A técnica de rejuvenescimento utilizada em nesse trabalho é bastante simples e uma das mais populares de acordo com o (COTRONEO et al., 2014). O sistema operacional de reiniciado, incluindo todos os processos em execução nele. Essa estratégia é eficiente porque todos os processos são interrompidos e nenhum buffer ou cache permanece carregado.

O presente experimento foi executado nos mesmos três computadores do experimento anterior.

26.7.1 Metodologia

Este trabalho adotou uma metodologia denominada SWARE. O objetivo da metodologia SWARE é analisar os indicadores de envelhecimento do software e observar os efeitos das ações de rejuvenescimento. O SWARE foi proposto por (TORQUATO et al., 2018); posteriormente, foi aplicado considerando a plataforma Docker em (TORQUATO; VIEIRA, 2019), mas com algumas restrições que são detalhadas na seção de trabalho relacionada (26.5). A metodologia é caracterizada por três fases: **Stress**, **Wait** e **Rejuvenation**.

Na fase **Stress**, o computador está sujeito a uma carga de trabalho constante como forma de exaurir os recursos do sistema e ativar bugs antigos. Na fase **Wait**, a carga de trabalho é interrompida e as consequências causadas pela fase de Stress são analisadas. Espera-se que os efeitos do envelhecimento do software acumulados durante a fase de estresse permaneçam durante a fase de espera. Na fase **Rejuvenation**, técnicas de rejuvenescimento de software são aplicadas. O objetivo é trazer o sistema de volta a um estado estável e sem acúmulo de erros. As figuras 271(a) e 271(b) exemplificam resultados na fase Wait sem envelhecimento e com envelhecimento.

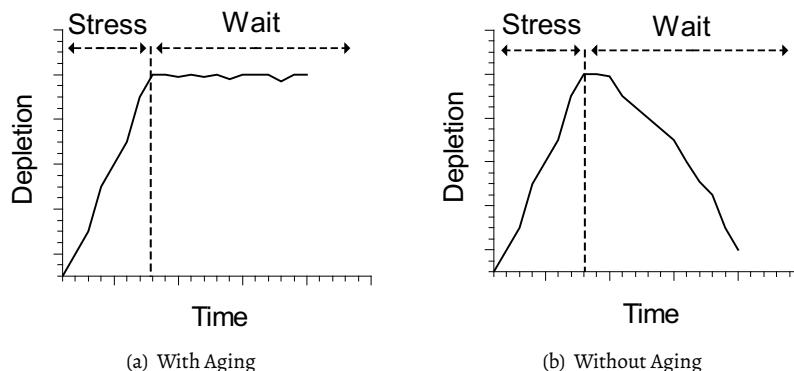


Figura 271 – Exemplificação da Fase de Wait

Vale ressaltar que para a metodologia SWARE, os sintomas de envelhecimento do software são reconhecidos quando o consumo de recursos na fase de espera permanece constante ou aumenta. Na fase de

Rejuvenescimento, as Figuras 272(a) e 272(b) exemplificam os resultados de rejuvenescimento eficiente e rejuvenescimento ineficiente.

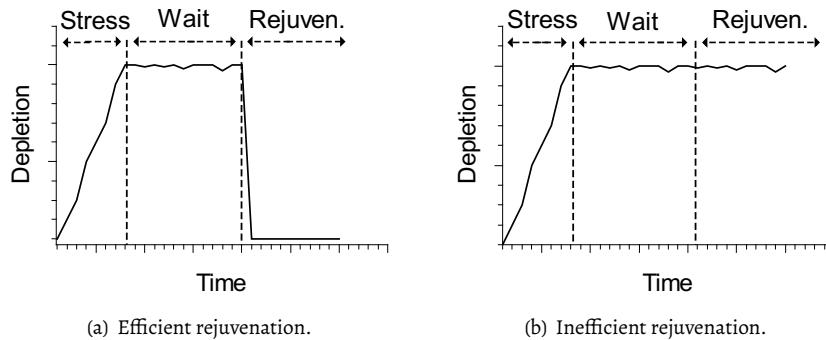


Figura 272 – Exemplificação da Fase de Rejuvenescimento.

Neste trabalho, foi aplicado a metodologia SWARE no mesmo ambiente de teste do experimento anterior (ou seja, o mesmo aparato computacional e o mesmo script de monitoramento). No entanto, as mudanças na geração de carga foram aplicadas (ver Algoritmo ??). Neste experimento, para agilizar os resultados, foram instanciados 480 containers em cada ciclo, com intervalo de dez segundos para cada instanciação de dezesseis containers e cinquenta segundos de espera após a exclusão dos containers. Como no experimento anterior, o tempo de longo prazo foi suficiente para entender o comportamento do envelhecimento de longo prazo, e agora com a metodologia SWARE, alguns dias foram suficientes para atingir o objetivo.

A fase de estresse durou mais de dois dias. A fase de espera durou um dia, e a fase de rejuvenescimento durou dezessete horas, conforme mostrado na Figura 273. Vale ressaltar que apenas a plataforma Docker e os scripts foram executados durante todo o experimento.

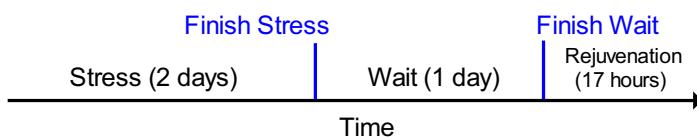


Figura 273 – Duração de Cada Fase da Abordagem SWARE

26.7.2 Resultados e Análise dos Experimentos

Esta seção apresenta os resultados do experimento utilizando a metodologia SWARE. As mesmas métricas mencionadas acima foram observadas em cada computador: RAM Memory, Swap Memory, CPU e dockerd process memory. Cada gráfico possui marcadores para cada fase do experimento. O final da fase de estresse é marcado com uma linha azul - o que significa que a geração de carga de trabalho foi interrompida. O final da fase de espera é marcado com uma segunda linha azul, com o sistema operacional reiniciado a partir desse ponto.

A Figura 274 mostra os resultados da RAM nos computadores A, B e C. Há um consumo crescente de RAM durante a fase de estresse nos três computadores. No entanto, enquanto os computadores A e B (menos potentes) já iniciam o processo com alto consumo (≈ 1100 MB), o computador C inicia a partir de ≈ 200 MB. Curiosamente, o consumo de pico do computador C é igualado no final desta fase com o computador A. Na fase de Espera, os três computadores apresentaram sintomas de envelhecimento à medida que o consumo se manteve constante. Os computadores A e C permaneceram praticamente no mesmo nível de consumo. Na verdade, os computadores A e B manifestaram um comportamento de leve queda, porém o computador B expôs uma queda de forma mais significativa. Quanto ao rejuvenescimento, os computadores B e C tiveram

queda significativa no consumo. No entanto, no computador A houve quase uma diminuição no consumo. Portanto, deve-se atentar para a recuperação dos recursos obtidos com o rejuvenescimento.

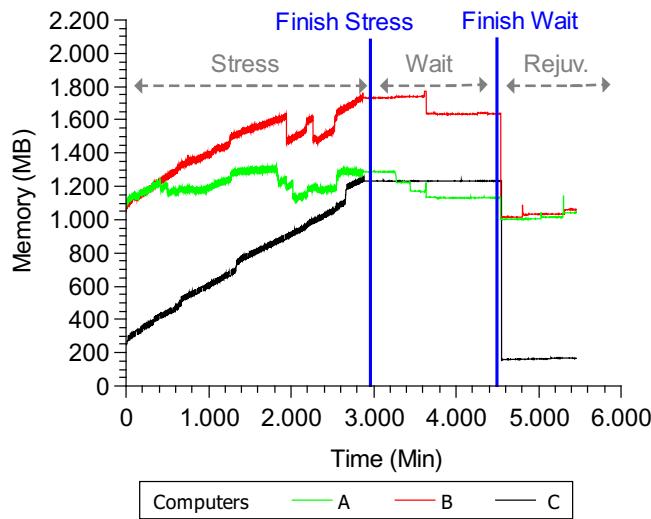


Figura 274 – Memória RAM - Computadores A, B e C

A Figura 275 mostra os resultados do uso de memória Swap nos computadores A, B e C. Os computadores B e C mostraram níveis superficiais de consumo de memória Swap. O computador B durante a fase de estresse atingiu um pico máximo de apenas ≈ 2 MB. Embora tenha um crescimento mínimo, o computador B mostrou um consumo constante durante a fase de espera. A memória Swap nunca foi ativada pelo sistema no computador C, mantendo exatamente 0 MB de memória durante todo o experimento. O computador A experimentou um consumo crescente de memória Swap durante a fase de estresse, atingindo um pico de ≈ 320 MB. A troca do computador A também permaneceu consistentemente alta durante a fase de espera. O computador A apresentou o envelhecimento mais significativo considerando a fase de estresse e espera. Em relação ao rejuvenescimento, o computador A mostrou que os níveis de consumo de memória Swap diminuíram drasticamente. Os computadores B e C já estavam em níveis de consumo superficiais quando o rejuvenescimento foi acionado, o que não foi tão impactante.

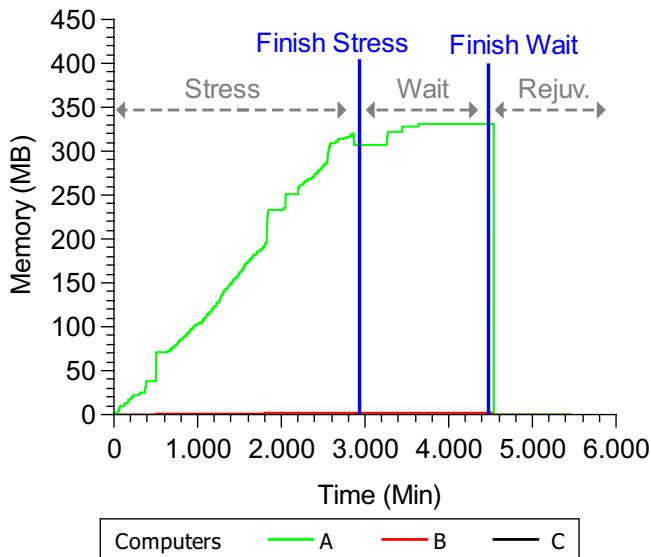


Figura 275 – Swap - Computadores A, B e C

A Figura 276 mostra a porcentagem do uso total da CPU para os computadores A, B e C. Os computadores A e B mostraram uso de CPU muito semelhante. O uso da CPU permaneceu baixo no computador A durante a fase de estresse, chegando a $\approx 8\%$. O computador B atingiu um consumo máximo de $\approx 4\%$ durante a fase de estresse. Durante a fase de espera, a porcentagem de ambos os computadores permaneceu ligeiramente constante. Durante a fase de espera, o sistema deve estar ocioso. Portanto, o uso do processador indica que ainda existem (quando não deveriam ter) tarefas que consomem recursos. Para o computador C, os resultados do uso da CPU novamente não foram significativos. O consumo começou com $\approx 1\%$ de uso, permanecendo baixo e estável na maior parte do tempo e diminuindo ligeiramente durante a fase de espera. O rejuvenescimento se mostrou eficiente quanto ao uso da CPU dos três computadores, apresentando apenas pequenos picos durante a inicialização.

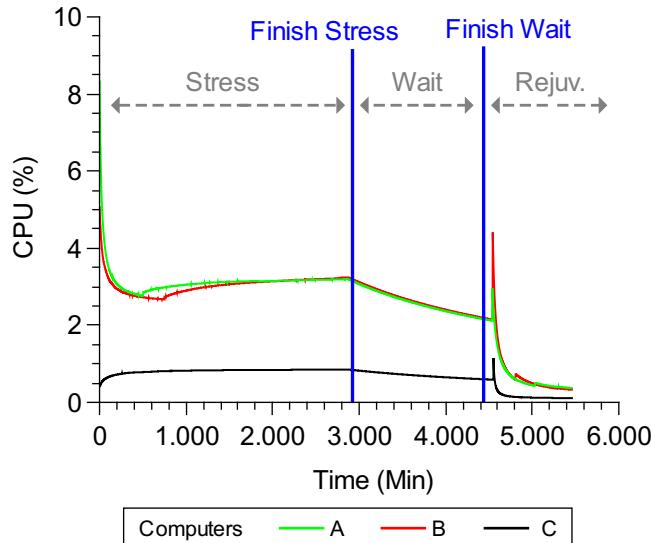


Figura 276 – CPU - Computadores A, B e C

A Figura 277 mostra a porcentagem de uso de memória do processo dockerd nos computadores A, B e C. A porcentagem de uso de memória do processo dockerd no computador A aumentou durante a fase de estresse, atingindo um pico de $\approx 7\%$, mantendo um uso constante durante a fase de espera. No computador B, os resultados foram semelhantes aos do computador anterior apenas na fase inicial. Na fase de espera, o consumo do computador B permaneceu maior. No computador C, a porcentagem de uso de memória do processo dockerd foi a mais baixa dos três computadores. Houve um uso crescente na fase de Estresse, atingindo um pico de $\approx 5\%$ e um uso constante na fase de Espera, repetindo o comportamento dos casos anteriores, mas de forma menor. O rejuvenescimento provou ser eficiente em todos os três computadores, fazendo com que a porcentagem de uso de memória no processo dockerd caísse para $\approx 0\%$.

Os resultados apresentados nesta seção corroboram os resultados obtidos em (TORQUATO; VIEIRA, 2019). Vale ressaltar que a fase de estresse visa apenas forçar o sistema a consumir mais recursos em um curto período. Assim, espera-se que os possíveis bugs de envelhecimento do software sejam ativados precocemente. Nos resultados obtidos, fica claro que após apenas dois dias de estresse, os efeitos do envelhecimento já são perceptíveis. Isso sugere que a ativação de bugs antigos pode ocorrer em situações típicas de consumo de recursos. Por exemplo, o consumo de CPU na fase de estresse não excedeu 6 %. No entanto, a fase de espera mostra que alguns efeitos se acumularam após o estresse.

26.8 Previsão e Análise de Envelhecimento de Software no Docker

Essa seção apresenta a adoção de modelos distintos para a previsão do comportamento do envelhecimento e sua comparação abrangente. Cinco modelos de previsão foram usados, quatro baseados em séries temporais e um baseado em aprendizado de máquina. O objetivo é definir o modelo mais eficaz para prever os efeitos

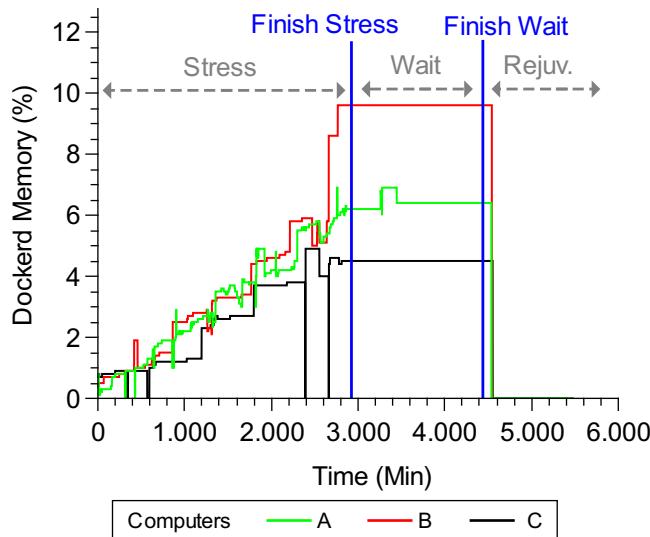


Figura 277 – Processor Dockerd - Computadores A, B e C

do envelhecimento. A série de dados usada foi a mesma do experimento de longo prazo sem interrupção (Seção 26.6). Três métricas de erro foram adotadas para comparar a precisão de cada um desses modelos:

- O *Erro Médio Absoluto (MAE)* mede a magnitude do erro médio em um conjunto de previsões, independente de sua direção. MAE é a média das diferenças absolutas entre a previsão e a observação real, em que todas as diferenças individuais têm peso igual. A seguinte equação descreve o MAE:

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (26.5)$$

f_i significa a previsão, y_i significa o valor real e n significa o número de pontos ajustados.

- O *Erro Percentual Médio Absoluto (MAPE)* prevê tendências de séries temporais, também é usado como uma função de perda para problemas de regressão em aprendizado de máquina. A seguinte equação descreve o MAPE:

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{A_i - F_i}{A_i} \right| \quad (26.6)$$

onde A_i é o valor real, F_i é o valor da previsão e n é o número de pontos ajustados.

- O *Root Mean Square Error (RMSE)* é uma medida frequentemente usada das diferenças entre os valores (valores amostrais e populacionais) previstos por um modelo ou estimador e os valores observados. O RMSE é descrito pela seguinte equação:

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(A_i - F_i)^2}{n}} \quad (26.7)$$

onde A_i é o valor da previsão, F_i é o valor real e n é o número de pontos ajustados.

26.8.1 Modelos de séries temporais

Esta seção apresenta os resultados da previsão. Quatro modelos foram adotados para comparar modelos baseados em séries temporais: ARIMA, Drift, Holt e Holt-Winters. Os dados foram divididos em duas partes, 80% para treinamento e 20% para teste.

As figuras 278, 279, 280 apresentam os resultados gráficos para os computadores A, B e C, respectivamente. Os erros de análise desses experimentos são mostrados respectivamente nas Tabelas 180, 181 e 182. Quatro gráficos com os modelos de previsão propostos são apresentados para cada computador. A linha vermelha representa os dados de teste. A linha azul representa o comportamento previsto. Os gráficos mostram o comportamento previsto para o uso de RAM do computador. A área cinza representa os intervalos de confiança da previsão. A área cinza mais clara corresponde a um intervalo de confiança de 95%. A área cinza mais escura corresponde a um intervalo de confiança de 80%.

A Figura 278, em geral, mostra que no computador A a previsão não foi tão precisa devido à alta variabilidade dos dados. As áreas do intervalo de confiança são significativamente amplas. A tabela 180 mostra os valores de erro para cada modelo para o computador A. Observa-se que os modelos ARIMA e Drift apresentam os menores valores de erro MAPE e MAE, respectivamente, indicando, neste caso, serem os melhores modelos de predição entre os comparados. É interessante notar que os gráficos mostram um ligeiro declínio no uso de RAM, o que coincide com os resultados obtidos no experimento de longo prazo.

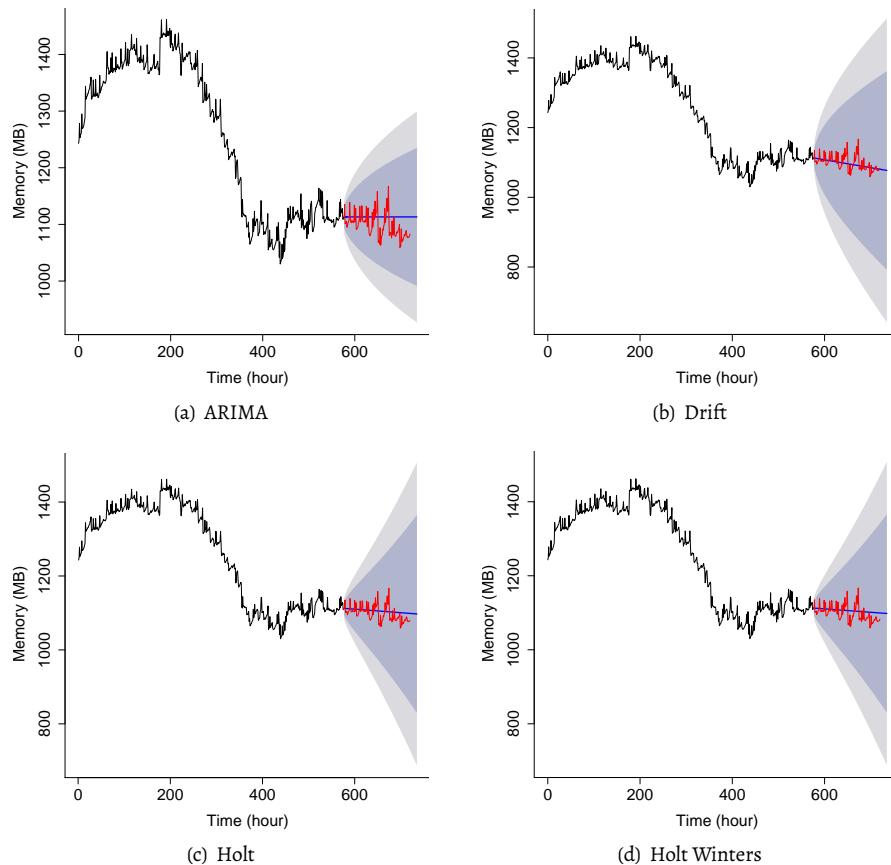


Figura 278 – Previsão de Memória - Computador A

Tabela 180 – Error MAPE, MAE e RMSE - Computador A

	ARIMA	Drift	Holt	Holt Winters
MAPE	9.6759	9.9013	9.9373	9.9198
MAE	21.567	12.9896	16.6951	17.1066
RMSE	25.3219	18.3073	20.2859	20.5696

A Figura 279 apresenta os resultados para o computador B. A variabilidade de dados do computador B é menor do que no computador A. Portanto, os intervalos de confiança são menores. No entanto, o resultado previsto para o modelo Drift (Figura 279(b)) foi visualmente muito diferente da linha de teste vermelha. A

Tabela 181 – Error MAPE, MAE e RMSE - Computador B

	ARIMA	Drift	Holt	Holt Winters
MAPE	7.0647	6.8637	7.0009	7.0104
MAE	23.6454	139.2827	41.2283	26.6698
RMSE	28.9456	162.2631	47.6148	32.4169

Tabela 182 – Error MAPE, MAE e RMSE - Computador C

	ARIMA	Drift	Holt	Holt Winters
MAPE	5.2318	5.0691	5.3423	5.3172
MAE	4.8141	24.1744	8.5791	4.8405
RMSE	10.5692	25.5645	13.1176	10.5478

tabela 181 mostra os valores de erro para cada modelo. Os valores MAPE são estatisticamente equivalentes, sendo inconclusivos. No entanto, para MAE e RMSE, o erro de Drift foi muito maior do que os outros. O modelo ARIMA obteve os menores valores MAE e RMSE.

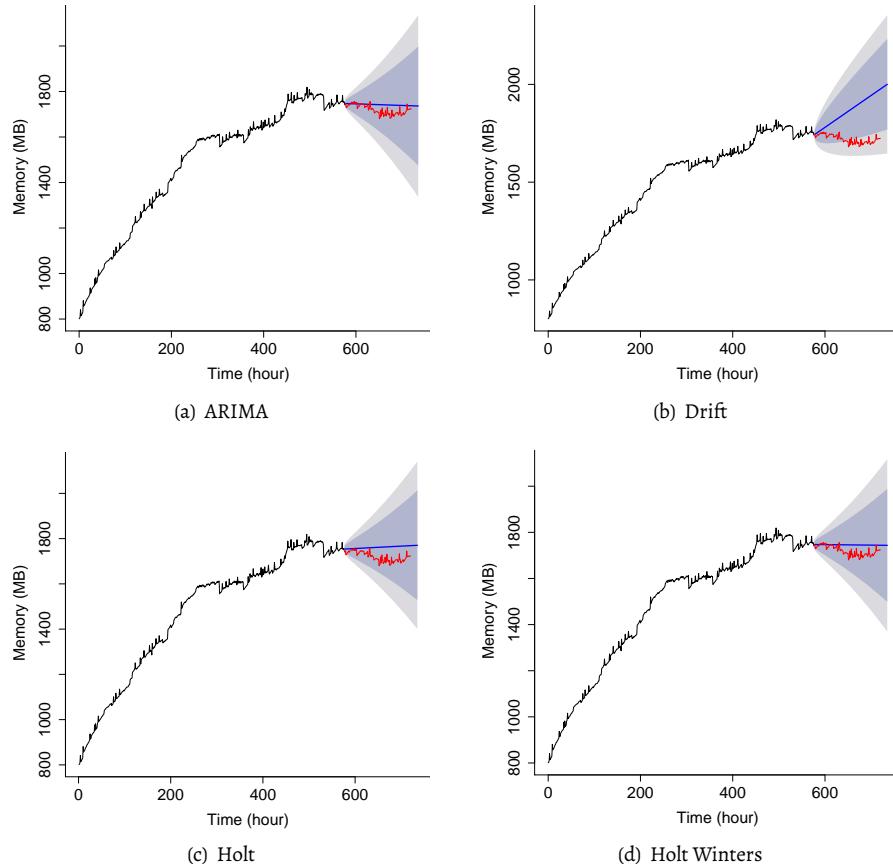


Figura 279 – Predição de Memória - Computador B

A Figura 280 mostra os resultados de predição para o computador C. Novamente, apenas observando os gráficos, o modelo Drift possui os intervalos de confiança mais significativos, e os outros três resultados são muito semelhantes. A tabela 182 mostra os valores de erro para cada modelo. Mais uma vez o ARIMA obteve os melhores resultados.

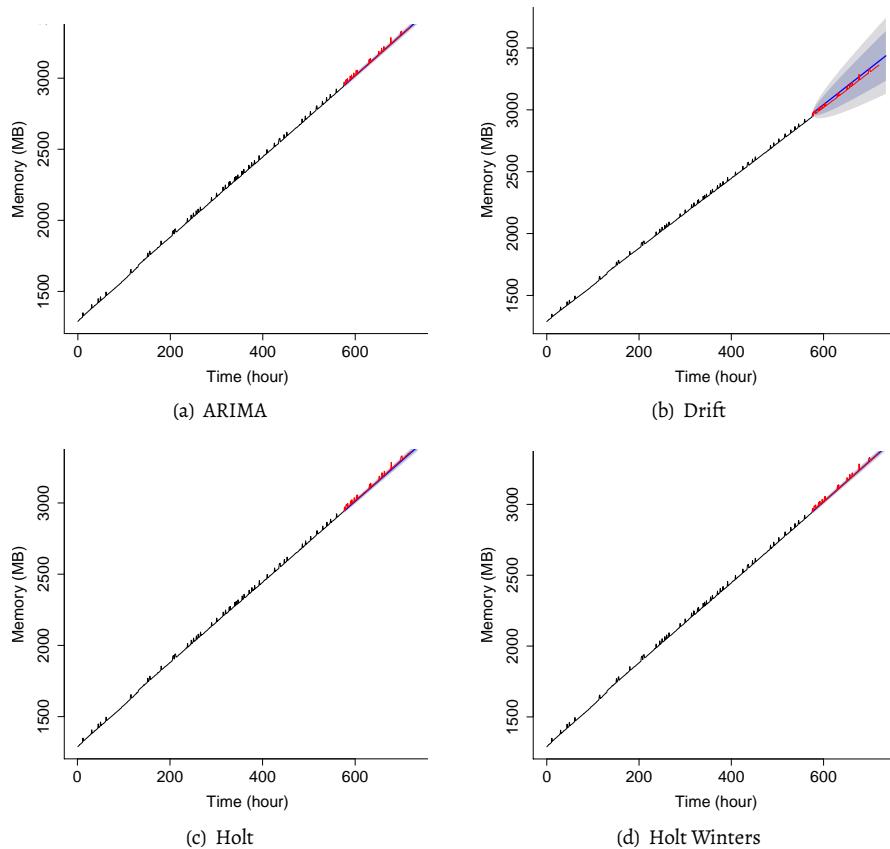


Figura 280 – Previsão de Memória - Computador C

26.9 Algoritmo de Previsão com Machine Learning

Os resultados anteriores foram obtidos por meio de modelos de previsão de séries temporais mais tradicionais. No entanto, algoritmos de aprendizado de máquina também podem ser aplicados para prever o envelhecimento do software de maneira dinâmica e não determinística. Neste trabalho, também foi adotado o algoritmo Long Short Term Memory (LSTM). O LSTM pode distinguir entre ocorrências recentes e distantes, dando pesos diferentes para cada ocorrência. LSTM também é capaz de ignorar memórias que podem ser consideradas irrelevantes para prever a próxima saída. Desta forma, o LSTM pode lidar com longos fluxos de dados em comparação com outras redes neurais recorrentes, bem adequadas para séries temporais de envelhecimento de software. Como nas previsões da seção anterior, as séries temporais foram divididas em treinamento (80%) e conjuntos de teste (20%). Os parâmetros que obtiveram os melhores resultados após os testes com LSTM foram os seguintes: Taxa de aprendizagem: 0,5, Estações: 50, Neurônios: 10.

A Figura 281 mostra a previsão do consumo de memória do Computador A. Os dados de previsão mostram que o algoritmo foi capaz de realizar uma previsão com alta precisão. A parte inferior do gráfico mostra melhor o quanto precisa foi a previsão. As linhas são muito próximas umas das outras. Os valores de erro obtidos foram: RMSE = 15,6263, MAPE = 1,0978 e MAE = 11,0796.

A Figura 282 apresenta o gráfico do resultado da previsão do consumo de memória do Computador B. Os resultados da previsão foram semelhantes aos resultados anteriores. Porém, apresentou valores menores. Os valores de erro foram: RMSE = 13,3021, MAPE = 0,5694 e MAE = 9,8236. O RMSE apresentou valor inferior à previsão anterior, demonstrando que o algoritmo apresentou maior índice de concentração de dados na linha de melhor ajuste. MAPE teve um valor inferior ao da previsão anterior. O valor mais baixo é porque o consumo de memória do Computador B segue um padrão mais linear. O MAE apresentou um

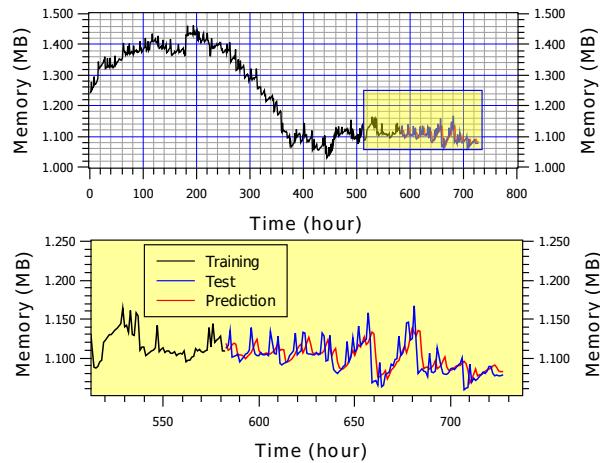


Figura 281 – Predição de Memória - Computador A

valor inferior porque a variação entre o valor observado e a previsão diminuiu. O resultado apresentado mostra que o algoritmo obteve uma previsão melhor para o Computador B do que para o Computador A, de forma ainda mais precisa.

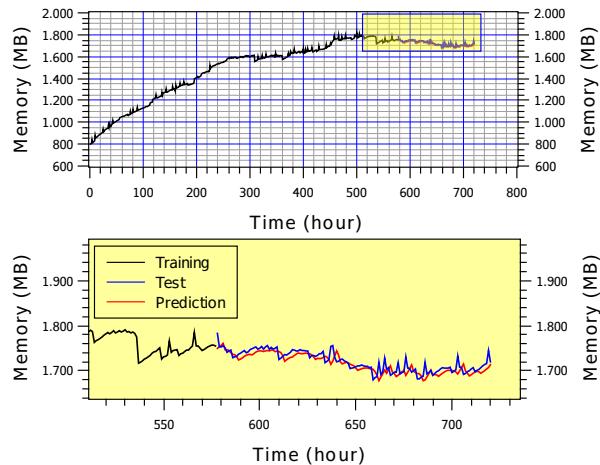


Figura 282 – Predição de Memória - Computador B

A Figura 283 mostra os resultados para o Computador C. Os valores de erro foram: RMSE = 11,1971, MAPE = 0,2381 e MAE = 7,4624. MAPE mostrou um valor inferior às previsões anteriores devido ao padrão muito linear. O MAE também obteve um valor inferior às previsões anteriores.

26.9.1 Análise e Comparação de Modelos de Previsão

Esta seção apresenta um resumo dos resultados obtidos a partir dos modelos de previsão múltiplos. Para cada preditor usado, três métricas de erro foram observadas: RMSE, MAE e MAPE. A eficácia de cada forma de previsão foi comparada.

A Figura 284(a) mostra os resultados do erro RMSE médio que cada preditor obteve nos computadores A, B e C. Observa-se que o modelo de previsão baseado em Aprendizado de Máquina (LSTM) possui a média dos menores valores de erros, enquanto o modelo de série temporal Holt-Winters possui a segunda menor média de erros. Portanto, para esta métrica de erro, é possível afirmar que o modelo LSTM é o modelo de previsão adequado entre os demais analisados para predizer o avanço do envelhecimento.

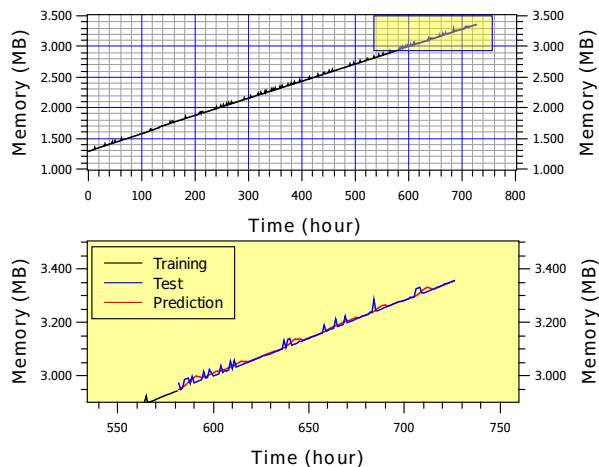


Figura 283 – Predição de Memória - Computador C

A figura 284(b) mostra os resultados dos erros médios do MAE. Como esperado, o LSTM obteve um erro médio ainda menor em comparação aos modelos de predição clássicos. O modelo de Holt-Winters obteve o segundo menor erro médio do MAE. Portanto, de acordo com o erro MAE, o modelo LSTM é o modelo de previsão mais adequado para prever o envelhecimento do Docker.

A figura 284(c) mostra os resultados do erro MAPE. O LSTM obteve novamente o menor erro médio. Os outros preditores foram equivalentes em média. Os resultados para os modelos clássicos foram próximos a 7,0, enquanto o LSTM foi próximo a 0,0. Portanto, considerando os três erros, o LSTM é o preditor mais eficaz para prever os efeitos do envelhecimento da docker.

26.9.2 Discussão

Imagens / aplicações diferentes: neste estudo, foram investigados cerca de dez artigos intimamente relacionados, nos quais apenas um explorou a variação de HW. Os resultados dos experimentos neste trabalho evidenciaram que este critério de comparação é altamente impactante nos comportamentos de envelhecimento. É possível notar que imagens e / ou aplicativos hospedados de forma distintas podem ter diferentes consumos de recursos, principalmente se tiverem tamanhos completamente diferentes e aplicativos internos implantados. No entanto, o foco deste estudo está na importância do hardware para o investimento em infraestrutura e implantação. É notável um amplo local de pesquisa sobre a investigação do envelhecimento em imagens distintas e / ou aplicativos hospedados em trabalhos futuros.

Abordagens de aprendizado de máquina: neste estudo, a hipótese geral é que o Docker pode apresentar efeitos de envelhecimento ao longo do tempo, e o tipo de hardware deve ser considerado. Os resultados experimentais apoiam ambas as hipóteses. A partir daí, este estudo apresentou a ideia de prever como seria o comportamento do envelhecimento, com base no traçado anterior. Usar algoritmos de série temporal é uma abordagem comum na literatura antiga, mas usar o modelo LSTM não é. Todas as fases deste trabalho são complementares como uma contribuição ao campo do envelhecimento de software. Serão encontradas pesquisas futuras produtivas sobre a adoção de diferentes abordagens de aprendizado de máquina (por exemplo, outras técnicas de regressão de última geração) para a previsão do comportamento do envelhecimento.

26.10 Conclusão

Este Capítulo apresentou análises sistematicamente experimentais de envelhecimento e rejuvenescimento de software em uma plataforma Docker e comportamentos de envelhecimento de software previstos com base em modelos de série temporal e um algoritmo de aprendizado de máquina. A metodologia sistemática de experimentos foi organizada em três etapas principais. Na primeira etapa, foi realizado um experimento

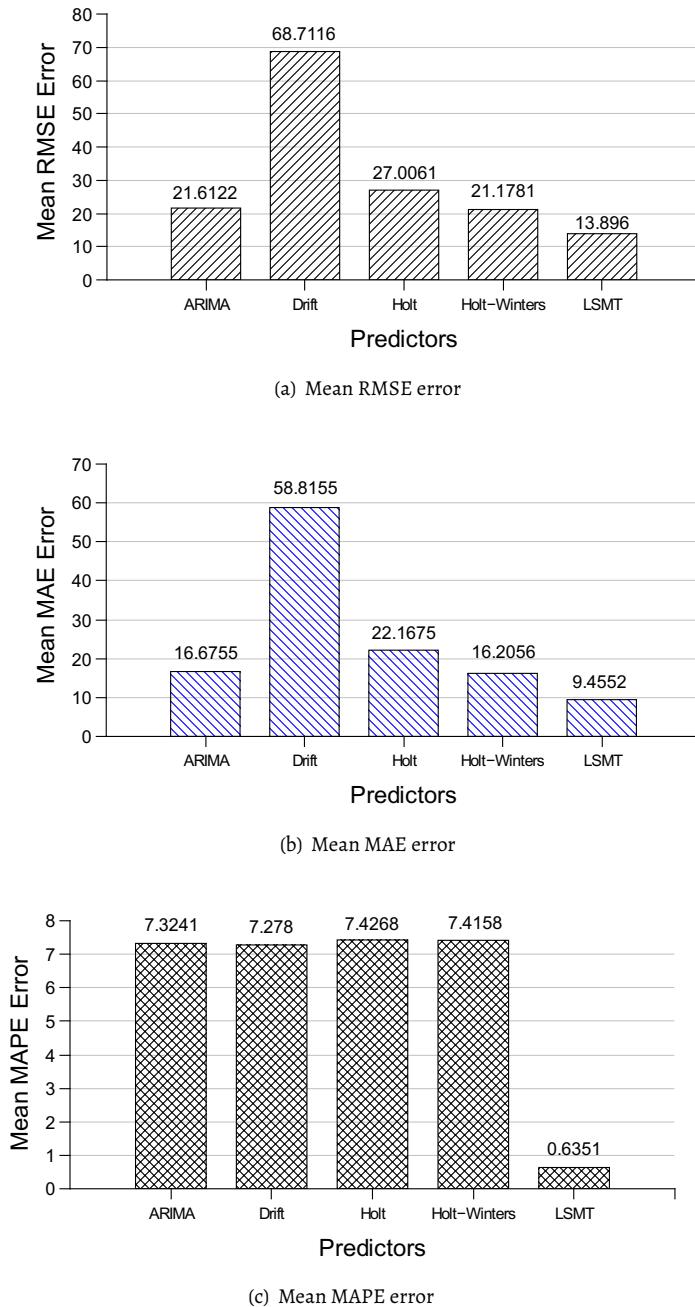


Figura 284 – Resultados Resumidos Comparando os Cinco Modelos de Previsão

de trinta dias para observar fenômenos de envelhecimento dentro da plataforma, que operam continuamente, sem interrupções. O experimento utilizando a metodologia SWARE foi realizado na segunda etapa com considerações mais amplas na etapa após a estagnação da carga de trabalho e ações de rejuvenescimento. Na terceira etapa, diferentes técnicas de previsão do envelhecimento foram adotadas e comparadas. Os resultados do experimento de longo prazo forneceram evidências claras dos efeitos do envelhecimento de Docker. Porém, é importante ressaltar que a memória Swap é eficaz apenas nos casos em que os recursos computacionais de uma máquina são baixos. Os resultados usando o método SWARE também indicaram os sintomas de envelhecimento do software. Apenas no computador de recursos mais baixos, ocorreu uma queda significativa na RAM ao interromper a carga de trabalho. Na máquina mais poderosa (16 GB

de RAM), o consumo de recursos permaneceu constante após a interrupção da carga de trabalho. Assim, havia sintomas de envelhecimento apenas para computadores com uma capacidade de recursos mais alta quando a carga de trabalho para se apenas a RAM fosse considerada. Os resultados da análise dos modelos de predição mostraram que o algoritmo LSTM de aprendizado de máquina adotado obteve resultados de predição mais eficientes do que os demais modelos de predição na predição de comportamentos de envelhecimento do software. A metodologia experimental e os resultados deste estudo oferecem uma abordagem prática na investigação do envelhecimento de software para plataformas PaaS, juntamente com contramedidas e técnicas de previsão eficientes de envelhecimento de software. Além disso, a avaliação de outras plataformas em conjunto com a plataforma Docker, como Kubernetes ou Swarm, continua sendo uma direção de pesquisa frutífera em trabalhos futuros.

Avaliação de Desempenho de Arquiteturas VANETs com Semáforos Inteligentes Cooperativos

Autores: Firmino Azevedo Neto e Francisco Airton Silva

Resumo

As redes veiculares são formadas por veículos automotores (carros, caminhões, ônibus, ambulâncias e etc). A finalidade de uma rede veicular é permitir a comunicação sem fio entre os veículos automotores. Estas redes emergiram devido ao crescimento das tecnologias de comunicação sem fio. Também faz parte das redes veiculares os equipamentos de beira de estradas conhecidos como *Road Side Unit* (RSU). A adoção de equipamentos RSUs no sistema de semáforos faz com que os sistemas de trânsitos fiquem mais inteligentes. O tráfego inteligente de veículos pode fazer uso de semáforos inteligentes que trabalham cooperativamente para alcançar maior mobilidade de veículos. A arquitetura de semáforos inteligentes oferece desafios relacionados à restrição de tempo de comunicação e quantidade de recursos. Este trabalho tem o objetivo de fornecer uma ferramenta para avaliar o desempenho de uma arquitetura *Vehicle Ad hoc Network* de semáforos inteligentes cooperativos através de modelos de redes de Petri estocásticas (SPN). O foco principal destes modelos é descobrir - através dos resultados obtidos nas simulações - qual a configuração ideal para uma RSU atender a uma determinada quantidade de veículos trafegando em uma via. Os modelos propostos calculam o tempo médio de resposta, utilização de recursos, probabilidade e número de descartes de mensagens. Os modelos são altamente flexíveis, permitindo aos desenvolvedores e administradores de sistemas calibrarem oito parâmetros, incluindo três transições e cinco marcações, representando quantidade de recursos, no modelo base e dezoito parâmetros no modelo estendido, incluindo oito transições e dez marcações. Com o objetivo de validar o modelo e avaliar sua precisão, foram comparados os seus resultados com as métricas coletadas em simulações reais. Os resultados coletados foram condizentes com os resultados fornecidos pelos modelos.

Palavras-chaves: Redes Veiculares, VANET, Redes de Petri Estocásticas, Semáforos Inteligentes, Modelos Estocásticos.

27.1 Introdução

As VANETs (*Vehicular Ad Hoc Networks*) são redes veiculares móveis que emergiram devido ao crescimento das tecnologias de comunicação sem fio. As VANETs foram projetadas para melhorar a eficiência do tráfego de veículos (FERNANDES et al., 2013). Tais redes têm como objetivo permitir a comunicação entre veículos automotores. As características de um ambiente de rede veicular trazem diversos desafios para a indústria e academia, tais como o fato de as VANETs apresentarem conectividade volátil e estarem sujeitas a diferentes densidades veiculares (JÚNIOR; FERREIRA, 2016). No contexto de um ambiente de rede veicular existem duas categorias de comunicação. A primeira, também conhecida como V2V (*Vehicle to Vehicle*), trata da comunicação direta entre veículos. A segunda, também conhecida como V2I (*Vehicle to Infrastructure*), trata da comunicação entre veículos e estações fixas (infraestruturas) localizadas nas margens das vias. Vastas pesquisas vêm sendo desenvolvidas na área de VANETs, no entanto muitos desafios ainda devem ser atacados.

Diversas aplicações podem ser desenvolvidas para VANETs, desde o controle de tráfego, gestão de vagas de estacionamento, alertas de riscos de acidentes, disponibilização de conteúdo interativo aos passageiros, dentre várias outras (SILVA et al., 2013). No contexto específico de controle de tráfego têm-se uma grande preocupação com a alta emissão de CO₂ através de congestionamentos nas grandes cidades. Felizmente, aos poucos o controle de tráfego que conhecemos hoje irá mudar, se tornando cada vez mais inteligente. Tráfego inteligente de veículos é uma área de estudo das redes veiculares que foca em aumentar a mobilidade nas cidades.

Uma das formas de tornar o tráfego mais eficiente é tornando os semáforos mais inteligentes e autônomos para se comunicar com veículos dando instruções de rotas. Outra possibilidade é formar uma cooperação entre semáforos inteligentes a fim de maximizar a quantidade de informações úteis aos veículos na área de cobertura. A arquitetura de semáforos inteligentes cooperativos traz consigo um grande, desafio principalmente relacionado à restrição de tempo de comunicação e quantidade de recursos. Assim, os semáforos inteligentes — acoplados por equipamentos *Road Side Unit* (RSU) — devem ser bem projetados para atender a uma determinada demanda de fluxo de veículos. Avaliar o desempenho de tais sistemas com natureza de tempo-real deve ser uma preocupação desde estágios iniciais de planejamento. No entanto, realizar experimentos reais em VANETs exige equipamentos de alto custo¹.

Redes de Petri são modelos matemáticos baseados em teoria da probabilidade usados para modelar e avaliar sistemas complexos. As características modeláveis de tais sistemas incluem paralelismo, concorrência e parametrização de taxa de entrada de requisições. Modelos de redes de Petri têm sido aplicados para avaliar arquiteturas VANETs (MALIK; PANDEY, 2018; ZHANG et al., 2016; ALHEETI et al., 2016; JUNIOR et al., 2017), porém nenhum desses modelos focou em avaliar o desempenho de arquiteturas VANETs de semáforos inteligentes. Este capítulo apresenta um modelo SPN para modelar uma arquitetura VANET de múltiplos semáforos inteligentes. O modelo proposto permite avaliar a utilização do conjunto de RSUs, o tempo médio de resposta (MRT), o número de descarte e a probabilidade de descarte de mensagens. As principais contribuições deste capítulo são: Fornecer dois modelos analíticos constituídos de métricas de desempenho; que são ferramentas úteis para avaliadores de redes VANETs checarem o desempenho de mudanças no sistema, antes que elas sejam implementadas no contexto de semáforos inteligentes. Uma análise numérica no modelo base, e duas análises numéricas no modelo estendido que servem como um guia prático para análise de desempenho em arquiteturas VANETs.

27.2 Definição do Problema

A avaliação do desempenho de sistemas de semáforo inteligentes é uma tarefa complexa. E muitas das vezes necessita de um ambiente de cenário real para fazer a avaliação. O custo com equipamentos para construir cenários reais para avaliação de infraestruturas VANETs de semáforos inteligentes é muito alto. Uma rede veicular tem como grande característica uma enorme volatilidade dos nós. E essa volatilidade

¹ Por exemplo, um aparelho RSU da marca Cohda custa \$ 4463,11, de acordo com o preço anunciado em [\(https://www.yoycart.com/Product/527650716827/\)](https://www.yoycart.com/Product/527650716827/) na data de escrita deste capítulo

permite que a rede apresente um alto número de nós. É preciso avaliar também o desempenho de sistemas de semáforos inteligentes com uma grande quantidade de nós na rede. E essa quantidade aumenta ainda mais o custo para fazer uma avaliação de desempenho de uma arquitetura VANET de semáforos. Contudo, faz-se necessário um sistema de modelagem que permita aos Engenheiros e Planejadores de arquiteturas VANETs avaliar o desempenho dessas arquiteturas sem precisar fazer investimentos altos com ambientes reais.

27.3 Objetivos

O objetivo geral desta capítulo é avaliar o desempenho de duas arquiteturas VANETs de semáforos inteligentes cooperativos através de modelos de redes de Petri estocásticas (SPN), possibilitando a criação de arquiteturas mais otimizadas nesse contexto.

Os objetivos específicos deste capítulo são:

1. Desenvolver dois modelos de Rede de Petri Estocásticas para analisar o desempenho de arquiteturas VANET com semáforos inteligentes cooperativos;
2. Identificar qual a configuração ideal dos dispositivos de RSUs para cada semáforo inteligente;
3. Mitigar os custos de desenvolvimento com experimentos reais em VANET;
4. Facilitar o entendimento do comportamento de uma rede veicular através de modelagem;

27.4 Organização do Capítulo

O restante deste capítulo se organiza da seguinte forma: A seção 27.5 contém alguns conceitos necessários para o entendimento deste capítulo com informações gerais sobre redes veiculares e redes de Petri Estocásticas; na seção 27.8, apresenta nossa arquitetura e modelo SPN básico com as respectivas métricas e estudos de caso; na seção 27.13, apresenta nossa arquitetura, um modelo SPN estendido também com as respectivas métricas, estudos de caso e acrescentamos uma subseção de validação do modelo estendido; A seção 27.17 discute os principais trabalhos relacionados com esta monografia contendo uma tabela comparativa entre os trabalhos; A seção 27.18 apresenta as principais conclusões sobre a monografia obtidas acerca dos estudos realizados; A seção ?? contém as publicações do autor desta monografia.

27.5 Referencial Teórico

Esta seção apresenta os principais conceitos necessários para melhor acompanhar a proposta deste capítulo. Primeiro, é apresentado um *overview* sobre redes veiculares (mais especificamente as VANETs), em seguida, algumas características das redes de Petri Estocásticas (SPN).

27.6 Redes Veiculares

Uma rede veicular é composta por automóveis, caminhões, ônibus e etc.; pode também serem formadas em qualquer tipo de ambiente. Cada nó da rede deve estar equipado com algum tipo de dispositivo sem fio capaz de comunicar com os dispositivos presentes nos outros veículos e, a partir daí, receber e transmitir pacotes pela rede. As principais características de uma VANET são: a capacidade de comunicação veículo para veículo (V2V), a qual não precisa de um suporte de infraestrutura estática e comunicação veículo para infraestrutura (V2I).

A finalidade de uma VANET é permitir a comunicação sem fio entre veículos automotores e também entre veículos e estações fixas infra-estruturadas situadas nas margens das ruas e estradas (SILVA et al., 2013). As redes veiculares são uma derivação de redes móveis *ad hoc* (MANET, Mobile Ad hoc Network). Estas redes emergiram devido ao crescimento das tecnologias de comunicação sem fio.

Uma VANET possui dois tipos de dispositivos: *OnBoard Unit* (OBU) e RSU. Esses nós se comunicam entre si através de canais sem fio. Cada veículo se define como um nó da rede. Os veículos são equipados com uma unidade de comunicação de bordo chamada OBU. A OBU é um dispositivo de hardware montado no veículo para prover a comunicação com outros nós na rede veicular. Uma OBU pode se conectar tanto com outras OBUs, como para outras RSUs. As RSUs são estações localizadas às margens da rodovia para gerenciar a comunicação na rede. As RSUs servem como nós intermediários para a troca de informações entre veículos e RSUs. O equipamento recebe informações de OBUs, processa localmente e retorna para outras OBUs. As informações trocadas também podem ser compartilhadas com outras RSUs.

O emprego dessas redes permitirá que as informações sobre o tráfego circundante e as condições da estrada sejam transmitidas ao motorista do veículo, permitindo assim uma maior conscientização sobre o ambiente. As possibilidades de informações recebidas variam de informações de congestionamento de tráfego, à frenagem de emergência de veículos imediatamente à frente. As aplicações de detecção podem ser incorporadas para permitir o monitoramento constante e em tempo real do ambiente ao redor das estradas, ou mesmo o monitoramento das condições da estrada. Os dados gravados podem, então, ser transmitidos às respectivas autoridades de transporte da área para correção oportuna, se necessário. A Figura 285 mostra um exemplo de aplicação prática que descreve um acidente e uma possível troca de informações entre vários veículos e RSUs. Na Figura 285, há três RSU informando um acidente ocorrido em uma via para os carros que estão em seu alcance.

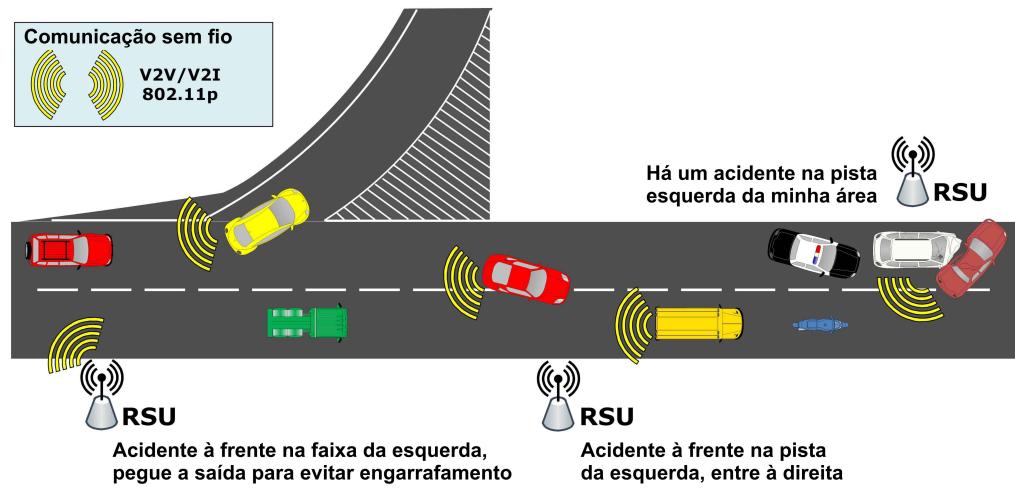


Figura 285 – Exemplo de um cenário VANET com um acidente na via (Fonte: (KULAKARNI; SHARMILA, 2015))

27.6.1 Desafios

As VANETs podem trazer diversos benefícios ao Sistema de Transporte de Terrestre. Há vastas pesquisas produzidas na área, mas ainda há diversos problemas de pesquisas a serem enfrentados. A introdução de mensagens maliciosas em um cenário de rede veicular é um desses problemas. Uma vez que uma mensagem mal intencionada é disseminada no trânsito, os veículos automotores podem receber mensagens falsas. A disseminação dessas mensagens pode acarretar em acidentes e/ou congestionamentos nas estradas. Pode ser citado também a alta mobilidade dos nós na rede. Nenhum outro tipo de rede móvel existente apresenta uma grande quantidade de nós movendo-se a mais de 10 quilômetros por hora. Pois, veículos automotores possuem uma capacidade de alta velocidade ao locomoverem-se. A alta velocidade dos nós exige um sistema de alto desempenho e rapidez de conexão. O tempo de permanência de conexão entre os nós também é um desafio em aberto. Os nós da rede, em certas circunstâncias podem ter um curto espaço de tempo para trocarem informações. A lista, a seguir, cita os principais desafios de uma rede veicular:

- Algoritmos de roteamento

- Endereçamento
- Entretenimento
- Segurança da comunicação
- Mobilidade dos nós

Além dos desafios já citados, este capítulo foca nos problemas com custos, planejamento, análises e ambientes de testes de uma rede veicular. Ambientes VANETs, apesar de muitos anos de pesquisas, ainda são vistos como algo novo. A dificuldade em encontrar ambientes de testes em VANETs é grande. E essa dificuldade faz com que pesquisadores e engenheiros tenham problemas com estudos e planejamento de uma rede veicular. Os custos com ambientes reais de testes também dificultam a vida dos pesquisadores engenheiros.

27.6.2 Melhorias de Segurança no Trânsito

Um dos principais objetivos das redes veiculares é prover mais segurança nos sistemas de trânsitos. Muitos acidentes de trânsito são ocasionados por ultrapassagem indevida, veículos lentos, parados e desgovernados. A troca de informações sobre localização e direção dos veículos pode ajudar a diminuir o número e a gravidade dos acidentes de trânsitos (ALVES et al., 2009). Imagine um motorista com pressa tentando fazer uma ultrapassagem e que o mesmo não tem visão suficiente para enxergar os carros a sua frente. A informação de que um veículo próximo está vindo em direção contrária pode evitar o acidente e até mesmo algum óbito.

Aplicações que fornecem informações sobre acidentes ocorridos em tempo real, contribuem também para um trânsito mais seguro. O recebimento de uma informação, sobre um acidente acontecido próximo na via, pode fazer com que o motorista dirija com mais cautela na estrada e evite uma colisão com um veículo quebrado na estrada.

27.7 Redes de Petri Estocásticas

Segundo (CARDOSO; VALETTE, 1997), as redes de Petri são um modelo matemático com representação gráfica. As redes de Petri vêm sendo amplamente utilizadas a mais de 30 anos, em vários domínios de atuação. A concepção das redes de Petri teve início em 1962, na tese de doutorado de Carl Adam Petri, intitulada *Communication with automata* (do alemão *Kommunikation mit Automaten*), e até os dias de hoje vem sendo utilizada e aprimorada, possuindo inúmeras extensões que as tornam empregáveis em diferentes áreas de conhecimento, como modelagem e avaliação de desempenho, técnicas inteligentes, redes de comunicação, projetos de hardware, etc (TEIXEIRA, 2009). Os domínios que mais se destacam são os de comunicação, de transporte, de informação, logísticos e, de forma geral, todos os sistemas de eventos discretos.

As redes de Petri nos permitem ter uma visão panorâmica do sistema. Com ela, pode-se observar mecanismos de concorrência, assincronicidade, distribuição, determinísticos ou estocásticos. Esses mecanismos são comuns de serem encontrados em sistemas reais. Através da observação destes mecanismos, pode-se avaliar o desempenho de um sistema complexo, tal como uma rede veicular. Em geral, esses sistemas complexos assumem diferentes estados ao longo de seu funcionamento, aos quais são modificados a partir da ocorrência de algum evento.

27.7.1 Elementos de uma Rede de Petri

A representação gráfica das redes de Petri são formadas por: lugares (Figura 286(a)) que correspondem às variáveis de estado; transições (Figura 286(b)) que representam as ações e eventos do sistema; arcos (Figura 286(c)) que presentam os fluxos de *tokens* pelo sistema; e *tokens* (Figura 286(d)), cujo conjunto representa o estado do sistema em um instante. A realização de uma ação ou evento (disparo de uma transição) no sistema está ligado a pré-condições. Uma ligação entre os lugares e a transição deve existir para que esta

possa ou não realizar a ação (disparar). Após o disparo, alguns lugares terão suas informações alteradas (*Tokens*), ou seja, levará a uma pós-condição.

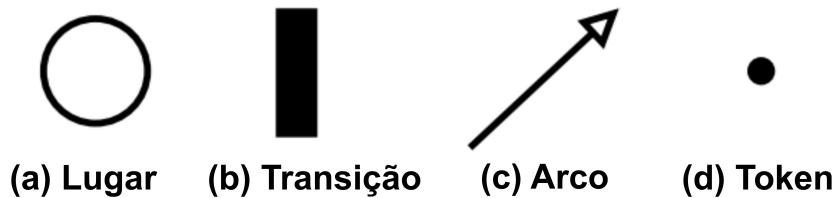


Figura 286 – Elementos de uma Rede de Petri (Fonte: Autor)

Um lugar corresponde a uma variável que armazena estados capturados do sistema modelado. Esses estados são representados por elementos chamados *tokens*. Um estado representa a situação de um sistema em um determinado momento. Um *token* pode ser enfileirado para ser processado, se houver uma posição livre na fila. Se a fila estiver cheia, no entanto, o *token* será descartado.

A marcação de um lugar, ou seja, a definição de seu estado pode ser ou não vazia, sendo possível que um lugar mude de estado, perdendo ou ganhando *tokens*. Essa mudança de estados é proporcionada por um elemento chamado transição (TEIXEIRA, 2009).

Uma transição pode ser entendida como um agente que aciona os estados. Para que uma transição mude o estado de um lugar, é necessário que esses elementos estejam interligados de alguma forma. Em redes de Petri, esse mecanismo de ligação entre lugares e transições é representado por arcos dirigidos que associam os vértices (lugares e transições) as suas regras de pré e pós-condições, que possibilitam a ocorrência de uma ação.

Os arcos podem ser de duas formas: normal ou arco inibidor. No arco comum, a transição de um *token* de um *place* para outro pode ocorrer a qualquer momento, sem restrições. Enquanto que no arco inibidor, sua principal função é impedir a passagem de um *token* para um *place* em um determinado momento.

27.7.2 Redes de Petri Estocásticas

As redes de Petri não foram desenvolvidadas inicialmente com o intuito de prover avaliação de desempenho. As redes de Petri possuem comportamento determinístico e não possuem tempos estocásticos, não sendo o suficiente para a avaliação de sistemas complexos. Esse panorama se modificou em 1982, quando M. K. Molloy (Molloy, 1982) apresentou as redes de Petri estocásticas como uma técnica capaz de, além de especificar sistemas, também apresentar uma análise probabilística dos mesmos (FRANCÉS, 2003). Ele definiu que todas as transições em uma SPN eram temporizadas e que possuíam um retardo exponencialmente distribuído (FRANCÉS, 2003).

A introdução de tempos estocásticos resultou na Rede de Petri Estocástica, que permitiu ligar as redes de Petri a avaliação de desempenho, uma área normalmente baseada na abordagem de modelagem estocástica. Redes de Petri estocásticas são uma forma de rede de Petri onde as transições disparam após um atraso probabilístico determinado por uma variável aleatória.

Existem várias extensões para as redes de Petri. Uma outra extensão, foi a adição de tempos estocásticos combinados com tempos nulos que permitiram modelar sistemas com tempos e condições lógicas. O paradigma resultante foi nomeado de SPNs generalizadas (*Generalized Stochastic Petri Net - GSPN*) (JÚNIOR; FERREIRA, 2016). Esse tipo de formalismo une as redes de Petri temporizadas e não temporizadas (imediatas), atribuindo variáveis aleatórias para a representação do tempo (TEIXEIRA, 2009).

27.8 Proposta Base

O trânsito nas grandes cidades faz as pessoas gastarem bastante tempo devido aos frequentes engarrafamentos. A grande quantidade de semáforos também faz com que os veículos retardem seus traslados. Os semáforos são programados de forma estática obedecendo um tempo baseado em um estudo prévio do fluxo esperado na respectiva via. Espera-se que nas redes veiculares esta realidade torne-se menos problemática através da comunicação entre veículos e semáforos. Considerando que os semáforos passem a ser equipados por dispositivos RSUs no futuro, o objetivo deste capítulo é prover — através de modelos analíticos — uma forma de avaliar tal arquitetura definindo a capacidade da RSU mais adequada à demanda da rede. Em outras palavras, pretende-se descobrir qual a configuração de uma RSU para suportar um número específico de carros solicitando informações de tráfego.

27.9 Arquitetura

A Figura 287 apresenta a arquitetura VANET aplicada neste projeto para analisar o desempenho a partir do nosso modelo SPN. Foi levado em consideração também que, o semáforo está equipado com funcionalidade RSU. A RSU pode se comunicar diretamente com o veículo, se estiver dentro da cobertura da comunicação. Cada veículo está equipado com uma OBU. A OBU transmite periodicamente as informações do veículo incluindo identificação do veículo, posição GPS, velocidade de movimento, direção e status. A RSU coleta o tráfego em tempo real agregando as informações das OBUs, podendo também determinar o tempo de ciclos de fechamentos dos semáforos. A RSU da interseção pode se comunicar com o veículo de todas as direções, enviando pacotes periodicamente. Depois que o veículo entra no alcance da comunicação, ele envia pacotes de dados para a RSU. Então, a RSU processa estes dados para posteriormente retornar os dados para o veículo. Quando os veículos receberem as informações do semáforo, a OBU calculará uma recomendação de velocidade para os motoristas. De acordo com a velocidade recomendada, o motorista pode mudar ou manter a velocidade atual para passar através dos cruzamentos com menor tempo de espera ou menos tempos de parada. Dada esta visão geral, o objetivo deste trabalho é responder à seguinte pergunta: **Qual a configuração mais adequada do aparelho RSU para atender a uma determinada demanda de mensagens?** Como mostra a Figura 287, os recursos atrelados a tal configuração referem-se à capacidade do *buffer* de entrada e saída da RSU e da sua capacidade de processamento paralelo.

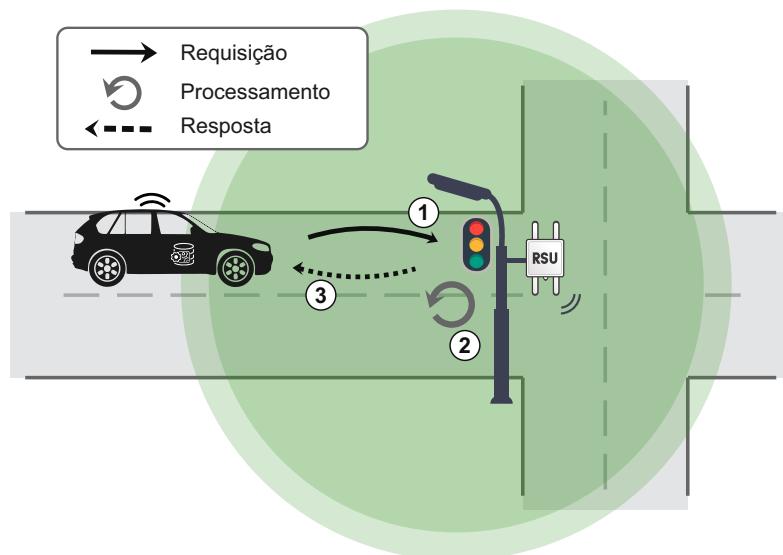


Figura 287 – Ilustração da arquitetura base a ser avaliada (Fonte: Autor)

É importante ressaltar que, diferente do modelo proposto em (JUNIOR et al., 2017), este capítulo considera uma arquitetura onde a RSU processa dados provenientes dos veículos e remete esses dados para

veículos que estão em seu alcance. Esse é um aspecto de suma relevância para o setor de transportes inteligentes, pois a RSU tem um papel fundamental nessa arquitetura, que é intermediar troca de informações entre os veículos. o desempenho da infraestrutura V2I de uma VANET é avaliada de uma forma mais realística para informar aos motoristas as condições do trânsito, além de prover recursos de mídia, controle de congestionamento, alertas de acidentes, dentre outros.

27.10 Modelo SPN Base

O foco do nosso modelo é auxiliar engenheiros e analistas a avaliarem com rapidez o desempenho de uma VANET, em particular a comunicação V2I. Mais especificamente, o modelo visa estimar tempos de respostas e probabilidade de falhas da comunicação V2I, dado parâmetros de variações de tráfego de veículos, assim como variações de configurações dos equipamentos (OBU e RSU) que compõem a arquitetura apresentada na seção anterior. Tal modelo é, portanto, uma ferramenta útil para o planejamento e ajustes da infraestrutura de VANETs.

Nesse sentido, o modelo SPN proposto neste trabalho é apresentado na Figura 288. Esse modelo está organizado em duas partes principais: a sub-rede *Admission* representa a geração das mensagens, ao passo que a sub-rede RSU representa a comunicação e distribuição das informações entre veículos.

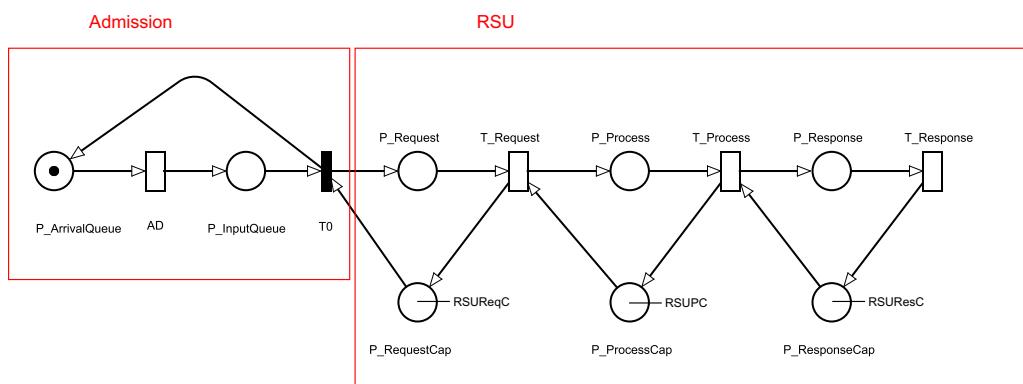


Figura 288 – Modelo SPN para a arquitetura VANET (Fonte: Autor)

Dada a visão geral do modelo, é descrito agora os detalhes do seu funcionamento. A sub-rede *Admission* é composta por dois lugares: *P_Arrival* e *P_InputQueue*. *P_Arrival* representa o estado do sistema esperando novas mensagens serem geradas. *P_InputQueue* representa a aceitação dessas mensagens na fila de entrada no sistema. Os *tokens* em *P_Arrival* e *P_InputQueue* representam quaisquer tipos de mensagens que envolvam entrada de dados e que possam ser distribuídos. Os tempos entre chegadas de trabalhos são atribuídos à transição *AD* (*arrival delay*). Para melhor entendimento do *AD*, a Figura 27.10 mostra que quando os valores de *AD* são maiores, isto significa que haverá menos carros requisitando informações por unidade de tempo. Apesar de que os tempos entre disparos foram considerados exponencialmente distribuídos, tal suposição pode ser modificada, alterando a distribuição. A sub-rede *Admission* também não considera o atraso proveniente da transmissão do cliente para o sistema. A transição *AD* leva em conta apenas o tempo que as mensagens entraram no sistema, ou seja, não são levadas em conta as perdas provenientes da rede. A transição imediata *T0* representa simplesmente a entrada das mensagens no sistema. Como esta transição é imediata, não apresenta atraso associado à sua execução.

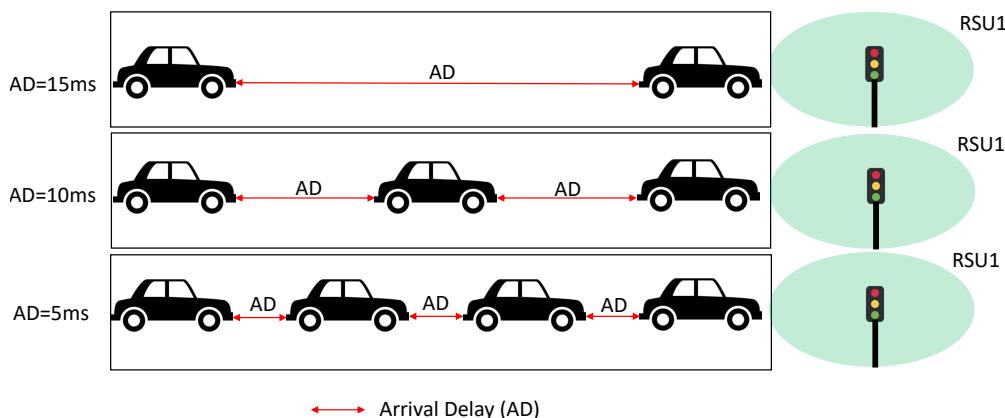


Figura 289 – Ilustração da variação do AD (Fonte: Autor)

Tabela 183 – Descrição dos Elementos do Modelo Base

Tipo	Nome do elemento	Descrição
Lugares	P_Arrival	Espera por novas mensagens
	P_InputQueue	Espera pela disponibilidade da fila
	P_Request	Mensagem na transmissão para RSU
	P_Process	Processamento na RSU
	P_Response	Mensagem na etapa de resposta
	P_RequestCap	Capacidade de transmissão
	P_ProcessCap	Capacidade de processamento da RSU
	P_ResponseCap	Capacidade de resposta
Transições Temporizadas	AD	Tempo entre chegadas de mensagens
	T_Request	Tempo na transmissão para RSU
	T_Process	Tempo de processamento na RSU
	T_Response	Tempo na resposta das mensagens
Marcações de lugares	RSUReqC	Número de canais disponíveis para transmissão
	RSUPC	Número de recursos disponíveis para processamento
	RSUResC	Número de canais disponíveis para transmissão

Fonte: Autor

A sub-rede RSU é composta por três etapas: transmissão, processamento e resposta. Nas três etapas existe um tempo associado. Na primeira etapa, a mensagem é transmitida para a RSU. Na segunda etapa, a mensagem é processada pela RSU. Na terceira etapa a mensagem é respondida para as OBUS. Quando T0 dispara, a sub-rede RSU é alcançada, um *token* é retirado de P_InputQueue e P_RequestCap. A quantidade de *tokens* em P_Request representa o enfileiramento das mensagens a serem enviadas para a RSU. O enfileiramento ocorre quando não há capacidade disponível para enviar a mensagem recém-chegada. Em P_Request a mensagem espera um tempo associado à transição T_Request para a mensagem passar para a segunda etapa. A segunda e terceira etapas seguem a mesma ideia desta primeira etapa. As marcações RSUReqC, RSUPC, e RSUResC representam a quantidade de recursos disponíveis para transmitir, processar e responder mensagens, respectivamente. A Tabela 183 apresenta uma descrição dos elementos presentes no modelo base.

27.11 Métricas de Desempenho

O modelo SPN apresentado na seção anterior, foi utilizado para obter as métricas de interesse para a avaliação de desempenho. Especificamente, foi calculado o tempo médio de resposta (ou comunicação V2I), erros de comunicação de dado pela probabilidade/número de descartes de mensagens de veículos ao sistema e, por fim, o nível de utilização da RSU. A seguir, foi descrito como foi obtido cada uma dessas métricas com o modelo SPN.

O tempo médio de resposta (MRT) pode ser obtido a partir da Lei de Little ([LITTLE, 1961](#)). A Lei de Little relaciona o número médio de requisições em progresso em um sistema (*RequestsInProgress*), a taxa de chegada (*ArrivalRate*) e o tempo médio de resposta (MRT). A taxa de chegada é o inverso do tempo de

chegada. Considerando a transição para tempo entre chegadas do modelo, temos que $ArrivalRate = \frac{1}{AD}$. Vale ressaltar que a Lei de Little requer um sistema estável, ou seja, que possua uma taxa de requisições menor que a taxa de processamento dos servidores. A equação para calcular MRT é portanto: $MRT = (Esp(P_Request) + Esp(P_Process) + Esp(P_Response)) \times AD$. $Esp(Lugar)$ indica quantos *tokens* ocupam aquele Lugar. $P(Lugar = n)$ calcula a probabilidade de existirem n *tokens* em “Lugar”.

No modelo é possível ter noções sobre o nível de utilização do sistema, o que permite saber qual a configuração correta dos equipamentos da arquitetura. Com essa noção, é possível saber, sem gastar recursos reais, se o valor que o projetista pretende investir em recursos está acima ou abaixo da quantidade necessária. Imagine que um projetista esteja, por exemplo, planejando uma infraestrutura VANET e precisa saber o nível de utilização. O projetista sabe a média de veículos que trafegam na via e quer que a infraestrutura apresente 50% de sua capacidade. O projetista configura os parâmetros no modelo e nossa modelagem vai informar a utilização que o sistema apresentará. A equação para utilização da RSU é: $URSU = \frac{Esp(P_Process)}{RSUC} \times 100$. A variável RSUC representa a quantidade de recursos disponíveis na RSU.

Em uma rede veicular, um ou vários descartes no sistema pode fazer com que o veículo saia do alcance de uma RSU sem obter informações sobre a via. Com a modelagem proposta, é possível saber qual a porcentagem e qual o número de mensagens descartadas no sistema. Para calcular o descarte é necessário existir *token* na fila de entrada P_Arrival e não restar mais nenhum recurso disponível dentro dos nós. A equação que define a probabilidade de descarte é dada por: $Discard = P((P_Input_Queue = 1) \wedge (P_Request = 0) \wedge (P_Process = 0) \wedge P_Response = 0)$. A Equação $DiscardNumber = P((P_Input_Queue = 1) \wedge (P_Request = 0) \wedge (P_Process = 0) \wedge (P_Response = 0)) \times (1/AD)$ define o número de mensagens descartadas em um determinado período de tempo T.

27.12 Estudos de caso

Nesta seção, a utilização do nosso modelo foi demonstrada a partir de um cenário prático onde foi avaliado o desempenho de uma determinada configuração da arquitetura, variando a carga de trabalho do sistema, ou seja, crescimento do tráfego de veículos na área de cobertura da infraestrutura V2I.

uma RSU instalada em uma via urbana típica onde o tráfego de veículos varia ao longo do dia, foi levada e consideração. Assim, existem diferentes tipos de tráfego por período de tempo, indo de leve (tráfego nas madrugadas) a pesado (tráfego no fim do expediente comercial: 17 às 18hs). A RSU também segue esse padrão e está habilitada para se conectar a cada um dos veículos em sua área de cobertura. No entanto, a RSU possui limitações de recursos, incluindo um número limitado de canais de requisição, resposta e threads de processamento de mensagens.

Para representar um tipo de tráfego em um intervalo de tempo, intervalos entre chegadas de mensagens na RSU foram assumidos seguindo distribuição exponencial, então foi variado o parâmetro dessa distribuição para obter o crescimento dos tempos entre chegadas, ou seja, tráfego pesado para leve. A distribuição exponencial também é utilizada para as outras três variáveis do modelo, que representam transições temporizadas, contudo para uma parametrização fixa como mostra a Tabela 184. Finalmente, todos os valores utilizados para os demais parâmetros do modelo, que representam configurações da arquitetura, são definidos na Tabela 185.

Tabela 184 – Valores Atribuídos para as Transições Temporizadas do Modelo

Transições Temporizadas	Tempo (ms)
AD	[1,0-19,0] (Acréscimo de 1)
T_Request	3,5
T_Process	10,5
T_Response	2,5

Fonte: Autor

Dadas essas definições, quatro análises numéricas foram apresentadas, a partir da resolução do modelo SPN para exemplificar como um projetista de uma VANET pode utilizar o modelo para dimensionar adequadamente a configuração dessa rede. A Figura 290(a) apresenta os resultados para o MRT. A princípio, espera-se que quanto maior o intervalo de tempo entre chegadas (AD), menor seja o valor de MRT, e o

Tabela 185 – Valores Atribuídos para os Recursos do Modelo

Marcações dos Lugares	Recursos
RSUReqC	5
RSUPC	5
RSUResC	5

Fonte: Autor

sistema estará mais ocioso com menos chegadas de mensagens. Como o sistema estará mais ocioso, ele apresentará um nível menor de utilização. Espera-se também que quanto maior for a quantidade de canais, representado pela variável RSUC (diferente da proposta base, no modelo estendido RSUC representa a quantidade de canais), menor será o valor de MRT, pois aumentará a disponibilidade de recursos, aumentando o nível de paralelismo. Estes dois comportamentos são observados nos resultados da análise numérica. Quando os valores de descartes de requisições são desprezíveis no modelo, para estes intervalos, o MRT decresce até o tempo mínimo para executar trabalhos sem enfileiramento no sistema.

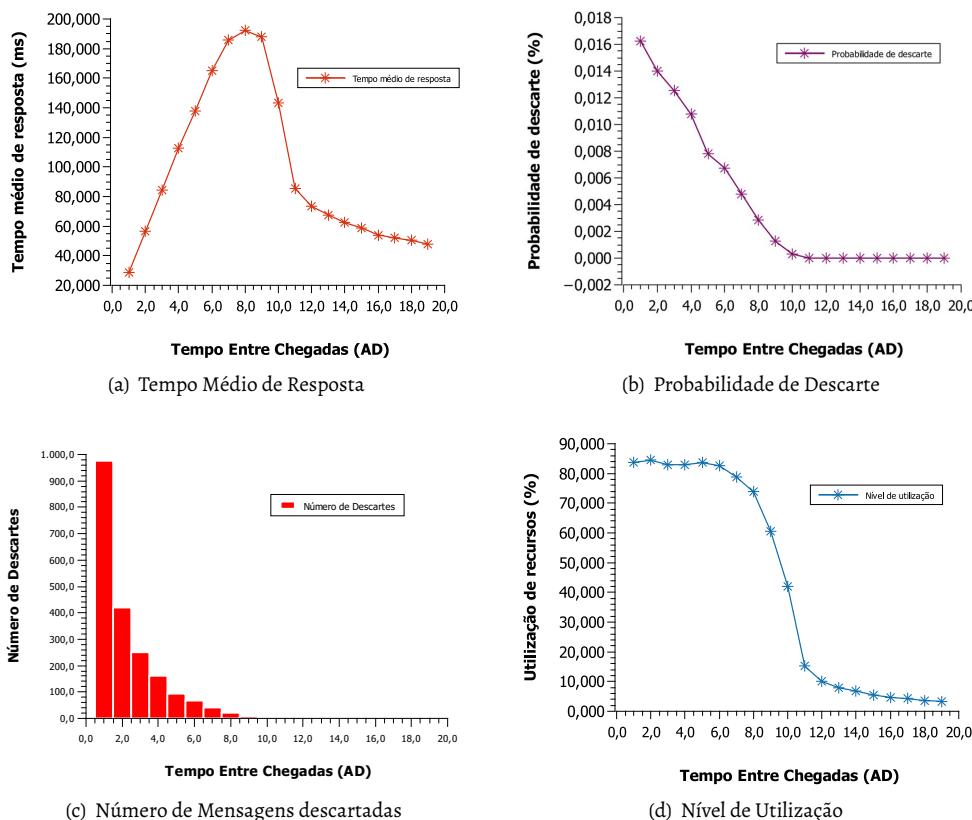


Figura 290 – Resultados das Análises Numéricas (Fonte: Autor)

A Figura 290(b) apresenta a probabilidade de descarte de novas mensagens. Foi realizada uma análise numérica, variando o tempo entre chegadas (AD) de 1.0ms a 19.0ms, levando em consideração uma RSU com capacidade de 5 threads de processamento, 5 canais de entrada e 5 canais de resposta. Conforme o tempo de chegada das mensagens aumenta, a probabilidade de haver descarte no dispositivo diminui. A partir do AD = 10.0, as chances de haver descarte é zero. Além disso, a redução da probabilidade de descarte impacta diretamente no tempo médio de resposta do sistema, pois, a partir do ponto em que não há mais chances de descarte, o tempo médio de resposta começa a decrementar.

A Figura 290(c) apresenta o número de descartes que ocorreram no sistema em um determinado período de tempo T. Pode-se observar que quanto maior o tempo entre chegadas, menor é o número de mensagens descartadas. Quando o tempo entre chegadas é maior que 9.0ms, o número de mensagens descartadas é 0. Quando o tempo é inferior a 9.0ms, o sistema começa a apresentar mensagens descartadas.

A Figura 290(d) apresenta o nível de utilização dos recursos de processamento da RSU. Como pode ser observado, quando o tempo entre chegadas aumenta para 6.0ms, o nível de utilização tende a cair. A queda é devido ao sistema começar a se recuperar de um alto número de requisições, pois, nos testes o número de requisições vai decrementando. Para os casos em que o tempo entre chegadas é menor que 6ms, pode-se notar que a RSU apresenta níveis altos de utilização, com casos em que chega próximo aos 85% de utilização.

27.13 Proposta Estendida

27.13.1 Arquitetura Avaliada

Esta seção apresenta uma arquitetura VANET com múltiplos semáforos que trabalham de forma cooperativa para prover informações de tráfego aos veículos na via. A Figura 291 ilustra tal arquitetura. Assume-se que todos os semáforos atuam como RSUs e cada veículo está equipado com uma OBU. As RSUs possuem capacidade de processamento e armazenamento limitadas. As OBUs possuem a capacidade de localização GPS. As RSUs estão conectadas, a princípio, por um cabeamento físico óptico, porém podem estar conectadas, por exemplo, por conexão wireless como LoRa, 5G C-V2X ou IEEE 802.11p². Vale ressaltar que a escolha destas tecnologias impacta diretamente no tempo médio de resposta. É importante notar também que nossa arquitetura e subsequente modelo SPN apresentam apenas quatro semáforos. No entanto, a proposta se adequa a qualquer quantidade de semáforos. Considere que cada um dos quatro semáforos/RSUs³ possuem uma área de cobertura que está constantemente disponível esperando requisições dos veículos que entram no alcance.

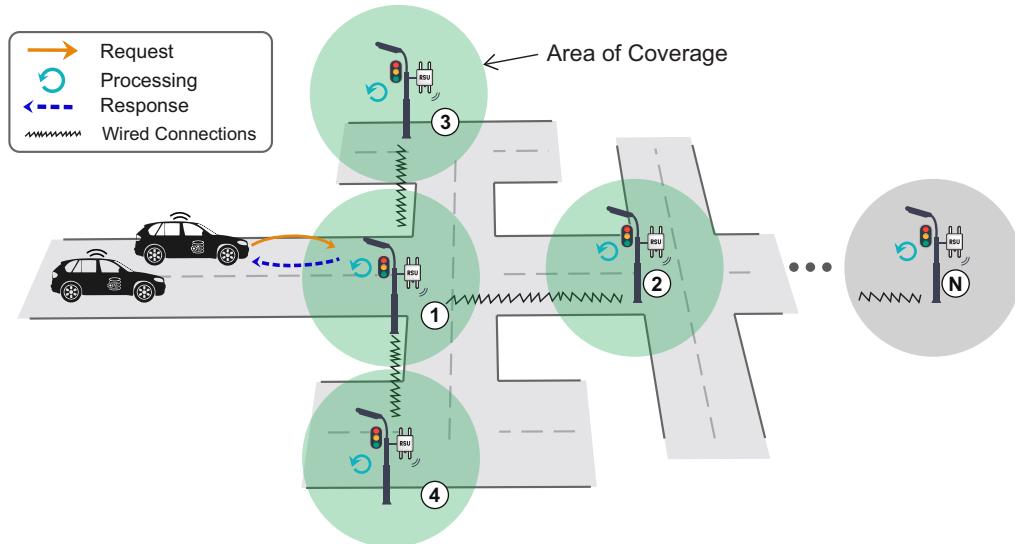


Figura 291 – Arquitetura VANET com Múltiplos Semáforos Cooperativos (Fonte: Autor)

Com relação às etapas de comunicação, quando o veículo entra no alcance da RSU1, ele requisita informações do tráfego daquela região. A chegada de novas requisições dependerá da quantidade e velocidade dos veículos na via, obedecendo assim a uma taxa de chegada que está normalmente alinhada com o período do dia. Horários de pico possuem maior tráfego. A requisição do veículo à RSU contém dados do próprio veículo, tais como velocidade, condições do veículo, rota previamente traçada, etc. A RSU1 por sua vez solicita informações sobre as regiões das três outras RSUs. Por fim, a RSU1 processa e agrupa tais informações e retorna uma resposta ao veículo requisitante. Quando os veículos receberem as informações dos semáforos, a OBU calculará uma velocidade recomendada para o motorista. De acordo com a velocidade

² LoRa é uma tecnologia de rede de área ampla de baixa potência. 5G C-V2X servirá como a base para os veículos se comunicarem uns com os outros e com tudo ao seu redor. IEEE 802.11p é uma emenda aprovada ao padrão IEEE 802.11 para adicionar acesso sem fio em ambientes veiculares.

³ A partir daqui, assumiremos que semáforo e RSU são sinônimos, já que estes são integrados.

recomendada, o motorista pode alterar ou manter a velocidade atual para atravessar os cruzamentos com menor tempo de espera ou menos tempo de parada. Ao receber informações do semáforo a OBU também será responsável por alertar ao motorista sobre possíveis acidentes e as melhores rotas a serem utilizadas com base no trânsito.

Dada esta arquitetura complexa, o desafio existente é planejar a configuração física dos elementos do sistema. A pergunta de pesquisa é, portanto: **Qual a capacidade das RSUs em termos de processamento e comunicação paralela que atendam a uma determinada taxa de chegada de veículos observando o tempo médio de resposta, a utilização de recursos e descarte de pacotes?** Obviamente tal resposta dependerá do tipo de serviço que as RSUs irão prover, bem como a quantidade de dados trafegados na rede. A próxima seção apresenta um modelo SPN que considera todas estas questões.

27.14 Modelo SPN Estendido

A Figura 292 apresenta o modelo SPN para a arquitetura anteriormente detalhada. O foco do modelo é auxiliar engenheiros e analistas a avaliarem o desempenho de uma arquitetura VANET V2I com múltiplos semáforos. O modelo está organizado em seis partes principais: Admission, RSU1_C, RSU2, RSU3, RSU4 e RSU1_P. A sub-rede Admission representa a geração das mensagens. A sub-rede RSU1_C representa a comunicação entre a RSU1 e as demais RSUs considerando envio e retorno da requisição. As sub-redes RSU2, RSU3, e RSU4 representam o processamento e a distribuição de informações dos semáforos numerados por 2, 3 e 4 respectivamente. A sub-rede RSU1_P representa o processamento dos dados advindos dos outros três semáforos que estão instalados próximos ao semáforo principal. Na parte inferior do modelo ressalta-se a possibilidade de adicionar N RSUs simplesmente incluindo uma nova sub-rede.

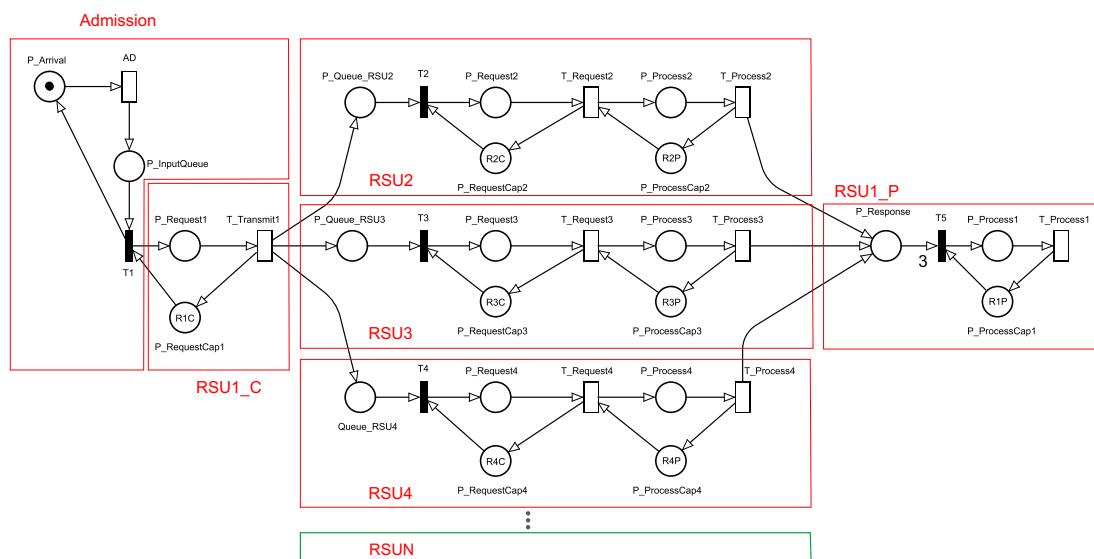


Figura 292 – Modelo SPN para a arquitetura VANET com semáforos cooperativos (Fonte: Autor)

Dada a visão geral do modelo, os detalhes do seu funcionamento são descritos a seguir. A sub-rede Admission é composta por dois lugares: P_Arrival e P_InputQueue. P_Arrival representa o estado do sistema esperando novas mensagens serem geradas. P_InputQueue representa a aceitação dessas mensagens na fila de entrada no sistema. Os tokens em P_Arrival e P_InputQueue representam quaisquer tipos de mensagens que envolvam entrada de dados e que possam ser distribuídos. Os tempos entre chegadas de trabalhos são atribuídos à transição AD (arrival delay). Apesar de que os tempos entre disparos foram considerados exponencialmente distribuídos, tal suposição pode ser modificada, alterando a distribuição. A sub-rede Admission também não considera o atraso proveniente do sistema, ou seja, não

são levadas em conta as perdas provenientes da rede. A transição imediata T0 representa simplesmente a entrada de mensagens no sistema. Como esta transição é imediata, não apresenta atraso associado à sua execução.

A sub-rede RSU1_C é composta por uma transição temporizada e dois lugares: P_Request1 e P_RequestCap1. P_Request1 é alcançado quando T1 dispara. P_RequestCap1 representa a capacidade de envio e recebimento de requisições para os outros semáforos. Quando T0 dispara, um *token* é retirado de P_Request1 e enviado para os blocos que representam as três RSUs. As sub-redes RSU2, RSU3 e RSU4 são compostas por três etapas: filas de chegada, transmissão e processamento. As três RSUs são acionadas de forma paralela através da transição T_Transmit1. A transição T_Transmit1 representa o tempo para uma requisição sair da RSU1 e retornar das três RSUs sem incluir os tempos de transmissão para as mesmas. Como as três sub-redes possuem o mesmo funcionamento, a explicação seguinte focará apenas na RSU2.

Quando o *token* chega no lugar P_QueueRSU2 indica que a requisição já saiu da RSU1. A quantidade de *tokens* acumulados em P_QueueRSU2 significa que existe um enfileiramento sob os recursos de transmissão para a RSU2. Tais recursos de transmissão são limitados pela marcação R2C no lugar P_RequestCap2. O *token* localizado no lugar P_Request2 representa que a RSU alocou um recurso de transmissão para a requisição em questão. A seguir, esta requisição levará um tempo de transmissão para poder chegar à próxima etapa de processamento. Este tempo de transmissão é representado pela transição T_Request2. A etapa de processamento é regida pela capacidade de processamento da RSU, através da marcação R2P em P_ProcessCap2. O tempo associado ao processamento de uma requisição é dado pela transição T_Process2.

Quando as três RSUs finalizam seus respectivos processamentos o lugar P_Response conterá 3 *tokens*. Somente com esta condição é que a transição T5 poderá ser acionada — observe o peso "3" no respectivo arco. A sub-rede RSU1_P representa o processamento das informações advindas dos três semáforos. A ideia é sintetizar os dados das quatro RSUs e retornar as informações ao veículo que realizou a requisição. P_ProcessCap1 indica a capacidade de processamento da RSU1. T_Process1 trata do tempo para processar e agrupar as informações das quatro RSUs.

27.14.1 Métricas

Esta seção apresenta as métricas de interesse obtidas a partir da resolução do modelo. Especificamente, foi calculado o tempo médio de resposta, erros de comunicação de dados pela probabilidade/número de descartes de mensagens de veículos ao sistema, e por fim, o nível de utilização das RSU somadas. A seguir, estão as descrições de como foi obtido cada uma dessas métricas com o modelo SPN.

Uma grande característica das redes veiculares é a mobilidade e alta velocidade dos nós. Um veículo pode entrar e sair do alcance de uma RSU em um curto intervalo de tempo. Nesse contexto, é necessário ter noções sobre o tempo médio de resposta e que esse tempo médio de resposta seja baixo para que uma VANET apresente desempenho satisfatório. A explicação do funcionamento do MRT foi explicado anteriormente na seção 27.11. Para obter o MRT basta multiplicar o número de requisições dentro do sistema pelo tempo entre chegadas. O número de requisições é dado pela soma dos valores esperados em cada Lugar por onde passa o fluxo de requisições no sistema. A expressão *Esp(Lugar)* indica quantos *tokens* ocupam aquele Lugar. Assim, a Equação 27.1 apresenta a forma de calcular o MRT:

$$\begin{aligned} \text{MRT} = & ((\text{Esp}(P_Queue_RSU2)) + (\text{Esp}(P_Queue_RSU3)) + (\text{Esp}(P_Queue_RSU4))) + \\ & (\text{Esp}(P_Request2)) + (\text{Esp}(P_Request3)) + (\text{Esp}(P_Request4)) + \\ & (\text{Esp}(P_Process2)) + (\text{Esp}(P_Process3)) + (\text{Esp}(P_Process4)) + \\ & (\text{Esp}(P_Response)) + (\text{Esp}(P_Process1)) + (\text{Esp}(P_Request1))) \times AD \end{aligned} \quad (27.1)$$

No modelo é possível ter noções sobre o nível de utilização do sistema, o que permite planejar qual a configuração correta dos equipamentos da arquitetura. Com essa noção, é possível saber, sem gastar com recursos reais, se o valor que o projetista pretende investir está acima ou abaixo da quantidade necessária. Imagine que o projetista sabe a média de veículos que trafegam na via e quer que a infraestrutura consuma

apenas 50% de sua capacidade para evitar sobrecarga. O projetista configura os parâmetros no modelo e pode obter a utilização que o sistema apresentará. A utilização é calculada para cada tipo de recurso das diversas RSUs. Os recursos podem ser de comunicação (Equação 27.2) ou processamento (Equação 27.3). Em ambos os casos basta dividir a esperança de existir *tokens* em um "lugar" dividido pela quantidade de recursos disponíveis na respectiva RSU. A letra N representa o índice da RSU, "c" representa comunicação e "p" processamento.

$$\text{UNC} = \frac{\text{Esp}(P_RequestN)}{RNC} \times 100 \quad (27.2)$$

$$\text{UNP} = \frac{\text{Esp}(P_ProcessN)}{RNP} \times 100 \quad (27.3)$$

Em uma rede veicular, um ou vários descartes no sistema pode fazer com que o veículo saia do alcance de uma RSU sem obter informações sobre a estrada. Com a modelagem proposta, é possível saber qual a porcentagem e qual o número de mensagens descartadas no sistema. Para calcular o descarte é necessário existir *token* na fila de entrada P_Arrival e ter probabilidade zero de existir recursos suficientes para atender às requisições na entrada do sistema. Relaciona-se a entrada do sistema à RSU1 no modelo proposto. Portanto, a equação que define a probabilidade de descarte é dada pela Equação 27.4, e o número de descartes é dada pela Equação 27.5.

$$\text{DiscardProbability} = P((P_InputQueue = 1) \wedge (P_RequestCap1 = 0)) \times 100 \quad (27.4)$$

$$\text{DiscardNumber} = \text{DiscardProbability} \times (1/AD) \quad (27.5)$$

27.15 Estudos de Caso

Esta seção apresenta duas análises numéricas de desempenho da arquitetura VANET proposta neste trabalho. Em ambas análises, foi considerado que, o tempo médio de resposta possui uma natureza crítica, normalmente inerente a contextos que a segurança dos passageiros é muito importante e toda comunicação da rede deve ocorrer praticamente em tempo real. Por conta disso as RSUs foram simuladas com altas capacidades de processamento e comunicação. O primeiro estudo analisa uma variação da taxa de chegada de veículos no campo de cobertura da RSU1. O segundo estudo analisa a possibilidade de utilizar três tipos de tecnologias de comunicação. Tanto a modelagem da rede de Petri, quanto as análises foram executadas na ferramenta *Mercury*. O *Mercury* possui recursos avançados de modelagem tanto para modelos de desempenho quanto de disponibilidade, confiabilidade, dentre outros (MACIEL et al., 2017).

27.15.1 Variação do Tempo entre Chegada de Veículos

O efeito da variação do tempo entre chegadas (*arrival delay*) dos veículos nas métricas de desempenho da rede apresentadas na Seção 27.14.1 é analisado agora. Observe que esse tempo se refere ao momento em que o veículo atinge a área de cobertura da RSU1 e automaticamente solicita as informações de tráfego da região. A RSU1, por sua vez, realiza todo o processo de comunicação com as RSUs secundárias para agrupamento de informações. Para melhor entendimento do cenário, a Figura 27.10 mostra que quando os valores de AD são maiores, isto significa que haverá menos carros requisitando informações por unidade de tempo.

Foram atribuídos valores hipotéticos tanto para as transições quanto para os lugares. Os valores utilizados para as transições foram: AD variando de 1,0 até 10,0ms com incrementos de 1,0ms; o tempo de 40,0ms atribuído aos tempos de processamentos incluindo T_Process1, T_Process2, T_Process3 e T_Process4; o tempo de 5,0ms atribuído aos tempos de transmissão incluindo T_Request1, T_Request2, T_Request3 e T_Response4. Em termos de quantidade de recursos atribuiu-se o valor 100 para marcações de comunicação (R1C, R2C, R3C e R4C) e o valor 200 para marcações de processamento (R1P, R2P, R3P e R4P). Esses valores foram utilizados para gerar análises, onde os resultados podem ser observados na Figura 27.15.1.

A Figura 293(a) apresenta os resultados para o MRT. Dois comportamentos são observados nos resultados da análise numérica, separados por uma curva no gráfico. Na primeira metade, com o aumento de AD o sistema se mantém com carga máxima de requisições, ou seja, o número de requisições dentro do sistema

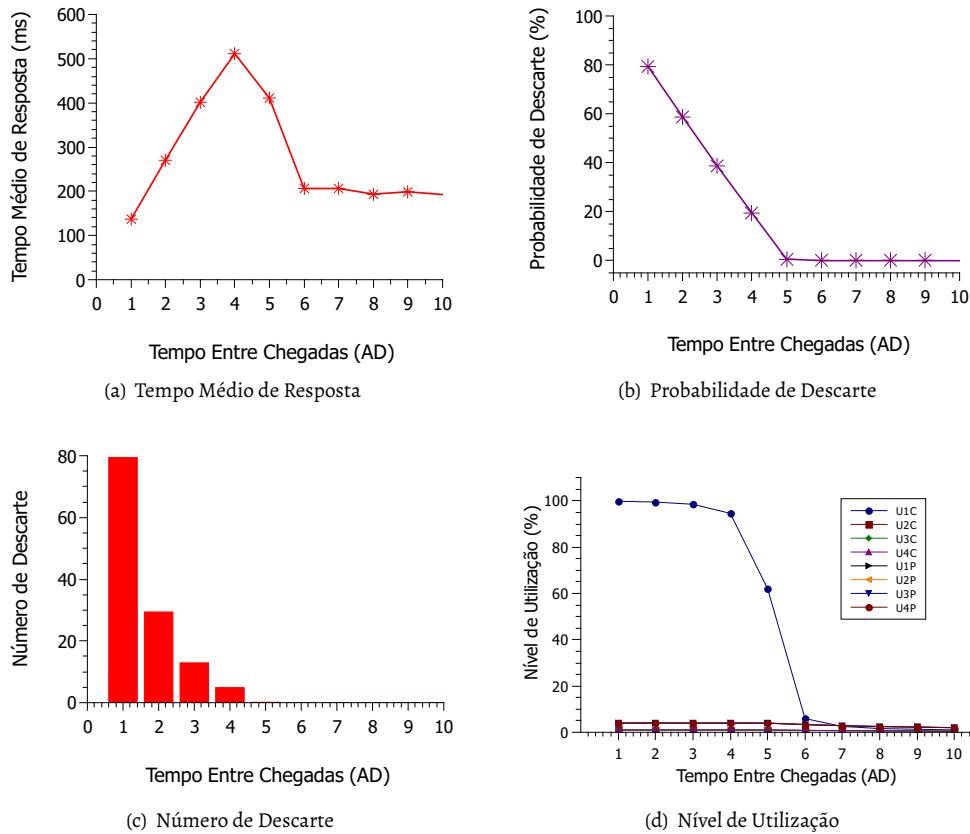


Figura 293 – Resultados da Análise Numérica Variando Tempo entre Chegadas (Fonte: Autor)

se mantém praticamente constante. Dessa forma, o MRT então aumenta proporcionalmente ao valor de AD. Após o pico da curva ($AD = 4,0\text{ms}$), a quantidade de elementos no sistema começa a cair numa proporção maior do que o aumento de AD, fazendo com que o MRT decaia também até estagnar em $AD = 6,0\text{ms}$. A Figura 293(b) apresenta a probabilidade de descarte. Conforme AD aumenta, a probabilidade de haver descarte diminui, pois terão menos requisições a serem atendidas. A partir do $AD = 5,0\text{ms}$ as chances de haver descarte é zero.

A Figura 293(c) apresenta o número de descartes de requisições que ocorreram no sistema em um determinado período de tempo t . Pode-se observar que quanto maior o AD, menor é o número de mensagens descartadas. Quando AD é maior que $5,0\text{ms}$, o número de mensagens descartadas é zero. Quando AD é baixo ($AD=1,0\text{ms}$ por exemplo), o número de descartes é de 80 descartes. A Figura 293(d) apresenta o nível de utilização dos recursos da arquitetura VANET. São quatro conjuntos de recursos para comunicação (U1C, U2C, U3C, U4C) e quatro conjuntos de recursos para processamento (U1P, U2P, U3P, U4P). Ocorreram altíssimos níveis de utilização apenas no recurso de comunicação da RSU1 pois é o primeiro componente alcançado pelas requisições, nos demais recursos a utilização foi sempre próxima de zero. A baixa utilização nos demais recursos foi causada pelo gargalo ocorrido na RSU1. Para contornar este problema seria sugerível aumentar a quantidade de recursos de comunicação da RSU1. A partir de $AD=6,0\text{ms}$ o gargalo deixa de existir, porém as utilizações dos demais recursos não se elevam, pois a quantidade de requisições se torna baixa suficiente para os recursos suportarem.

27.15.2 Variação do Tempo de Comunicação entre RSUs

Nesta seção, o impacto da tecnologia de comunicação usada para a conexão entre as RSUs foi analisado. Em (DJILALI et al., 2018) foi apresentado as principais indicadores chave de performance para três tecnologias de comunicação wireless em redes VANETs, como pode ser visto na Tabela 186. Observando o critério

Velocidade de Transmissão, pode-se notar que 5G ($\cong 28\text{GHz}$) é mais rápida do que a tecnologia IEEE 802.11p ($\cong 5.9\text{GHz}$), que por sua vez é mais rápida do que a tecnologia LoRa ($\cong 868\text{MHz}$).

Tabela 186 – Indicadores Chave de Performance (KPIs) para Lora, 5G C-V2X, IEEE 802.11P (DJILALI et al., 2018)

KPIs	LoRA	IEEE 802.11P	5G C-V2X
Grau de Cobertura	+++	+	+++
Confiabilidade	+	+++	+++
Escalabilidade de Rede	+	+++	+++
Velocidade de Transmissão	+	++	++
Segurança	+++	+++	+++
Supporte a Mobilidade	+	+++	+++

Agora, diferentes configurações foram analisadas para uma arquitetura de semáforos inteligentes cooperativos, comparando as três tecnologias mencionadas para conexão entre as RSUs, assim como desenvolvedores e administradores tipicamente analisam o desempenho de VANETs. Nesse sentido, foi considerado que, os tempos de comunicação atribuídos às transições do tipo T_RequestN foram configuradas com três tempos: 2,5ms para 5G C-V2X, 5,0ms para IEEE 802.11p e 7,5ms para LoRa. Os valores utilizados para as demais transições foram: AD = 5,0ms, e o tempo de 20,0ms atribuído às transições de processamento, ou seja, transições do tipo T_ProcessN; Em termos de quantidade de recursos, atribuiu-se o valor 100 para marcações de comunicação (R1C, R2C, R3C e R4C) e o valor 200 para marcações de processamento (R1P, R2P, R3P e R4P). Os resultados desta análise podem ser observados na Figura 294.

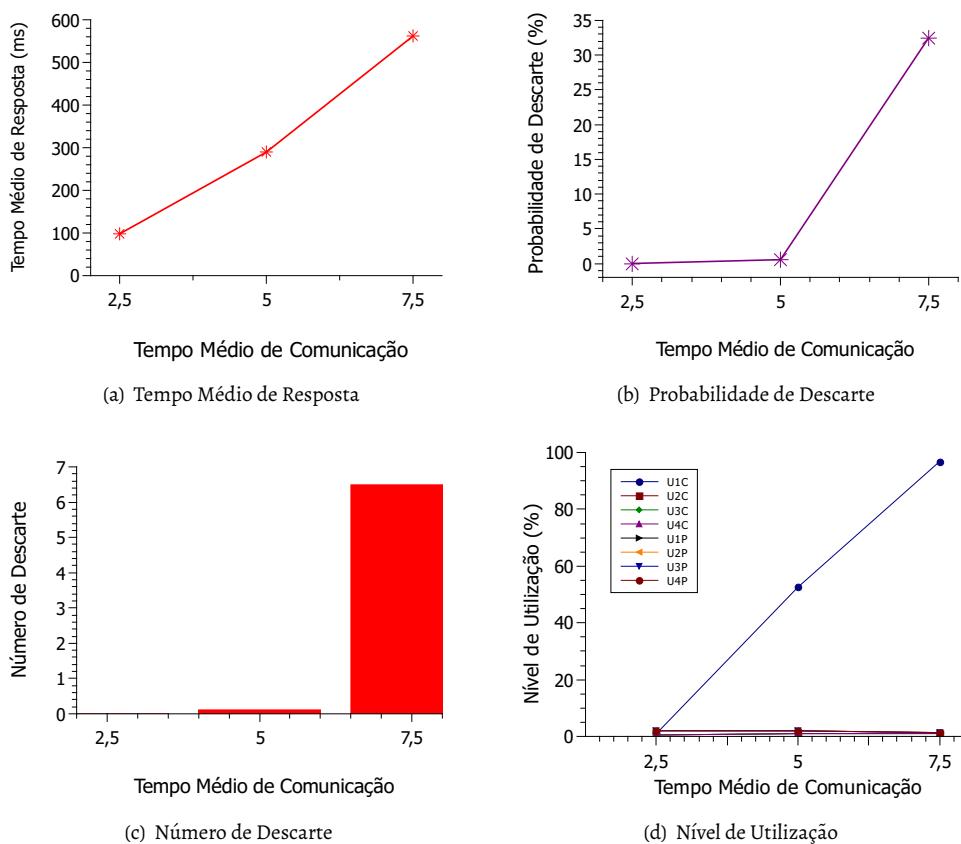


Figura 294 – Resultados da Análise Numérica Variando Tempo de Comunicação (Fonte: Autor)

A Figura 294(a) apresenta os resultados para o MRT. Percebe-se que o comportamento é bem distinto da análise anterior. Inicialmente, o tempo de comunicação é mínimo, e não ocasionam descarte no modelo. Mas ao aumentar o tempo de comunicação o MRT passa a ter valor maior que o dobro do valor anterior. Isso ocorre porque mais de uma transição está sendo alterada em relação ao tempo de comunicação. Quando

o tempo de comunicação chega a 7,5ms pode-se perceber que o MRT chega ao valor máximo observado. Isso ocorre não somente devido ao aumento do processamento de dados, mas também devido ao tempo de espera nas filas serem maiores. A Figura 294(b) apresenta a probabilidade de descarte de novas requisições. Inicialmente, a probabilidade está em 0% tanto para a tecnologia 5G quanto a tecnologia IEEE 802.11p. Porém, com o tipo LoRa, que é mais lento, a probabilidade de descarte é de aproximadamente 50%.

A Figura 294(c) apresenta o número de descartes que ocorreram no sistema. Para LoRa observa-se um número de descarte de 10 requisições. A Figura 294(d) apresenta o nível de utilização das RSUs. A utilização de comunicação da RSU1 (U1C) é o único recurso afetado à medida que se aumenta o tempo médio de comunicação das RSUs, novamente porque esta é o primeiro componente alcançado pelas requisições. A utilização U1C aumenta proporcionalmente ao tempo médio de comunicação, chegando a 100% com a tecnologia LoRa. Os demais recursos sempre ficam próximos de zero pois o AD fixo na análise foi de 5,0ms, o que gera poucas requisições, como visto na análise anterior.

27.16 Validação do Modelo SPN

Esta seção apresenta a validação do modelo SPN proposto. O objetivo desta validação foi comparar o MRT calculado pelo modelo e o MRT coletado por experimentos em um cenário real.

27.16.1 Ambiente Experimental

Diferente das análises anteriores, nesta análise um contexto onde o tempo médio de resposta não é ponto crítico do sistema foi considerado. Portanto, foi considerado valores menores para capacidades de comunicação e processamento das RSUs. Um protótipo de aplicação que simula a comunicação entre quatro RSUs com base na arquitetura proposta na Seção 27.13.1 foi desenvolvido. Nessa arquitetura, a RSU1 é a unidade concentradora que responde cada requisição recebida para ser processada pelas demais RSUs. Em outras palavras, a RSU1 recebe requisições, encaminha para as outras três RSUs, obtém o retorno das tarefas executadas por essas RSUs, processa um resultado final e o transmite aos veículos na sua área de cobertura. Para representar as RSUs foram utilizados quatro microcontroladores *Raspberry Pi*⁴ mostrados na Tabela 187.

Os recursos de processamento e comunicação da aplicação foram representados pelo número de *threads* configuradas para cada microcontrolador. Tais recursos podem ser representados também por número de núcleos do processador, número de containers/VMs, assim como número de portas ou canais para recursos de comunicação. A aplicação recebe requisições com o tempo entre chegadas obedecendo a uma determinada distribuição de probabilidade. Para avaliação do modelo foi utilizado uma distribuição exponencial. A tarefa executada em cada RSU foi um algoritmo de ordenação com complexidade quadrática, que recebe e processa um vetor de números. Outros tipos de tarefas podem ser facilmente adaptadas à aplicação em trabalhos futuros.

Tabela 187 – Dispositivos Adotados nos Experimentos

Configuração dos Dispositivos	Perfis	Funções
Intel Core i5 2,5Ghz 12GB	Gerador de requisições	Gera requisições para serem processadas nas RSUs
Raspberry Pi 3 modelo B+ 1,4 GHz 1GB	RSU1	Recebe, agrupa e encaminha os dados
Raspberry Pi 3 modelo B 1,2 GHz 1 GB	RSU2	Recebe, processa e retorna os dados
Raspberry Pi 3 modelo B 1,2 GHz 1 GB	RSU3	Recebe, processa e retorna os dados
Raspberry Pi 3 modelo B 1,2 GHz 1 GB	RSU4	Recebe, processa e retorna os dados

Fonte: Autor

Um experimento inicial foi realizado para coletar os tempos médios de cada etapa do sistema a fim de alimentar as transições do modelo, dado que a aplicação registra em cada etapa de comunicação e processamento os respectivos tempos associados. A partir desse experimento, os seguintes tempos das transições foram obtidos para alimentar o modelo: $T_{Process1} = 23,5\text{ms}$, $T_{Process2} = 11,0\text{ms}$, $T_{-Process3} = 11,0\text{ms}$, $T_{-Process4} = 11,0\text{ms}$. Todas as marcações de capacidade de processamento (RnP) e

⁴ A Raspberry Pi é um computador de placa única do tamanho de um cartão de crédito ou até menor

de comunicação (RnC) foram configuradas com o valor 4. Esta configuração representa que cada RSU possui apenas quatro *threads* para tratar da transmissão de dados e quatro *threads* para tratar do processamento de requisições. Vale ressaltar que no experimento, para cada AD foram disparadas 15 requisições. Quanto ao modelo, o mesmo foi configurado para disparar somente 15 requisições para ser compatível com o experimento real. Quantidades de requisições diferentes teriam impacto sobre o MRT por causa das filas que são geradas.

27.16.2 Resultados da Validação

A Figura 295 compara os MRTs do experimento com os do modelo, para três casos distintos de tempo de chegada entre requisições. Quanto maior o AD, menos requisições são enviadas e menor é o valor para o MRT. Pode-se observar que um desvio padrão (linha vertical sobre a barra) é expressivo para os MRTs do experimento. Isso ocorre devido ao crescente número de filas geradas, pois, foi simulado um número pequeno de recursos de comunicação e processamento nas RSUs (quatro *threads*). Apesar disso, os resultados dos MRTs foram próximos para cada um dos três casos.

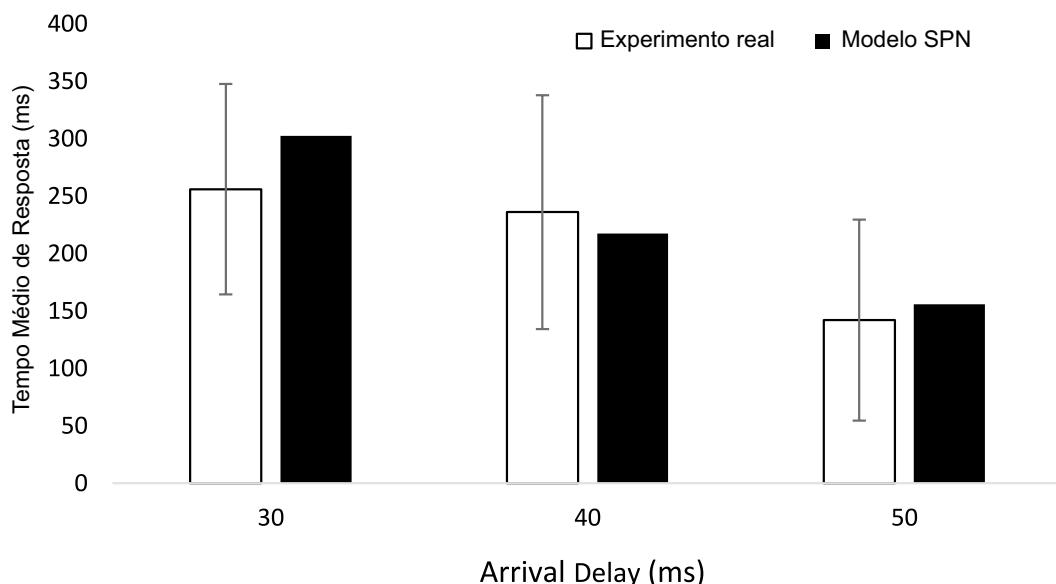


Figura 295 – Comparação entre os MRTs do Modelo e do Experimento (Fonte: Autor)

Após verificar que as amostras de MRT obtidas no experimento possuíam distribuição normal, o Teste T foi aplicado para comparar o MRT gerado pelo modelo com o MRT obtido nas execuções da nossa aplicação nos três casos mostrados na Figura 295. O Teste T é um teste de hipótese útil na estatística quando é necessário comparar médias. Você pode comparar uma média amostral com um valor hipotético ou com um valor alvo usando um teste t para uma amostra. O Teste T usa formulação estatística para rejeitar ou não uma hipótese nula. A seguir, foi verificada a hipótese nula, ou seja, a hipótese que as médias são iguais, a partir do valor p . Considerando um intervalo de confiança de 95%, os valores p foram: AD = 30,0ms, $p = 0,070$; AD = 40,0ms, $p = 0,471$; AD = 50,0ms, $p = 0,573$. Em todos os casos, o valor p é maior que 0,05, rejeitando a hipótese nula e indicando que os resultados gerados pelo modelo são equivalentes estatisticamente ao experimento.

27.17 Trabalhos Relacionados

Semáforos inteligentes e monitoramento do tráfego de veículos em vias públicas tem ganhado foco das pesquisas sobre sistemas inteligentes de transporte, contribuindo para o rápido desenvolvimento da área (Ngo; Huynh-The; Kim, 2019). A infraestrutura para esses sistemas apresentam como elemento básico a RSU. Algoritmos para controle adaptativo de semáforos (YOUNES; BOUKERCHE, 2015a; SILVA; SOARES, 2019; XIAO et al., 2015) requerem ao menos uma RSU como ponto central de processamento e tomada de decisões. As RSUs também auxiliam os protocolos distribuídos de coleta de informações sobre densidade e velocidade do tráfego como o ECODE (YOUNES; BOUKERCHE, 2015b). Adicionalmente, a comunicação entre RSUs é fundamental para as novas abordagens que visam eliminar colisões entre veículos, como em (Ngo; Huynh-The; Kim, 2019), que propõe o uso de três RSUs no dilema do sinal amarelo (YLD). Essas pesquisas e seus possíveis produtos devem se beneficiar do nosso modelo para analisar o impacto de vários padrões de tráfego e ajustá-la aos níveis de desempenho desejáveis.

Diferentes trabalhos propuseram modelos analíticos para avaliação de desempenho de infraestruturas VANETs. Contudo, esses modelos focam em aspectos específicos da camada física ou aplicações específicas em VANETs, e são apenas parcialmente aplicáveis ao domínio de semáforos inteligentes, que deve ser mais amplo, abrangendo inclusive a comunicação entre RSUs. Na Tabela 188, sumarizamos os trabalhos mais relevantes sobre avaliação do desempenho de infraestruturas em VANETs com modelos analíticos, que discutimos a seguir.

Tabela 188 – Trabalhos Relacionados a Avaliação de Desempenho de Infraestruturas VANETs

Trabalho	Avaliação	Atraso/Tempo de Resposta	Utilização da Infraestrutura	Múltiplas RSUs
(LI et al., 2017a)	QoS para Camada Física e MAC	x		
(SHAH; ILHAN; TURELI, 2019)	QoS para Camada Física e MAC	x		
(MALIK; PANDEY, 2018)	Protocolos de autenticação	x		
(ALHEETI et al., 2016)	Detecção de veículos maliciosos	x		
(ZHANG et al., 2016)	Tempo para estacionar	x		
(JUNIOR et al., 2017)	Planejamento de infraestrutura	x	x	
Nossa proposta	Planejamento de infraestrutura	x	x	x

Fonte: Autor

Modelos de fila vêm sendo utilizados eficientemente para avaliar aspectos da camada física e de Controle de Acesso ao Meio (MAC) sem fio em VANETs, especificados no padrão IEEE 802.11p (IEEE, 2010). Um modelo baseado em cadeias de Markov com duas dimensões foi proposto em (LI et al., 2017a) para analisar broadcast e enfileiramento de processos simultaneamente. Em (SHAH; ILHAN; TURELI, 2019), um novo modelo Markoviano foi proposto com foco na mobilidade dos veículos, onde os autores correlacionaram QoS com número, densidade e velocidade dos veículos. Esses modelos focam especificamente em métricas de qualidade de serviço (QoS) como colisão, probabilidade de sucesso na transmissão, atraso e vazão.

Modelos de redes de Petri foram propostos para avaliar desempenho de múltiplas aplicações em VANETs. Em (MALIK; PANDEY, 2018), foi proposto um modelo de redes de Petri para analisar o desempenho de protocolos de autenticação de veículos na infraestrutura quanto à possibilidade de falhas e aplicabilidade em tempo real. Métodos para detecção de veículos maliciosos em VANETs são analisados em (ALHEETI et al., 2016) com um modelo baseado em redes de Petri Difusas (FPNs) para analisar padrões de comunicação entre veículos e assim detectar suspeitos. Um modelo para analisar busca de vagas em estacionamentos foi proposto em (ZHANG et al., 2016) utilizando Redes de Petri Programadas para estimar o tempo para estacionar, considerando tempo de condução, ocupação do estacionamento e tempo de confirmação do provedor. Esses modelos focam em aplicações específicas em VANETs, e não tratam cenários para uso de semáforos inteligentes, que é o foco do nosso trabalho.

O planejamento e dimensionamento de infraestruturas para VANETs foi o foco de apenas um dos trabalhos relacionados. Em (JUNIOR et al., 2017), um modelo SPN foi proposto para analisar o desempenho da comunicação V2I em arquiteturas VANETs com foco em uma única RSU. Os autores analisam tempo de resposta, atrasos e utilização da infraestrutura dado a variação do tráfego de veículos, mas não tratam da comunicação entre múltiplas RSUs, aspecto relevante no monitoramento de tráfego para os semáforos inteligentes (YOUNES; BOUKERCHE, 2015b; Ngo; Huynh-The; Kim, 2019). Este trabalho é o único até o

momento que foca nesse aspecto, assim como provê, em um único modelo, as métricas para avaliação de infraestruturas mencionadas, que podem ser utilizadas para configuração de semáforos inteligentes em diferentes cenários.

27.18 Conclusões

Este trabalho propôs dois modelos SPN para representar e avaliar o desempenho de duas arquiteturas VANETs. Os dois modelos permitem estimar o MRT, o nível de utilização de RSUs, a probabilidade de descarte e o número de mensagens descartadas de um cenário de rede veicular. O primeiro modelo representa e avalia o desempenho de uma arquitetura VANET com um semáforo. O Projetista pode configurar até 6 parâmetros no primeiro modelo. O segundo modelo representa e avalia o desempenho de uma arquitetura VANET com múltiplos semáforos cooperativos. O projetista pode configurar até 18 parâmetros no segundo modelo, o que permite um alto nível de flexibilidade de avaliação. As análises numéricas permitiram observar o comportamento das quatro métricas de duas formas: em função da variação do tempo entre chegadas de novas mensagens e em função da variação do tempo de comunicação entre as RSUs, porém muitas outras análises podem ser realizadas. Os resultados das análises permitiram concluir que o tempo médio de resposta é muito dependente do enfileiramento de mensagens nas RSUs. Os descartes e o nível de utilização decaem com o aumento progressivo do tempo entre chegadas. Por fim o modelo foi validado com um experimento real usando microcontroladores e um gerador sintético de requisições. Como trabalho futuro, pretende-se estender o modelo para explorar a comunicação entre OBUS sem infraestrutura intermediária, ou seja, arquiteturas V2V.

Alocação de Servidores em Computação na Borda Móvel Usando Redes de Petri Estocásticas

Autores: Daniel de Carvalho Borges e Francisco Airton Silva

Resumo

O Mobile Edge Computing (MEC) é uma arquitetura de rede que tira proveito dos recursos de computação em nuvem (como alta disponibilidade e elasticidade) e utiliza recursos computacionais disponíveis na borda da rede para aprimorar a experiência do usuário móvel, diminuindo o serviço latência. Para atingir esse objetivo, as soluções MEC precisam alojar dinamicamente as solicitações o mais próximo possível de seus usuários. No entanto, o posicionamento da solicitação não depende apenas da localização geográfica dos servidores, mas também de suas configurações. Com base nesse fato, este trabalho propõe um modelo de Rede de Petri Estocástica (SPN) para representar um cenário do MEC e analisar seu desempenho, focando nos parâmetros que podem impactar diretamente o Tempo Médio de Resposta (MRT) e o Nível de Utilização. Para apresentar a aplicabilidade do nosso trabalho, apresentamos três estudos de caso com análises numéricas usando valores reais. O objetivo principal é fornecer um guia prático para ajudar os administradores de infraestrutura de computadores a adaptar suas arquiteturas, encontrando uma troca entre MRT e nível de uso.

Palavras-chaves: Computação Móvel de Borda, Internet das Coisas, Modelos Estocásticos, Alocação de Servidores.

28.1 Introdução

De acordo com um relatório da CISCO, o número de dispositivos móveis chegará a cerca de 11,6 bilhões até 2020 ([JUNG, 2016](#)). Essa grande adoção é fundamentalmente impulsionada pelo aumento do número de novos usuários móveis, bem como pelo desenvolvimento de aplicativos interativos ([KITANOV; MONTEIRO; JANEVSKI, 2016](#)) ([BECK et al., 2014](#)). No entanto, o crescente número de dispositivos móveis tem um efeito

extremo sobre as redes móveis, trazendo desafios significativos para as empresas de telecomunicações (CAU et al., 2016). Os dispositivos móveis têm baixa capacidade de armazenamento e as redes celulares são caracterizadas por alto consumo de energia, baixa largura de banda e alta latência (ORSINI; BADE; LAMERSDORF, 2015). Além disso, o crescimento exponencial da Internet das Coisas (IoT) promete tornar as redes sem fio ainda mais difíceis (BORGIA et al., 2016). Para lidar com esses desafios, muitos pesquisadores propuseram arquiteturas otimizadas para esse contexto, como o Mobile Cloud Computing (MCC) (MAROTTA et al., 2015; Sucipto et al., 2017).

A MCC é a integração da computação em nuvem e da computação móvel, que fornece recursos adicionais para dispositivos móveis ao centralizar seus recursos na infraestrutura de nuvem (DINH et al., 2013). No entanto, com vários dispositivos móveis, a MCC enfrenta desafios notáveis, como alta latência, vulnerabilidade de segurança e transmissão de dados limitada, porque os recursos da nuvem estão frequentemente longe dos usuários finais (JARARWEH et al., 2016a).

Como uma evolução da arquitetura do MCC, surgiu o Mobile Edge Computing (MEC). O principal objetivo do MEC é abordar os desafios que a MCC vem enfrentando, implantando recursos ainda mais perto dos usuários finais, ou seja, na borda da rede, permitindo que as operações de computação e armazenamento sejam realizadas mais perto do dispositivo de origem (JARARWEH et al., 2016b). O conceito do MEC surgiu com o objetivo de unir as operadoras de telecomunicações, TI e empresas de computação em nuvem para fornecer serviços em nuvem diretamente na borda da rede. Diferentemente dos sistemas de computação em nuvem tradicionais em que nuvens públicas remotas são utilizadas, os servidores MEC são de propriedade da operadora de rede e são implementados diretamente nas Estações Base (BSs) celulares ou nos Pontos de Acesso sem fio (APs) locais usando uma plataforma de computação genérica. Com essa posição, o MEC permite a execução de aplicativos nas proximidades dos usuários finais, reduzindo substancialmente o atraso de ponta a ponta e liberando a carga sobre as redes de *backhaul*. Como o MEC ainda é considerado um tópico recente, algumas oportunidades de pesquisa estão abertas para serem exploradas, como avaliação de desempenho de arquiteturas MEC (Tran et al., 2017). Outro aspecto está relacionado ao posicionamento dos servidores de borda no campo MEC. Pouca atenção foi dada ao efeito de descarregar as cargas de trabalho de usuários móveis para servidores de borda e a colocação de servidores de borda no desempenho de aplicativos móveis.

Modelos analíticos são adequados para avaliar o desempenho de sistemas complexos, principalmente durante os estágios iniciais. A modelagem estocástica é um método formal adequado e amplamente utilizado para avaliação de desempenho em sistemas concorrentes com mecanismos de sincronização e comunicação. As Redes de Petri Estocásticas (SPNs) são casos especiais de modelos estocásticos (DESROCHERS; AL-JAAR; SOCIETY, 1995), que são capazes de configurar equações do estado, equações algébricas e outros modelos matemáticos que governam o comportamento dos sistemas. O uso de SPNs já foi aplicado com sucesso no contexto da MCC em trabalhos anteriores (SILVA et al., 2018; PINHEIRO et al., 2018b; PINHEIRO et al., 2018a). No entanto, de acordo com o nosso conhecimento, os modelos analíticos de desempenho nas arquiteturas do MEC são escassos até o momento presente.

Neste documento, nos concentramos na colocação do servidor MEC em um ambiente de computação de borda móvel que fornece cobertura de Internet sem fio para usuários móveis em uma área metropolitana de grande escala. Para isso, propomos um modelo SPN para representar a arquitetura MEC e avaliar o *trade-off* entre *tempo médio de resposta (MRT)* e *utilização de recursos*. O *MRT* significa o tempo para uma solicitação ser processada por servidores MEC e retornar para dispositivos clientes móveis. A *utilização de recursos*, como o nome indica, significa a porcentagem de recursos que estão no modo ativo. Em outras palavras, a utilização de recursos está relacionada ao número de computadores (máquina / VMs / containeres físicos) que estão sendo usados.

Em resumo, as principais contribuições deste trabalho são:

- apresentamos um modelo analítico, que serve como uma ferramenta útil para administradores de sistemas avaliarem o desempenho de diferentes arquiteturas MEC, especialmente durante a fase de planejamento antes da implementação;
- nós fornecemos um conjunto de análises numéricas com dados reais que servem como um guia prático para a análise de desempenho em arquiteturas MEC.

28.1.1 Definição do Problema

A capacidade dos servidores e a forma em que estão distribuídos podem influenciar diretamente no desempenho da arquitetura MEC. No entanto, avaliações de desempenho aprofundadas nesse aspecto possuem alta complexidade e alto custo financeiro.

28.1.2 Objetivos

28.1.2.1 Objetivo Geral

O objetivo geral deste trabalho é avaliar o desempenho de uma arquitetura MEC sob a perspectiva de planejamento de arquiteturas mais eficientes, e auxiliar na alocação dos recursos afim de obter o melhor desempenho possível.

28.1.2.2 Objetivos Específicos

1. Diminuir o tempo médio de resposta.
2. Reduzir o nível de utilização de recursos, evitando ociosidade.

28.1.3 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma: A seção 28.2 apresenta os conceitos necessários para entender a proposta; A seção 28.3 discute os principais trabalhos relacionados; A seção 28.4 apresenta a arquitetura do MEC que foi usada como base para a construção do modelo; no seção 28.5, apresentamos nosso modelo SPN básico com as respectivas métricas e estudos de caso; na seção 28.6, apresentamos um modelo SPN estendido também com as respectivas métricas e estudos de caso; A seção 28.7 apresenta a validação do modelo SPN proposto com o objetivo de comparar os resultados obtidos no modelo e no experimento real, e a seção 28.8 traça algumas conclusões e trabalhos futuros.

28.2 Background

Esta seção apresenta os principais conceitos necessários para melhor acompanhar a proposta deste trabalho. Primeiro, apresentamos os aspectos gerais dos SPNs e, em seguida, algumas características da arquitetura do MEC são detalhadas.

28.2.1 Redes de Petri Estocásticas

Petri Nets (PNs) são gráficos matemáticos de modelagem aplicável a muitos sistemas e útil para descrever e estudar sistemas de processamento de informações que são caracterizados como sendo concorrentes, assíncronos, distribuídos, paralelos, não-determinísticos e estocásticos. As redes de Petri Estocásticas (SPNs) são casos especiais de PNs. Os modelos SPN foram inicialmente projetados como uma ferramenta que permitia a integração da descrição formal, prova de correção e avaliação de desempenho. A Figura 296 apresenta os componentes usados para modelar um SPN.

A Figura 297 apresenta um modelo básico de SPN (OLIVEIRA et al., 2017) para avaliação de desempenho adotado neste trabalho. A transição “generate” gera tokens que correspondem a solicitações de serviço. Para cada token gerado colocado no lugar “generated”, é feita uma escolha. Um token pode ser enfileirado para ser processado pelo servidor se houver uma posição livre na fila. Se a fila estiver cheia, no entanto, o token será descartado. O lugar “free” corresponde aos recursos disponíveis (nós de serviço). Mais especificamente, o número de recursos disponíveis é marcado com k . Os tokens que esperam pelo serviço são colocados no local “queue”. O serviço de transição representa o processamento dos pedidos enfileirados pelo servidor.

Agora, vamos considerar um exemplo, ilustrado na Figura 298. Bob, um administrador do sistema, tem hoje um servidor executando um serviço da Web específico para seus usuários. O servidor hoje não é suficiente para atender a demanda, e ele quer comprar mais servidores, mas ao mesmo tempo evitando a

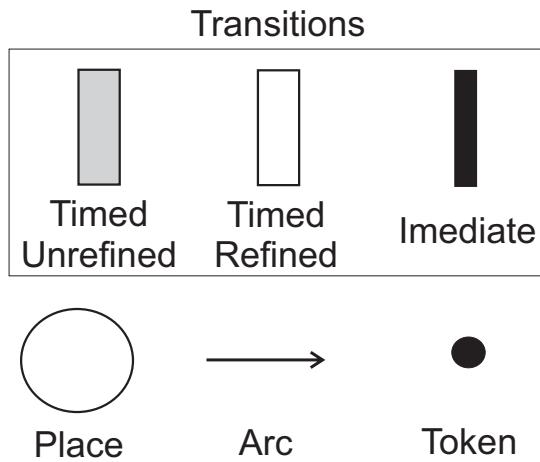


Figura 296 – Componentes SPN.

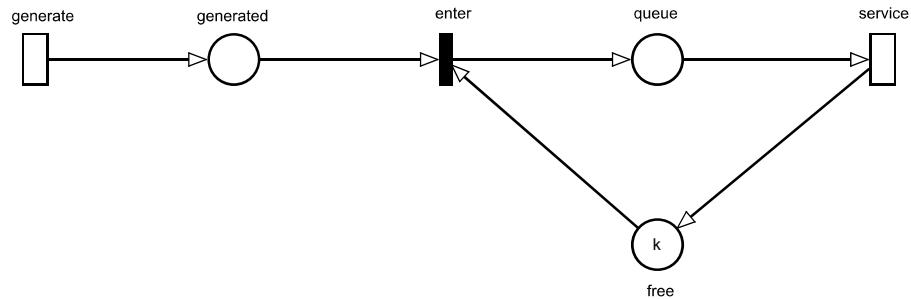


Figura 297 – Exemplo de Modelo SPN.

ociosidade. *Como predir quantos servidores Bob precisa usar no modelo básico SPN da Figura 297?*. Basicamente Bob precisa inserir três informações: (i) gerar transição - intervalo de tempo médio entre a chegada das solicitações atuais. (ii) transição de serviço - tempo médio necessário para processar apenas uma solicitação. (iii) marcação k - quantos servidores Bob está disposto a comprar. Depois disso, através de simulação ou análise, resolvemos o modelo SPN para calcular o MRT e a utilização de recursos. Essas métricas são detalhadas na seção 28.5.1.

28.2.2 Computação de Borda Móvel

O termo MEC foi padronizado pelo Instituto Europeu de Normas de Telecomunicações (ETSI)¹ e pelo Industry Specification Group (ISG). O grupo ISG inclui a Nokia Networks, a Intel, a Vodafone, a IBM, a Huawei e a NTT DOCOMO, para citar algumas. O MEC também é reconhecido pela European 5G Parceria público-privada de infra-estrutura como uma tecnologia emergente para redes 5G (HU et al., 2015b).

De acordo com o ETSI, o MEC é definido da seguinte forma: "A computação de borda móvel fornece um ambiente de serviços de TI e recursos de computação em nuvem na borda da rede móvel, dentro da rede de acesso de rádio (RAN) e próxima aos assinantes móveis" (HU et al., 2015b). O MEC oferece recursos de computação em nuvem no RAN. Permitindo o tráfego móvel direto entre a rede principal e o usuário final, o MEC conecta o usuário diretamente a rede de borda habilitada para serviço de nuvem mais próxima. Implantar o MEC na estação base melhora a computação e evita gargalos e falhas do sistema (JARARWEH et al., 2016a; SATRIA; PARK; JO, 2017a). De acordo com o white paper publicado pela ETSTI, o MEC pode ser caracterizado pelas seguintes premissas:

¹ <https://www.etsi.org/technologies/multi-access-edge-computing>

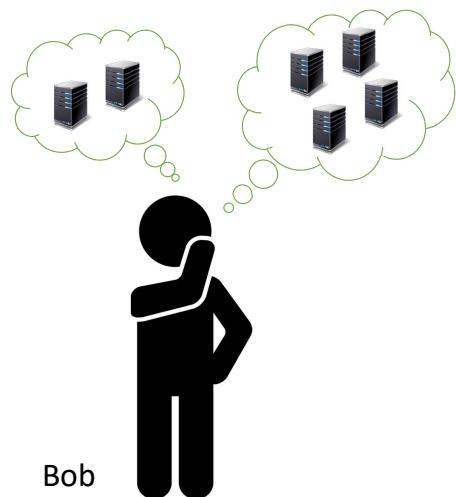


Figura 298 – Administrador do Sistema com Perguntas Sobre Qual Servidor Montar.

Local: As plataformas MEC podem ser executadas isoladas do resto da rede, enquanto elas têm acesso a recursos locais. Isso é muito importante para cenários de máquina a máquina (M2M). A propriedade do MEC de segregação de outras redes também torna menos vulnerável.

Proximidade: Sendo implantado no local mais próximo, o MEC tem uma vantagem para analisar e materializar grandes volumes de dados. Isto também é benéfico para dispositivos com fome de computação, como realidade aumentada (AR), análise de vídeo etc.

Latência Inferior: Os serviços MEC são implantados no local mais próximo aos dispositivos do usuário, isolando a movimentação de dados da rede principal. Assim, a experiência do usuário é considerada de alta qualidade, com latência ultrabaixa e alta largura de banda.

Reconhecimento de local: Dispositivos distribuídos por borda utilizam sinalização de baixo nível para compartilhamento de informações. MEC recebe informações de dispositivos de borda dentro da rede de acesso local para descobrir a localização dos dispositivos.

Informações de Contexto da Rede: Aplicativos que fornecem informações de rede e serviços de dados de rede em tempo real podem beneficiar negócios e eventos implementando o MEC em seu modelo de negócios. Com base nas informações em tempo real da RAN, esses aplicativos podem estimar o congestionamento da célula de rádio e da largura de banda da rede. Isso os ajudará no futuro a tomar decisões inteligentes para uma melhor prestação de serviços aos clientes.

Portanto, o MEC traz recursos de computação e armazenamento para a borda da rede móvel, melhorando assim a velocidade com que os dados são processados (MACH; BECVAR, 2017). Como uma tecnologia de ponta promissora, o MEC pode ser aplicado a cenários móveis, sem fio e com fio usando plataformas de software e hardware, localizadas na borda da rede e próximas aos usuários finais, enquanto ainda fornece integração perfeita de vários serviços de aplicativos de provedores e fornecedores de assinantes móveis, empresas e outros (ABBAS et al., 2017).

Os servidores MEC são implantados na plataforma de computação genérica dentro da rede e permitem que aplicativos com reconhecimento de atraso sejam executados em estreita proximidade com os usuários finais. Essa abordagem alivia o *backhaul* e a rede principal e é crucial para permitir serviços móveis de baixa latência, alta largura de banda e agilidade (TRAN et al., 2016). Devido à distribuição geográfica densa, proximidade aos consumidores, suporte de alta mobilidade e plataforma aberta, o MEC pode suportar aplicativos e serviços com latência reduzida e melhor qualidade de serviço. Assim, o MEC está se tornando um importante facilitador de aplicativos e serviços de IoT centrados no consumidor que exigem operações em tempo real (CORCORAN; DATTA, 2016; CHEN et al., 2016; LI; WU, 2018; PREMSANKAR; TALEB, 2018).

A arquitetura tradicional do MEC permite o uso de seus serviços com baixa latência, reconhecimento de localização e suporte a mobilidade para compensar as desvantagens da computação em nuvem ([SATRIA; PARK; JO, 2017b](#)). No MEC, aplicativos sensíveis à latência e intensivos de computação, como realidade aumentada e processamento de imagens, podem ser hospedados na borda da rede ([GUPTA; JAIN; CHAN, 2016](#)).

A Figura 299 apresenta tal arquitetura. A mesma está dividido em três camadas:

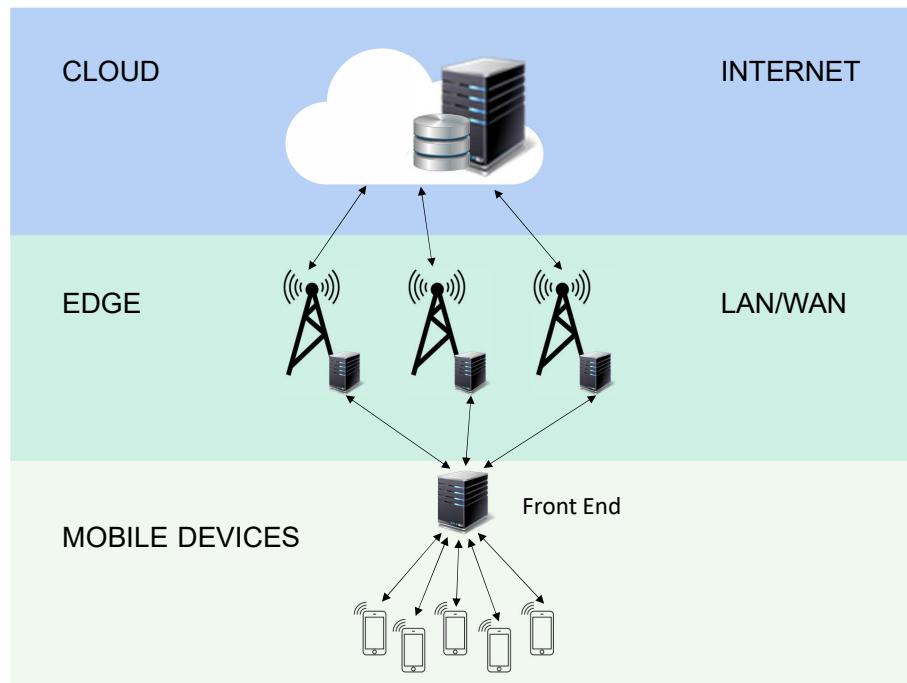


Figura 299 – Arquitetura MEC Completa

1. Cloud: grande armazenamento de dados.
2. Edge: Exibe as torres onde os servidores serão instalados.
3. Dispositivos móveis: exibe dispositivos móveis (smartphones, tablets) junto com o FrontEnd.

Dispositivos móveis enviam solicitações para FrontEnd, que é responsável por encaminhá-las para o servidor na torre mais próxima. Quando a solicitação exige uma resposta mais rápida, o processamento ocorre no próprio servidor, sem a necessidade de ser roteado para a nuvem. A comunicação que ocorre nesta camada é feita através de LAN/WAN. Quando a solicitação demora mais para ser atendida, a fim de obter uma resposta mais precisa, ela vai para a camada de nuvem, onde a comunicação ocorre pela Internet, portanto, pode levar um pouco mais de tempo para ser resolvido. Com isso em mente, concentraremos nosso trabalho nas duas primeiras camadas, onde o processamento ocorre mais próximo do usuário e onde a latência é menor. Mais especificamente, este trabalho explora o problema da colocação de servidores. Considerando essas três torres que estão localizadas em lugares distintos onde devo implantar servidores com poderes de computação distintos.

28.3 Trabalhos Relacionados

Alguns trabalhos relacionados propõem arquiteturas MEC em **cooperação com outras camadas**. No entanto, esses documentos não exploram diferentes configurações de recursos computacionais. Tong et. al propõe

implantar servidores em nuvem na borda da rede e posicionar os servidores em hierarquias distribuídas geograficamente para usar a nuvem para atender aos picos de cargas de solicitações provenientes do MEC (TONG; LI; GAO, 2016). Chen et. al foca em um algoritmo de otimização para calcular onde processará a carga de trabalho (MEC ou nuvem). Os autores usam como validação, aplicativos genéricos que exigem um alto nível de processamento (CHEN et al., 2016). Autores em (LI; WU, 2018) concentram-se na capacidade do MEC e da nuvem de processar grandes cargas de dados. O objetivo principal do trabalho é tentar reduzir a latência entre as duas camadas atribuindo pesos às complexidades das tarefas a serem executadas remotamente.

Alguns artigos enfocam o problema do consumo de energia de dispositivos móveis com o auxílio de uma arquitetura MEC. Zhang et. al estudaram mecanismos de *offloading* investigando uma arquitetura MEC e redes 5G heterogêneas (KE; S; L, 2016). Os autores formularam um problema de otimização para minimizar o consumo de energia, observando o tempo de processamento e a transferência de dados. Trinh et. al estudaram o potencial do MEC para mitigar a limitação de bateria de dispositivos IoT (TRINH; YAO, 2017). Os autores utilizaram uma aplicação de reconhecimento facial para demonstrar a viabilidade de descarregar políticas de decisão. Mao et. al também propuseram um algoritmo de otimização de bateria em dispositivos móveis (MAO; ZHANG; LETAIEF, 2016). O algoritmo inclui frequências do servidor MEC, ciclo da CPU e taxa de latência. Uma vantagem desse algoritmo é que as decisões dependem apenas das informações instantâneas do lado do servidor, sem exigir informações sobre a distribuição das solicitações da tarefa.

Ao contrário desses estudos, nosso trabalho não analisa diretamente o gasto de energia de dispositivos clientes. No entanto, as métricas de MRT adotadas aqui estão proporcionalmente relacionadas ao gasto de energia de dispositivos clientes. Quanto mais tempo a solicitação levar para executar, maior será o consumo de energia.

Os trabalhos relacionados mais próximos são aqueles que lidam com o **planejamento de infraestrutura do MEC**. Premsankar et. al realizara uma avaliação experimental real com um jogo eletrônico de alta interatividade, mas com a limitação de ter apenas um cliente móvel (PREMSANKAR; TALEB, 2018). O trabalho de Jararweh et. al desenvolveu um simulador de arquitetura que incluía uma camada Cloudlet e uma camada MEC (JARARWEH et al., 2016a). Os autores tentam aumentar a área de cobertura para usuários móveis, onde os usuários podem fazer solicitações com custos mínimos em termos de gasto de energia. Algumas limitações deste trabalho advêm do fato de que os autores só adotam o número de pedidos como um parâmetro de entrada e o uso de um simulador customizado sem explicações detalhadas sobre suas características. Liu et. al adotaram um modelo de cadeia de Markov como um processo de decisão sobre onde as tarefas devem ser executadas (localmente ou no servidor MEC) (LIU et al., 2016). O modelo leva em consideração o status de enfileiramento do buffer de tarefas, o estado da execução do dispositivo móvel e o estado da rede. No entanto, os autores não consideram o servidor MEC com vários nós paralelos e, portanto, não aproveitam o potencial do paralelismo de servidor. Badri et. al também usou cadeias de Markov para decidir onde executar solicitações em várias torres MEC (BADRI et al., 2017). O algoritmo leva em consideração o movimento dos usuários, o custo de comunicação entre usuários e servidores, o custo de execução de cada servidor e o custo de alocação. Nossa trabalho não se concentra em vários servidores, mas no planejamento de uma infraestrutura mínima com um único servidor MEC, considerando o paralelismo em vários *containeres*.

Ao contrário da nossa proposta, os trabalhos mencionados acima não exploram o nível de utilização de recursos e apenas alguns consideram a métrica MRT. Todos os artigos são limitados em termos de parametrização das avaliações, com no máximo três parâmetros de configuração de arquitetura. Nenhum dos artigos (com exceção de (PREMSANKAR; TALEB, 2018)) abordou aplicativos com um alto nível de interação. Além disso, nenhum dos jornais adotou redes de Petri Estocásticas.

28.4 Visão Geral da Arquitetura

Este trabalho apresenta um SPN para modelar uma arquitetura MEC em que os recursos de servidores únicos são paralelizados com microsserviços em execução em vários *containeres*. A Figura 300 ilustra a arquitetura do MEC que estamos considerando para modelagem e análise de desempenho. A arquitetura é

composta de três partes principais: **Mobile Devices**, **FrontEnd** e **Edge Computing**. Os dispositivos móveis solicitam serviços para o FrontEnd e o FrontEnd distribui solicitações entre a computação de borda.

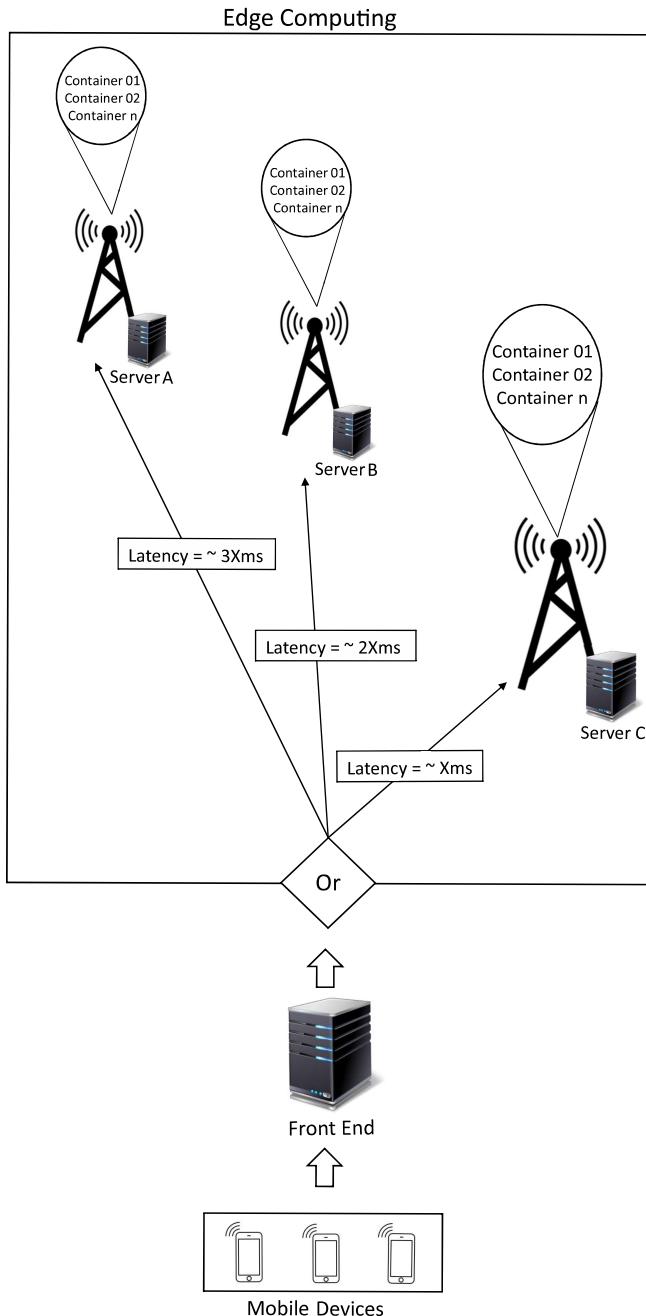


Figura 300 – Arquitetura MEC Adotada para Avaliação de Desempenho.

Os fluxos de dados são gerados por aplicativos em execução no **Mobile Devices**. Esses aplicativos podem variar de aplicativos de monitoramento de integridade ([RAHMANI et al., 2018](#)) para aplicativos de jogos ([ZHANG et al., 2019](#)), por exemplo. Como mencionado anteriormente, neste documento, nos concentramos na borda, e não na nuvem, com o objetivo de trabalhar com aplicativos altamente responsivos nos quais é necessária uma resposta rápida. Portanto, a proximidade de servidores e o alto poder do computador são obrigatórios para fornecer disponibilidade eficiente de recursos e baixo tempo médio de resposta.

O **FrontEnd** é responsável por receber solicitações dos dispositivos móveis e encaminhá-las aos servidores. Para isso, o servidor mestre é responsável por escolher qual servidor é mais adequado para executar a solicitação. Neste trabalho, para tomar essa decisão, estamos considerando dois parâmetros diferentes: (a) cada servidor pode ter diferentes configurações (por exemplo, número de núcleos) e (b) a distância da estação base para os dispositivos móveis, o que afeta a decisão de posicionamento do servidor. Dada essa arquitetura, o objetivo principal deste trabalho é propor um SPN para avaliar o desempenho de diferentes configurações e encontrar a solução de posicionamento de recursos mais eficiente.

Na camada **Edge Computing**, estamos considerando servidores MEC com latência distinta (Servidor A, Servidor B e Servidor C, como mostrado na Figura 300). Cada servidor tem um conjunto de nós escravos que são basicamente micro serviços executados em *containeres*. Cada *container* é executado em um único núcleo de servidor. Embora as máquinas virtuais tenham sido altamente adotadas no campo de pesquisa do MEC, os *containeres* permitem maior flexibilidade para dimensionar o poder de computação de acordo com a demanda dinâmica.

28.5 Modelo SPN Base

Um modelo de SPN é útil quando um gerente de serviços deseja planejar e analisar alterações no sistema antes de implementá-las. A Figura 301 apresenta o modelo SPN proposto, composto por três blocos principais (destacados em vermelho):

1. **Bloco de Admission:** responsável por gerar solicitações de usuários;
2. **Bloco FrontEnd:** representa o servidor FrontEnd e é responsável por receber as solicitações do usuário e encaminhá-las para um dos servidores disponíveis. As políticas de平衡amento de carga podem ser aplicadas neste momento. Neste trabalho as solicitações são encaminhadas para qualquer um dos servidores disponíveis;
3. **Bloco Edge Computing:** representa os servidores (A, B e C) que são responsáveis por receber e processar os pedidos, distribuindo os dados entre os escravos do nó, que em nosso trabalho são *containers*.

Tabela 189 apresenta a descrição dos elementos do modelo. As transições temporizadas são configuradas com distribuições de probabilidade e o administrador do sistema deve preencher essas distribuições de acordo com a literatura ou realizar medições do sistema. Todas as transições temporizadas têm infinitas semânticas de servidor, o que significa que as transições são disparadas em paralelo quando as transições são acionadas.

O bloco *Admission* consiste em dois lugares, *P_Arrival* e *P_ArrivalQueue*. *P_Arrival* representa a geração de solicitações de usuários e *P_ArrivalQueue* representa a aceitação desses pedidos na fila. A transição *T_Arrival* é configurada com a taxa de solicitação de chegada. Neste modelo, consideramos que o tempo de chegada é distribuído exponencialmente, mas pode ser facilmente modificado para se adequar a outras distribuições. A transição *ND* (atraso de rede) representa os tempos de envio e recebimento da solicitação para o servidor *FrontEnd*. *ND* é acionado assim que há um token em *P_ArrivalQueue* e pelo menos um token em *P_FrontEndCapacity*.

Quando a transição *ND* dispara, o bloco *FrontEnd* é atingido e um token é consumido dos lugares *P_ArrivalQueue* e *P_EncurtoCapacidade* e um token é produzido em locais *P_FrontEndInProcess* e *P_Arrival*. A quantidade de tokens em *P_FrontEndInProcess* representa o enfileiramento de pedidos em *FrontEnd*. A fila acontece porque não há recursos disponíveis nos servidores. Quando o processamento da solicitação é iniciado, os tokens são consumidos a partir do *P_FrontEndInProcess*. Se houver recursos disponíveis em um dos três servidores, uma das transições *T_A*, *T_B* ou *T_C* será acionada e a solicitação seguirá o caminho a ser processado.

As transições *T_A*, *T_B* e *T_C* representam o tempo de transferência dos servidores *FrontEnd* para o MEC. Os blocos de servidores do MEC têm a mesma operação. Então, considerando o Servidor A, a transição *P_A* representa o início do processamento. O fogo *P_A* está condicionado ao número de nós disponíveis para processar pedidos em *P_SlavesA* (marcação SAC). Dado o modelo proposto, é possível configurar 15

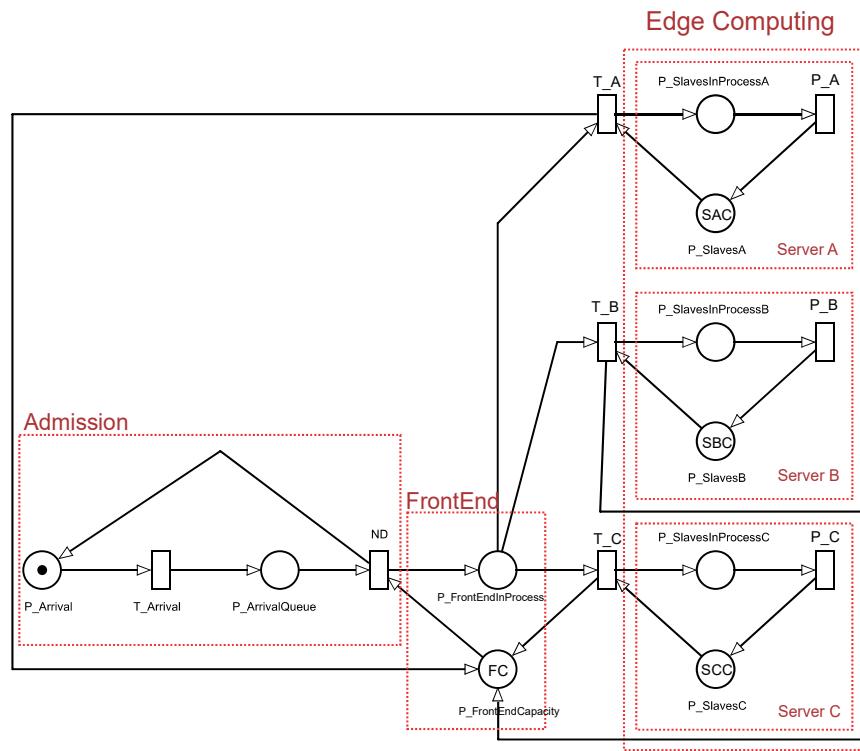


Figura 301 – Modelo SPN para a Arquitetura MEC.

parâmetros (ver Tabela 189), dando a possibilidade de executar um grande número de diferentes cenários. Os parâmetros configuráveis são: 8 transições temporizadas e 4 lugares relacionados às capacidades dos recursos. Qualquer configuração desses parâmetros impacta substancialmente o tempo médio de serviço e, consequentemente, o custo da infraestrutura. A possibilidade de executar um grande número de cenários é uma das principais contribuições deste trabalho.

28.5.1 Métricas de Desempenho

Nesta subseção, definimos as métricas de desempenho que usamos para avaliar a arquitetura MEC com base no modelo de SPN proposto. Neste trabalho, consideramos duas métricas: o tempo médio de resposta (MRT), que se refere ao tempo necessário para que uma solicitação seja executada no sistema, e o nível de utilização que lida com o consumo de uma dada execução nas máquinas presentes nos servidores. O MRT pode ser obtido a partir da Lei de Little (LITTLE, 1961), que relaciona o número médio de pedidos em andamento em um sistema, *RequestsInProcess*, a taxa de chegada de novas solicitações, *ARR* e tempo médio de resposta, *MRT*. A taxa de chegada é o inverso do atraso de chegada, *AD*. A Lei de Little requer um sistema estável, o que significa que a taxa de chegada é menor que o tempo de serviço. Portanto, Equação ?? corresponde à Lei do Little para o MRT usado em nosso modelo:

$$MRT = \frac{RequestsInProcess}{ARR} \quad (28.1)$$

A equação 28.2 define como calculamos o número de solicitações em andamento, ou seja, *RequestsInProcess*. Para isso, somamos a quantidade de tokens em cada lugar que representa uma requisição em processo, *Esp(Place)*, onde $Esp(Place) = (\sum_{i=1}^n P(m(Place)) = i) times i$ e $Place = \{FC, P_EslavesInProcessA, P_SlavesInProcessB,$

Tabela 189 – Descrição dos Elementos do Modelo SPN

Tipo	Elemento	Descrição	Semântica do Servidor
Lugares	P_Arrival P_InputQueue	Aguarda novos pedidos Fila de entrada antes de acessar o servidor FrontEnd	X X
	P_FrontEndInProcess P_FrontEndCapacity	Solicitações na fila do servidor FrontEnd Capacidade do FrontEnd	X X
	P_SlavesInProcessA P_SlavesA	Solicitações na fila do servidor A Capacidade do Servidor A	X X
	P_SlavesInProcessB P_SlavesB	Solicitações na fila do servidor B Capacidade do Servidor B	X X
	P_SlavesInProcessC P_SlavesC	Solicitações na fila do servidor C Capacidade do Servidor C	X X
	ND	Atraso na rede	Servidor Infinito
Transições Temporizadas	T_A T_B T_C	Tempo de transferência para o servidor A Tempo de transferência para o servidor B Tempo de transferência para o servidor C	Servidor Infinito Servidor Infinito Servidor Infinito
	P_A	Tempo de processamento do servidor A considerando uma solicitação	Servidor Infinito
	P_B	Tempo de processamento do servidor B considerando uma solicitação	Servidor Infinito
	P_C	Tempo de processamento do servidor C considerando uma solicitação	Servidor Infinito
	FC SAC SBC SCC	Capacidade máxima do FrontEnd Capacidade máxima do servidor A Capacidade máxima do servidor B Capacidade máxima do servidor C	X X X X
Marcações de Lugares			

$P_{SlavesInProcessC}$.

$$\begin{aligned} \text{RequestsInProcess} = & Esp(P_{FrontEndInProcess}) + \\ & Esp(P_{SlavesInProcessA}) + \\ & Esp(P_{SlavesInProcessB}) + \\ & Esp(P_{SlavesInProcessC}) \end{aligned} \quad (28.2)$$

O uso percentual de recursos de cada máquina, n , é dado por $U(n)$ (veja Equações 28.3, 28.4 e 28.5). O uso de recursos é calculado pelo número de tokens na fila que estão consumindo recursos do servidor dividido pela capacidade total de recursos daquele servidor. Em outras palavras, o uso de recursos é a quantidade de nós escravos que estão sendo usados. Dada a utilização por servidor, também podemos calcular o uso de todos os recursos disponíveis no sistema usando a Equação 28.6, em que $n_{servers}$ é o número total de servidores no sistema.

$$U(A) = \frac{Esp(P_{SlavesInProcessA})}{SCA} \quad (28.3)$$

$$U(B) = \frac{Esp(P_{SlavesInProcessB})}{SCB} \quad (28.4)$$

$$U(C) = \frac{Esp(P_{SlavesInProcessC})}{SCC} \quad (28.5)$$

$$U(\text{all}) = \frac{U(A) + U(B) + U(C)}{n_{servers}} \quad (28.6)$$

28.5.2 Estudos de caso

A principal contribuição do nosso modelo proposto é a versatilidade e flexibilidade para avaliar diferentes cenários. Assim, para mostrar a aplicabilidade de nosso modelo proposto, apresentamos duas análises numéricas que exemplificam como nosso modelo pode ser utilizado de forma prática por um administrador de sistemas. Em nossos cenários, a arquitetura do MEC é composta de três servidores, e as máquinas disponíveis para avaliação possuem 8, 16 ou 32 núcleos.

Em (PREMSANKAR; TALEB, 2018), os autores avaliaram uma arquitetura MEC com um único dispositivo móvel como um cliente e *containeres* executando os serviços. Os autores avaliaram um jogo 3D chamado *Neverball*, onde o jogador deve inclinar o chão para controlar a bola, a fim de coletar moedas e chegar a um ponto de saída antes que o tempo se esgote.

Nós consideramos os parâmetros do sistema usados em (PREMSANKAR; TALEB, 2018) como parâmetros de entrada para nosso modelo. Portanto, nosso estudo desenvolve o trabalho em (PREMSANKAR; TALEB, 2018), realizando análises numéricas para avaliar os cenários considerando múltiplos parâmetros.

Consideramos um dos seus cenários com uma resolução de jogo de 800 *times* 600 pixels. O valor do parâmetro adotado correspondente ao atraso de processamento de uma solicitação é de 10.7ms. Todos os parâmetros utilizados são apresentados na Tabela 190.

Tabela 190 – Valores Atribuídos as Transições Temporizadas.

Transições Temporizadas	Tempo (ms)
T.A	8.3
T.B	5.2
T.C	2.5
P.A	10.7
P.B	10.7
P.C	10.7
ND	2.3

28.5.2.1 Estudo de Caso 1

No estudo de caso 1, planejamos uma arquitetura composta de três servidores MEC; e para isso, o objetivo é minimizar o MRT e alocar eficientemente tarefas em máquinas que não apresentam alta taxa de consumo de recursos. O objetivo é minimizar os recursos desdobrados, ao mesmo tempo em que atende a demanda solicitada, a fim de diminuir os investimentos financeiros.

Assumimos que o MRT desejado é de até 40ms, e a taxa de uso desejada deve estar entre 40% e 50% para evitar sobrecarga e ociosidade nos servidores. Nesse caso, a taxa de chegada do pedido é de 0,05 solicitações por milissegundos. A lista de todas as 27 configurações possíveis das máquinas é apresentada na Tabela 191.

A Figura 302 apresenta os resultados deste estudo de caso. A configuração 1 apresenta uma configuração com MRT igual a 995,82ms, muito acima das outras, por isso omitimos essa configuração para fornecer uma melhor visualização. As configurações que satisfazem os requisitos (MRT e taxas de utilização de recursos) são #27, #18, #24, #15, #9, #6, #26, #21, #12, #7, #23, #3, #14 e #8.

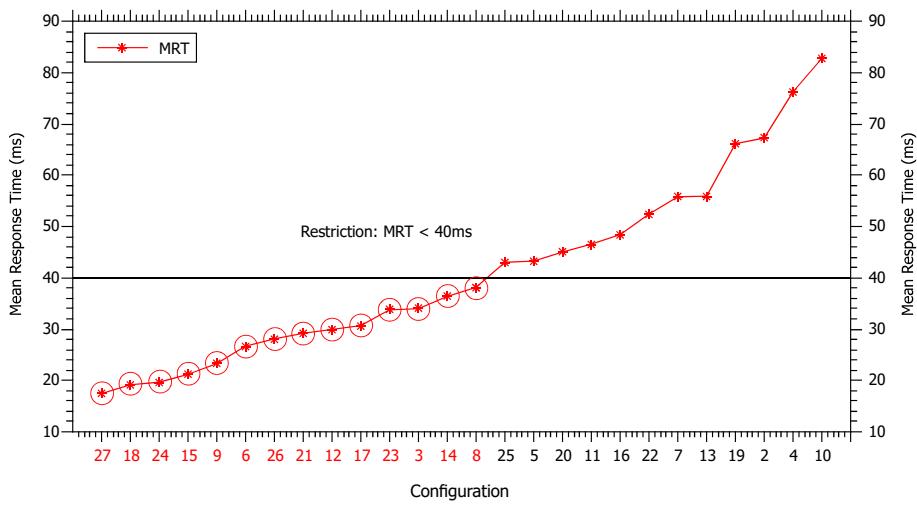
A melhor configuração para este caso é #27, que é composta dos melhores servidores disponíveis (todos com 32 núcleos), e a segunda melhor configuração (ServidorA = 16, ServidorB = 32, ServidorC = 32) é focada em dar mais poder de computação para servidores que estão longe do usuário.

A Figura 303 apresenta a utilização dos recursos das configurações das máquinas. Agora, as configurações que se ajustam ao requisito são #25, #14, #8, #3, #16, #20, #22 e #5. Nesse caso, a melhor solução (ServidorA = 32, ServidorB = 32, ServidorC = 8) é baseada em máquinas com mais poder de computação mais próximo dos usuários.

Então, analisando os resultados de MRT e utilização de recursos, pode-se notar que as configurações #14 (ServidorA = 16, ServidorB = 16, ServidorC = 16), #8 (ServidorA = 8, ServidorB = 32, ServidorC = 16), e #3 (ServidorA = 8, ServidorB = 8, ServidorC = 32) se encaixam nos dois requisitos, conforme apresentado na Tabela 192. Na configuração #14, a solução equilibra o poder computacional igualmente, mas na configuração

Tabela 191 – Configurações Possíveis para os Estudos de Caso 1 e 2.

Configuração	Servidor A	Servidor B	Servidor C
#1	8	8	8
#2	8	8	16
#3	8	8	32
#4	8	16	8
#5	8	16	16
#6	8	16	32
#7	8	32	8
#8	8	32	16
#9	8	32	32
#10	16	8	8
#11	16	8	16
#12	16	8	32
#13	16	16	8
#14	16	16	16
#15	16	16	32
#16	16	32	8
#17	16	32	16
#18	16	32	32
#19	32	8	8
#20	32	8	16
#21	32	8	32
#22	32	16	8
#23	32	16	16
#24	32	16	32
#25	32	32	8
#26	32	32	16
#27	32	32	32

Figura 302 – Resultados do Estudo de Caso 1 Considerando MRT \leq 40ms.

#8 e #3, as soluções são heterogêneas, alocando mais poder computacional em servidores mais distantes do usuário.

Tabela 192 – Configurações Propostas para o Estudo de Caso 1.

Configuração	MRT (ms)	Utilização Média (%)
#3	33,98	44,22
#8	38,14	43,13
#14	36,37	42,05

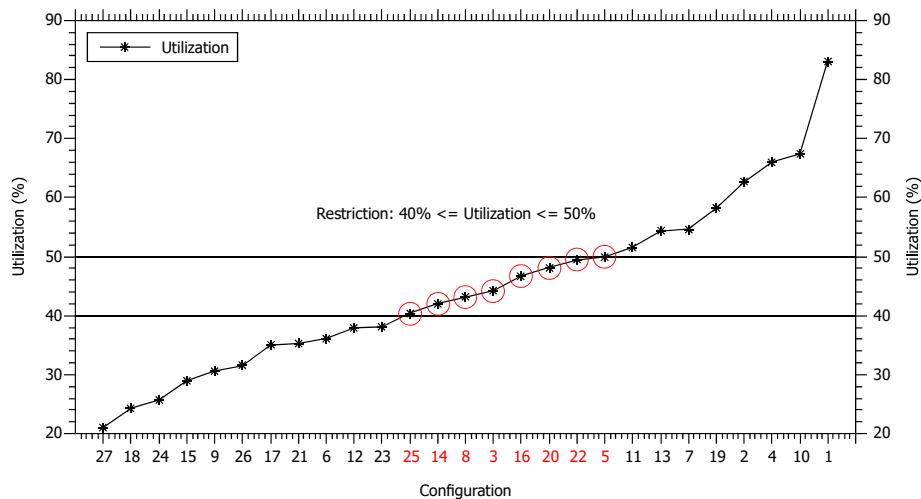


Figura 303 – Estudo de Caso 1, Considerando $40\% \leq \text{Utilização} \leq 50\%$.

28.5.2.2 Estudo de Caso 2

Neste estudo de caso, consideramos hipoteticamente que as três máquinas já foram escolhidas (8, 16 e 32 núcleos), o objetivo do estudo de caso 2 é minimizar o MRT e a utilização de recursos. A taxa de chegada do pedido é de 0,02 req/ms e a Tabela 193 apresenta todas as configurações possíveis para este estudo de caso.

Tabela 193 – Configurações Possíveis para o Estudo de Caso 2.

Configuração	Servidor A	Servidor B	Servidor C
#1	32	16	8
#2	32	8	16
#3	8	16	32
#4	8	32	16
#5	16	8	32
#6	16	32	8

As Figuras 304 e 305 apresentam o MRT e a utilização de recursos, respectivamente. Ambas as métricas apresentam a mesma ordem, mostrando que a configuração #3 (ServidorA = 8, ServidorB = 16, ServidorC = 32) produz o melhor MRT (21.08ms) e utilização de recursos (14.41%). Por outro lado, a configuração 1 (ServidorA = 32, ServidorB = 16, ServidorC = 8) obteve os piores resultados, tendo MRT igual a 44,5ms e utilização de recursos igual a 24,53%.

Nesse caso, podemos afirmar que, para minimizar o MRT e a utilização de recursos, é necessário ter um servidor com o maior poder computacional (32 núcleos) próximo ao Front End e o servidor com a menor capacidade distante do Front End..

Configurações #5 (ServidorA = 16, ServidorB = 8, ServidorC = 32) e #4 (ServidorA = 8, ServidorB = 32, ServidorC = 16) têm diferentes configurações de servidor, mas apresentam resultados semelhantes: configuração #5 apresentado MRT = 24,23ms e utilização de recursos = 16,13%; e a configuração #4 apresentou MRT = 25.77ms e utilização de recursos = 16.99%. No entanto, como a configuração #5 possui os melhores recursos no ServidorC (melhor latência e capacidade), essa configuração seria preferível à configuração #4.

28.6 Modelo SPN Extendido

Quando desenvolvedores de aplicativos e provedores de serviços desejam planejar e projetar um ambiente MEC, eles devem ser capazes de estimar o tempo médio de resposta do sistema. As Funções de Distribuição Cumulativa (CDFs) podem indicar esse momento pela máxima probabilidade de absorção.

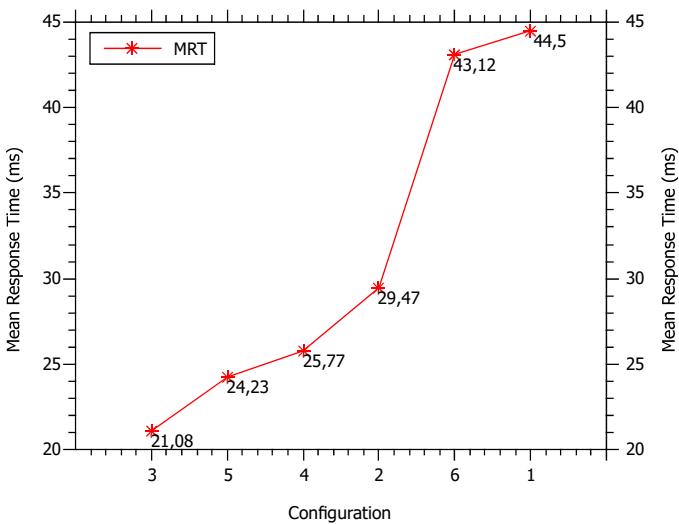


Figura 304 – MRT para as 6 Configurações

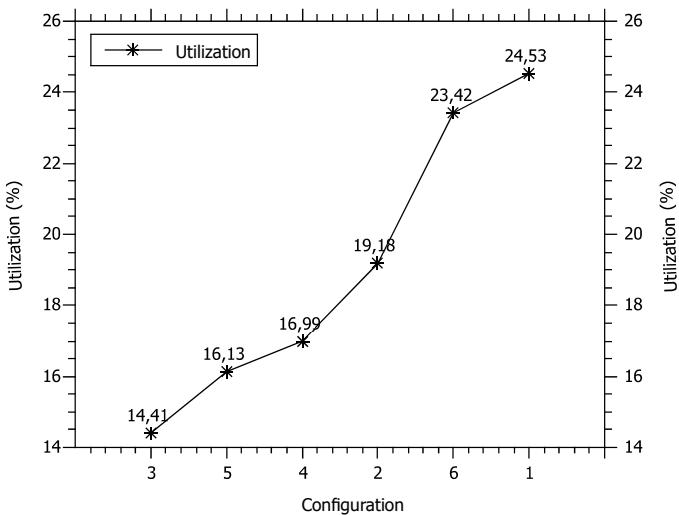


Figura 305 – Utilização para as 6 Configurações

As CDFs estão associadas a uma distribuição de probabilidade específica e podem ser obtidas através de avaliações transitórias, gerando probabilidades com tempo tendendo a t . Em outras palavras, pode-se calcular a probabilidade **absorção (AP)** entre $[0, t]$, através de avaliações transitórias, quando $F(t)$ se aproxima de 1. Portanto, é possível estimar a probabilidade de concluir a execução do aplicativo antes de um tempo específico $[P(T < t)]$.

Além disso, os CDFs também podem indicar a probabilidade de um aplicativo ser processado e concluído dentro de um determinado intervalo de tempo $[P(t_1 < T < t_2) = P(T < t_2) - P(T \leq t_1)]$.

Os CDFs podem ser obtidos resolvendo modelos SPN adaptando o modelo a um estado absorvente (SILVA et al., 2018). A Figura 306 mostra a adaptação que fizemos em nosso modelo SPN apresentado anteriormente para calcular os CDFs. Basicamente, foram feitas três alterações: (a) um local de estado absorvente (denominado Finish) foi adicionado na parte direita do modelo, indicando que quando as solicitações chegam a este local, tais solicitações não mudam de estado; (b) no bloco de Admissão, o ciclo de feedback ($ND \rightarrow P_{Arrival}$) foi retirado, indicando que novas solicitações não serão geradas de forma inconfundível; e (c) existe um novo parâmetro chamado BATCH (no local $P_{Arrival}$) que representa o

número de jobs (tokens) que serão processados.

O CDF calcula a probabilidade desses trabalhos concluir o processamento do aplicativo em um determinado momento e em um determinado intervalo de tempo.

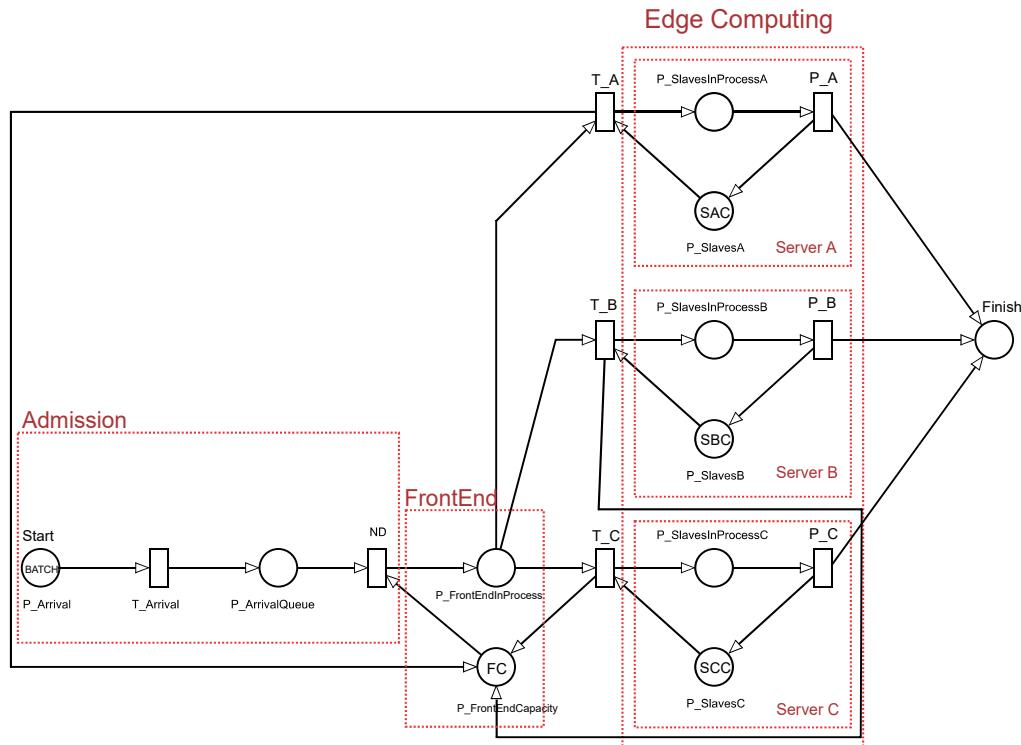


Figura 306 – SPN Usado para Calcular o CDF

28.6.1 Estudos de Caso

Esta seção apresenta dois estudos de caso para mostrar a aplicabilidade da métrica CDF. Para ambos os estudos, usamos um nível de confiança de 95% e uma taxa de erro relativa máxima de 10%. Para cada teste, geramos 300 amostras aplicando a simulação transitória.

28.6.1.1 Estudo de Caso 1

A Tabela 194 apresenta os parâmetros usados na simulação deste estudo de caso. Observe que definimos $AD = 8.0ms$ e configuramos a capacidade em todos os servidores para 32, tentando executar um BATCH de 100 solicitações. Apenas a capacidade do FrontEnd (FC) foi alterada. De fato, variamos a capacidade do FrontEnd em três valores: 50, 100 e 150. As probabilidades foram calculadas de $t = 0ms$ a $t = 1000ms$ para as três execuções.

Tabela 194 – Valores dos Parâmetros para o Estudo de Caso 1.

Execução	AD	FC	SAC	SBC	SCC	BATCH
Execução 1	8.0	50	32	32	32	100
Execução 2	8.0	100	32	32	32	100
Execução 3	8.0	150	32	32	32	100

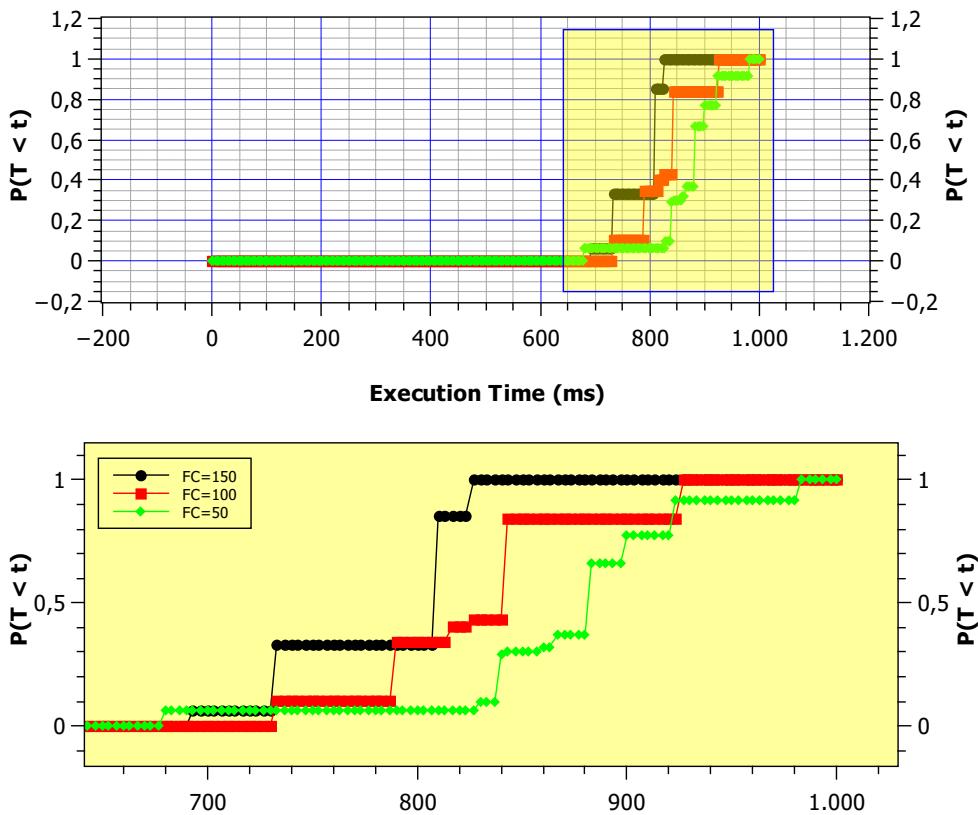


Figura 307 – CDF para BATCH=100 e FC = {50,100,150}

28.6.1.2 Probabilidade de Absorção (AP)

A Figura 307 representa a CDF deste estudo de caso. A partir dos resultados, podemos ver que quanto maior a capacidade FrontEnd, menor será o tempo necessário para que a probabilidade de absorção atinja 100%. O tempo necessário para atingir AP = 100% depende da capacidade do FrontEnd, quanto maior a capacidade, menor o tempo necessário.

Quando a capacidade FrontEnd é 50, o CDF variou de AP = 0% em 680ms e levou 303ms para alcançar AP = 100%, enquanto execuções com capacidades de 100 e 150 levaram apenas 190ms e 130ms, respectivamente. Considerando AP = 50%, quando a capacidade FrontEnd é 50, 100 e 150, o tempo associado foi de aproximadamente 881ms, 841ms e 808ms, respectivamente. Observe que, conforme esperado, a capacidade menor apresentou um tempo maior.

Ao considerar um tempo específico, sua probabilidade de absorção é a seguinte. Em 800ms, FC = 50 está próximo de 0%, enquanto FC = 100 e FC = 150 apresentam AP = 32%. Neste caso, como os resultados são tão próximos, sugere-se adotar a opção FC = 100, a fim de diminuir os custos de investimento financeiro.

28.6.1.3 Probabilidade de Absorção por Intervalo (API)

Para calcular a probabilidade de absorção por intervalos (API), basicamente deve-se considerar dois pontos da curva CDF e subtrair as probabilidades correspondentes. Como exemplo, na Figura 307, se considerarmos HR = 100 e os intervalos 840ms (50%) e 940ms (100%), a API para 100 solicitações nesse intervalo será de 100% – 50% = 50%. A Figura 308 apresenta as APIs levando em consideração dois intervalos de tempo, 700ms < T < 800ms e 800ms < T < 900ms.

No primeiro intervalo (700ms < T < 800ms), as APIs eram aproximadamente 0%, 34% e 27%, para FC = 50, FC = 100 e FC = 150, respectivamente. Nesse intervalo, a execução com FC = 50 apresentou API =

0%; isso ocorreu porque, durante todo o intervalo de tempo, a API foi de 6,45% e, como nenhum aumento ocorreu, a API não era maior que 0%. Por outro lado, as APIs quando $HR = 100$ (34%) e $HR = 150$ (27%) eram significativas. A API de $FC = 100$ é maior que $FC = 150$ porque, embora a 800ms, eles estão no mesmo nível, no ponto 700ms, a API de $FC = 150$ é ligeiramente superior a $FC = 100$.

No segundo intervalo de tempo ($800ms < T < 900ms$), as APIs são mais altas porque tendem a atingir 100%, assim como as curvas são mais acentuadas nesse intervalo. As APIs quando $FC = 50$, $FC = 100$ e $FC = 150$ eram 70%, 50% e 65%, respectivamente. A probabilidade quando $FC = 100$ é menor que os outros porque apresenta um aumento mais lento. Enquanto $FC = 100$ varia de 30% a 80%, $FC = 50$ varia de 10% a 80%.

Portanto, se alguém quiser que a execução do aplicativo de 100 solicitações termine entre 700ms a 800ms, a simulação sugere a adoção de $FC = 100$, ou seja, uma capacidade FrontEnd de 100 (por exemplo: 100 containers), porque se adotar $FC = 50$, a API é 0%.

Considerando o segundo intervalo, se um requerer que o tempo médio de execução fique entre 800ms e 900ms, a configuração $FC = 50$ retornará a maior probabilidade de conclusão. Este é um resultado muito interessante, porque ao adotar $FC = 50$ há uma economia de recursos, considerando as três possibilidades para a capacidade FrontEnd.

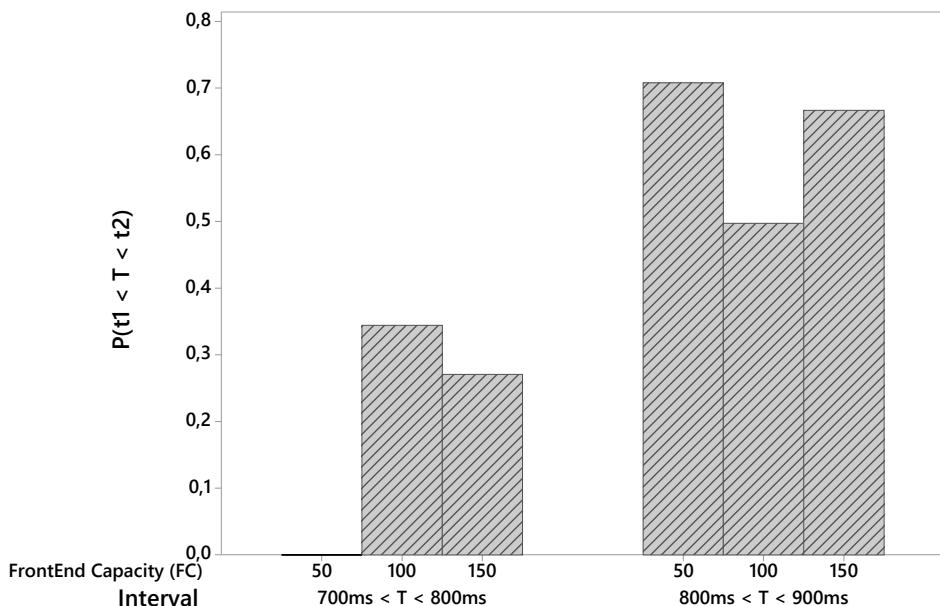


Figura 308 – Probabilidade de Absorção com Base em Dois Intervalos de Tempo para o Estudo de Caso 1

28.6.1.4 Estudo de Caso 2

Tabela 195 apresenta os parâmetros usados no estudo de caso 2. Observe que definimos $AD = 2.0ms$ e definimos a capacidade de todos os servidores para 64, para executar um BATCH de 1000 solicitações. Aumentamos o valor de BATCH de 100 para 1000 para ter uma carga de trabalho maior em comparação com o estudo de caso anterior. Nós variamos a capacidade do FrontEnd (marcação FC) em três valores: 250, 500 e 1000.

Tabela 195 – Valores dos Parâmetros para o Estudo de Caso 2.

Execução	AD	FC	SAC	SBC	SCC	BATCH
Execução 1	2.0	250	64	64	64	1000
Execução 2	2.0	500	64	64	64	1000
Execução 3	2.0	1000	64	64	64	1000

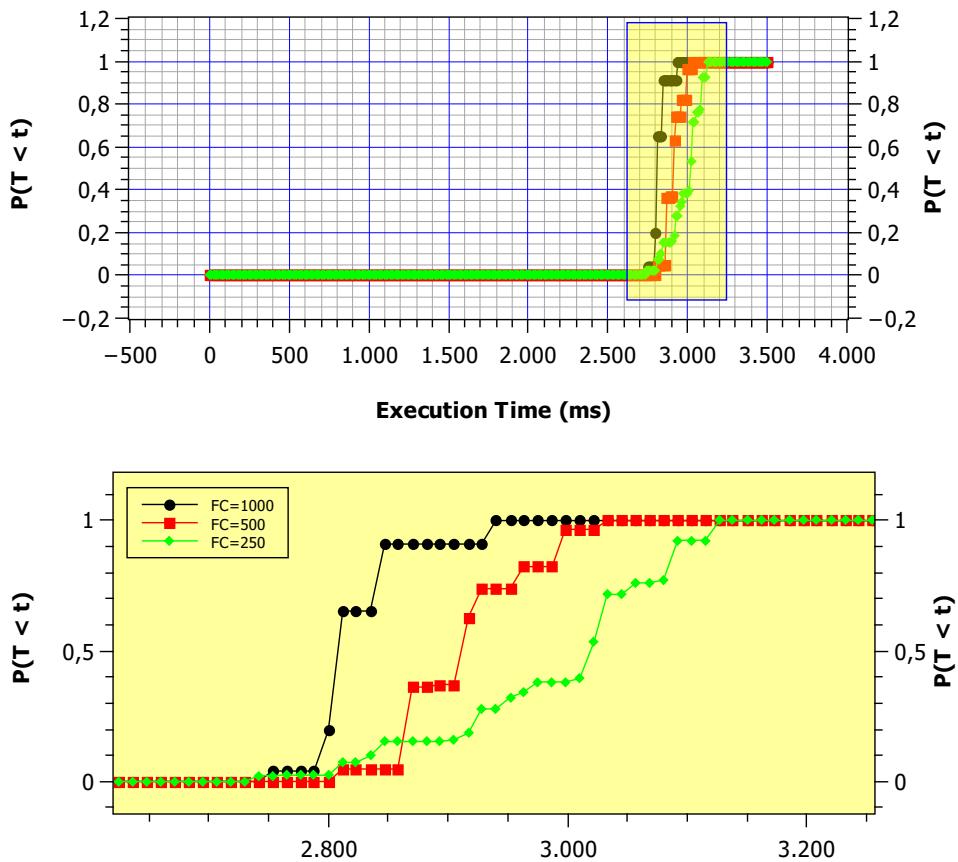


Figura 309 – CDF para BATCH=1000 e FC = {250,500,1000}

28.6.1.5 Probabilidade de Absorção (AP)

A Figura 309 exibe o CDF correspondente do estudo de caso 2. Os pontos de acesso foram calculados de $t = 0ms$ a $t = 3500ms$ para todos os experimentos. Como esperado, quanto maior a capacidade do FrontEnd, menos tempo é necessário para o AP atingir 100%. Considerando AP = 50%, os tempos médios de absorção associados às capacidades FC = 250, FC = 500 e FC = 1000 são aproximadamente 3020ms, 2915ms e 2810ms, respectivamente.

Em geral, FC = 1000 apresenta os melhores resultados quando comparado com os outros. O sistema atinge AP = 100% a 3127ms, 3033ms e 2940ms para as capacidades FrontEnd 250, 500 e 1000, respectivamente. Note que para 50% ou 100%, as diferenças nos tempos de absorção entre diferentes capacidades são baixas ($\text{cong}100\text{ms}$). No entanto, em sistemas críticos, 100 ms podem representar um tempo de execução inaceitável. Curiosamente, em 2850ms, o AP para FC = 250 (15%) é maior que FC = 500 (5%).

Portanto, se aceitarmos o tempo máximo de 2850 ms, é mais vantajoso ter FC = 250, o que significa uma economia significativa de recursos.

28.6.1.6 Probabilidade de Absorção por Intervalo (API)

A Figura 310 mostra as APIs que obedecem a dois intervalos de tempo: $(2800ms < T < 2900ms)$ e $(2900ms < T < 3000)$. No primeiro intervalo, há um padrão ascendente, em que quanto maior a capacidade FrontEnd, maiores são as APIs para 1000 solicitações.

Quando FC = 250, FC = 500 e FC = 1000, as APIs eram 12%, 37% e 70%, respectivamente. A API é bem diferente para esse primeiro intervalo, com uma variação de aproximadamente $\cong 28\%$ entre as três capacidades FrontEnd. Vale a pena mencionar que uma API = 70% é bastante significativa, mas alojar 1000 recursos pode ser muito caro.

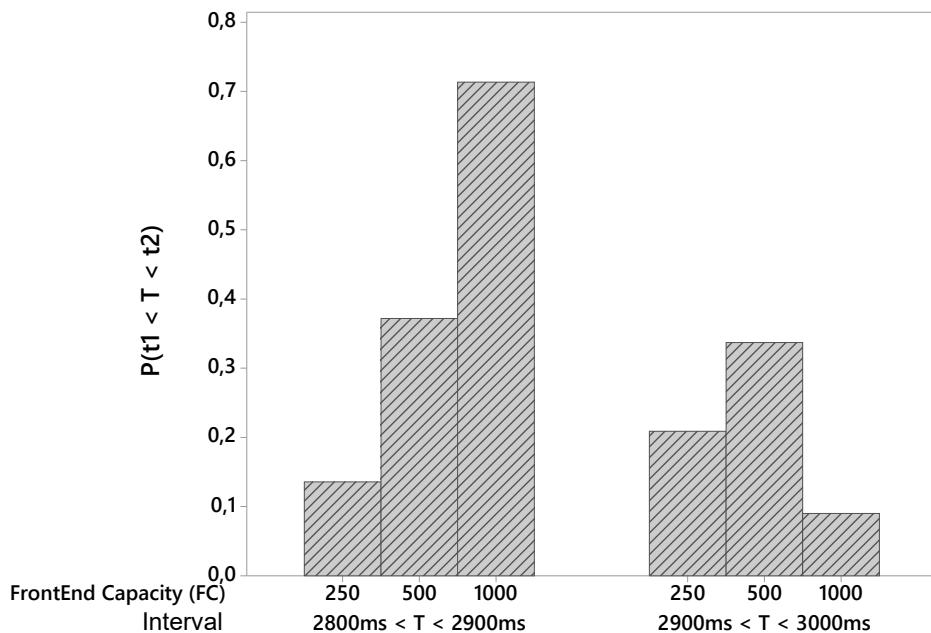


Figura 310 – Probabilidade de Absorção com Base em Dois Intervalos de Tempo para o Estudo de Caso 2

No segundo intervalo, ($2900ms < T < 3000$), há um padrão menos intuitivo, com $FC = 500$ (35%) superando os outros. Quando $FC = 500$, a API praticamente não mudou nos dois intervalos, significando que de 2800ms a 3000ms há uma certa constância (veja a Figura 310).

No segundo intervalo, quando $FC = 1000$, apresenta uma API menor que no primeiro intervalo, cerca de 9%. Interessante que mesmo com uma alta capacidade, a API é menor que as outras duas opções. Isso é verdade porque, a 2800ms, a API de $FC = 1000$ já é 90% e, em 2850ms, já atingiu 100%. Portanto, quando a API tende à probabilidade máxima, a API tende a zero.

28.7 Validação do Modelo SPN

Esta seção apresenta a validação do modelo SPN proposto. O objetivo desta validação foi comparar o MRT calculado pelo modelo e o MRT coletado por experimentos em um cenário real. A Figura 311 apresenta um esboço do experimento prático realizado. Assim como na arquitetura proposta neste trabalho, na validação simulamos a existência de três torres com um servidor cada. A capacidade que adotamos para todos os servidores foi de 4 núcleos, onde cada um deles roda um *container* diferente. Cada requisição é processada por apenas um *container*, portanto, se chegar 4 requisições em uma torre, a mesma irá distribuí-las entre todos os *containers*. Para realizar a validação, foi desenvolvido um sistema sintético que simula o envio de requisições obedecendo uma distribuição exponencial.

A Figura 312 apresenta um diagrama de sequência referente ao experimento, com os passos para coletar os dados. As chamadas destacadas em azul se referem às funções principais referentes às requisições. As demais chamadas representam a coleta dos *timestamps* a um servidor local que atua como relógio global. Os tempos são coletados no início e fim de cada tarefa realizada, com o objetivo de saber a duração das etapas. Quando o arquivo de entrada é enviado, a máquina requisita ao servidor do relógio global o *timestamp* e salva em um log, quando o arquivo é recebido em outra máquina, esta realiza a mesma ação. Com esses dois *timestamps* obtemos o tempo necessário para envio de uma requisição. Os tempos obtidos com esse algoritmo foram utilizados para alimentar as transições do modelo SPN estendido, apresentado na seção 28.6. A partir desses valores, as transições T_A , T_B e T_C foram preenchidas com os tempos de envio do FrontEnd para as torres A, B e C respectivamente. Nas transições P_A , P_B e P_C , adicionamos os tempos de processamento para cada servidor. A Tabela 196 apresenta a configuração dos computadores utilizados para

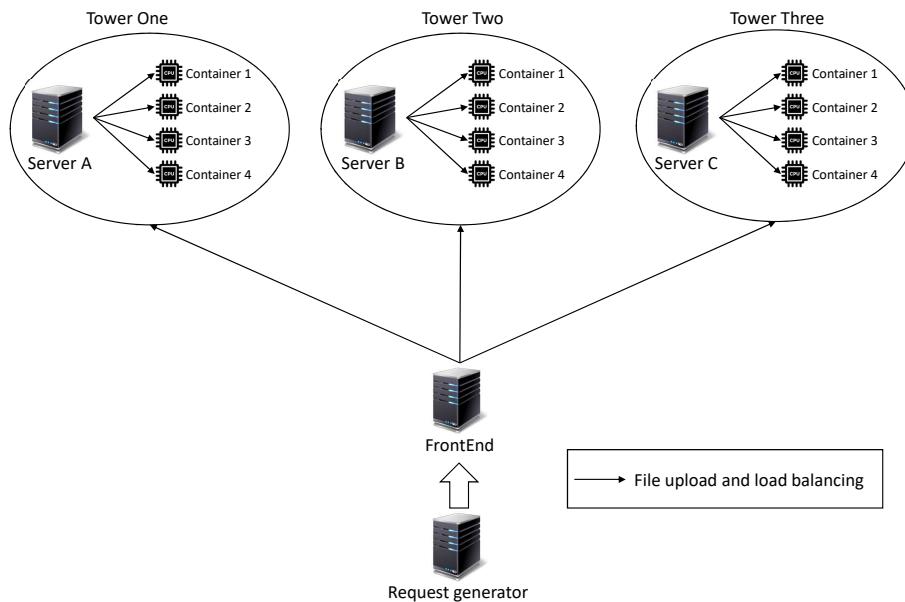


Figura 311 – Esboço Prático da Experiência

realizar a validação juntamente com as suas respectivas configurações e funções.

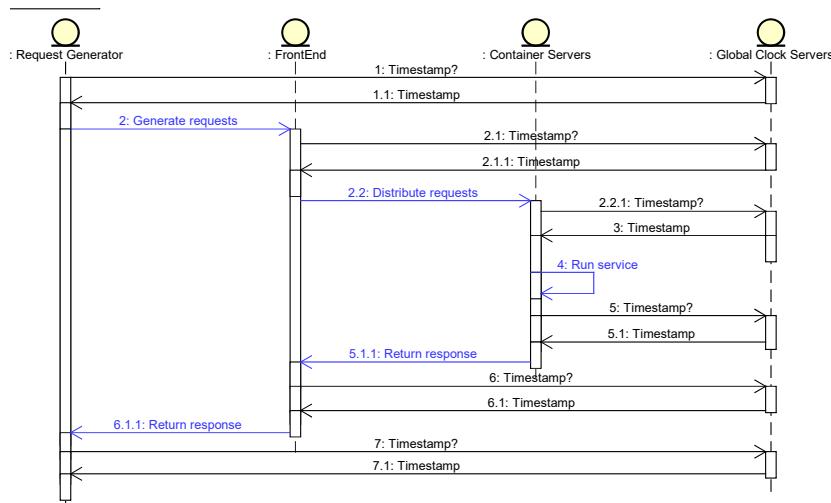


Figura 312 – Diagrama de Sequência do Experimento Prático

Tabela 196 – Máquinas Utilizadas no Experimento Prático

Configuração do Computador	Perfil	Função
Intel Core i3 1.7Ghz 8GB	Gerador de requisições	Gera solicitações para serem processadas nos servidores
Intel Core i3 3.7Ghz 4GB	FrontEnd	Recebe solicitações e encaminha para servidores
Intel Core i3 3.7Ghz 4GB	Servidor A	Recebe as solicitações e as processa em seus núcleos
Intel Core i3 3.7Ghz 8GB	Servidor B	Recebe as solicitações e as processa em seus núcleos
Intel Core i3 3.7Ghz 8GB	Servidor C	Recebe as solicitações e as processa em seus núcleos

Cada requisição utilizada no experimento prático consiste em um arquivo contendo uma matriz de tamanho 10000x10000 preenchida com valores aleatórios, onde o mesmo é enviado para um dos servidores para ser ordenado e retornado para o gerador de requisições. O gerador de requisições simula uma

quantidade x de clientes que enviam as solicitações ao mesmo tempo, o FrontEnd recebe todos de forma simultânea e os escalona entre as torres, essas por sua vez as escalona entre os servidores existentes. Com o objetivo de tornar o resultado mais confiável, para cada configuração dos servidores repetimos 15 vezes a execução do algoritmo. A Tabela 197 apresenta o MRT e as configurações das máquinas para as execuções realizadas, enquanto a Tabela 198 apresenta os MRTs obtidos nas execuções do modelo.

Tabela 197 – Tempos Obtidos no Experimento Prático

Id da Execução	MRT (ms)		
	3 Requisições	6 Requisições	9 Requisições
1	1380	2892	4395
2	1105	3302	3621
3	1376	2803	3830
4	1221	2139	4394
5	1013	2336	4819
6	1162	2081	4526
7	1098	2378	4412
8	1107	2725	3937
9	1015	2274	4567
10	1063	2263	4434
11	1008	2561	3928
12	995	2178	4370
13	1090	2496	4685
14	1068	2068	4418
15	1046	2071	4633

Tabela 198 – Tempos Obtidos no Modelo SPN

Número de Requisições	MRT (ms)
3	1103
6	2504
9	4457.4

Note que quanto mais requisições forem enviadas, maior será o MRT. O algoritmo apresentou-se estável, pois os resultados não apresentaram valores muito distantes entre si. O Teste T de uma amostra ² é usado para fazer inferências sobre a média da população, com base nos dados de uma amostra aleatória. Todas amostras utilizadas foram normais. Adotamos o Teste T de uma amostra para comparar o MRT gerado pelo modelo com a média do MRT obtido nas execuções. Para verificar se o Teste T é significante utilizamos o valor p, sendo ele uma probabilidade que mede a evidência contra a hipótese nula. Um valor de p menor fornece uma evidência mais forte contra a hipótese nula, sendo ela o MRT obtido na execução do modelo SPN, pois é o valor que queremos comparar. Geralmente, um nível de significância de 0,05 funciona bem, pois indica que o risco de se concluir que existe uma diferença, quando, na verdade, não existe nenhuma diferença real, é de 5%. A Tabela 199 apresenta os resultados obtidos com esse teste.

Tabela 199 – Resultados Obtidos com o Teste T de Uma Amostra

Número de Requisições	Quantidade da Amostra	Média (ms)	Desvio Padrão	Valor P
3	15	1116.5	122.1	0.676
6	15	2437.8	360	0.488
9	15	4398	415	0.59

Observe que o desvio padrão aumenta à medida em que há uma maior quantidade de requisições, isso ocorre porque além do algoritmo ter mais requisições para processar, ainda terá um trabalho maior para escaloná-las. Observe também que em todos os casos, o valor p é maior que 0.05 e, portanto, não podemos refutar a hipótese nula em todos os casos com 95% de confiança. Os resultados gerados pelo modelo são equivalentes estatisticamente ao experimento. As Figuras 313, 314 e 315 apresentam os gráficos boxplot referentes aos valores p de cada cenário.

Pode-se concluir que há indícios que o modelo é confiável e preciso dada a similaridade dos resultados obtidos com o experimento real. Vale observar também que o valor do modelo está dentro da margem de erro do experimento. O modelo reflete o ambiente real e é útil para o planejamento de arquiteturas MEC.

² Teste T de uma amostra <https://tinyurl.com/yanthw4e>

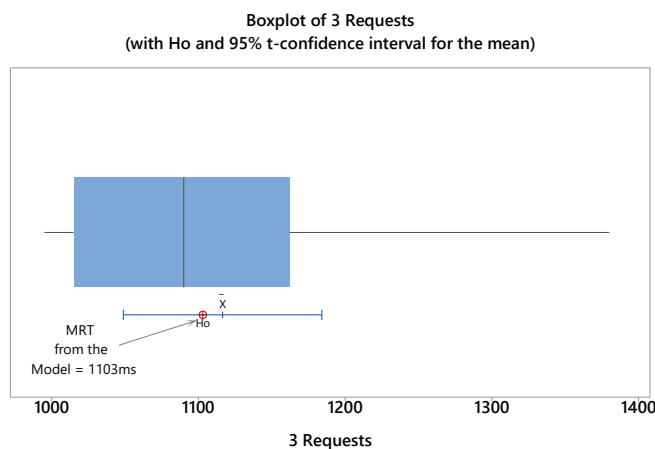


Figura 313 – Boxplot das 3 Solicitações

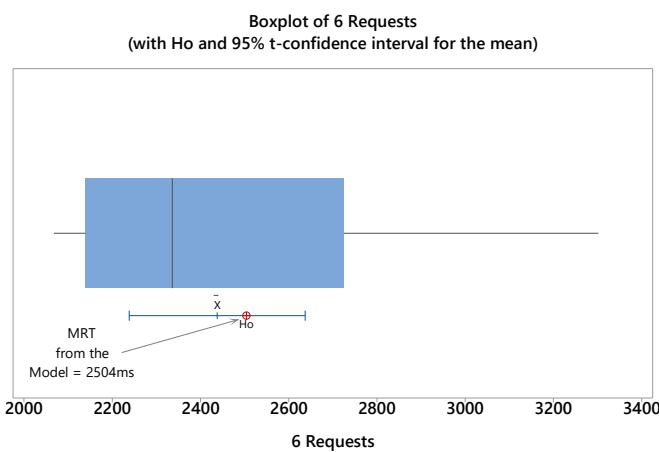


Figura 314 – Boxplot das 6 Solicitações

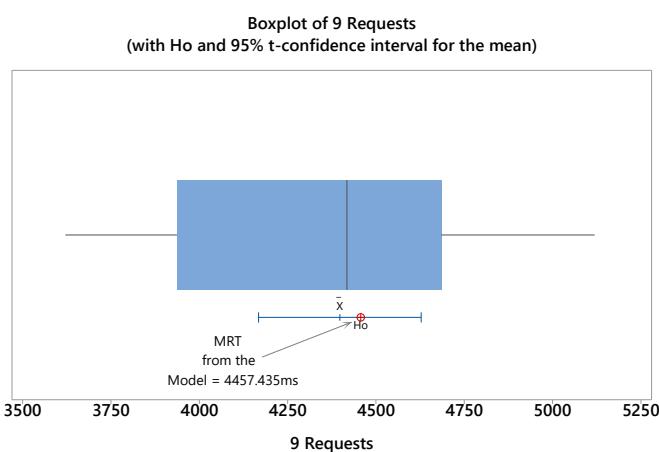


Figura 315 – Boxplot das 9 Solicitações

28.8 Conclusões e Trabalhos Futuros

Este trabalho propôs um modelo SPN para representar e avaliar o desempenho de uma arquitetura MEC composta por servidores n . Nosso modelo permite estimar o tempo médio de resposta (MRT) e o nível de utilização de recursos na borda da rede. Um dos aspectos mais importantes do nosso modelo é a sua flexibilidade de avaliação devido ao alto número de parâmetros que podem ser configurados pelo usuário (15 parâmetros), tornando os testes mais efetivos e decisivos. Para demonstrar a aplicabilidade do modelo proposto, foram realizadas análises numéricas com dados reais coletados de um trabalho de referência. Através das análises numéricas, pudemos observar o comportamento do MRT e a utilização dos recursos, demonstrando a utilidade do modelo na escolha da melhor maneira de implementar uma arquitetura MEC.

Como trabalho futuro, planejamos estender o modelo de SPN para incluir métricas de disponibilidade e medir energia. Dado o modelo estendido, planejamos realizar novas análises numéricas usando cenários heterogêneos.

Avaliação de Desempenho e Disponibilidade de uma Arquitetura de Hospital Inteligente

Autores: Laécio Andrade Rodrigues e Francisco Airton Silva

Resumo

Baixa latência e alta disponibilidade de recursos são características essenciais para garantir a qualidade dos serviços nos sistemas de saúde. Os sistemas hospitalares devem ser eficientes para evitar perdas de vidas humanas. Os Hospitais Inteligentes prometem uma revolução na saúde ao capturar e transmitir dados do paciente aos médicos em tempo real por meio de uma rede de sensores sem fio. No entanto, existe uma dificuldade significativa na avaliação do desempenho e disponibilidade de tais sistemas em contextos reais, devido a falhas não serem toleradas e elevados custos de implementação. Este trabalho adota modelos analíticos para avaliar o desempenho e a disponibilidade de sistemas hospitalares inteligentes sem ter que investir em equipamentos reais com antecedência. Dois modelos de redes de Petri estocásticas foram propostos para representar arquiteturas hospitalares inteligentes, sendo um usado para avaliar o desempenho e outro para avaliar a disponibilidade. Os modelos são bastante paramétricos, permitindo calibrar a capacidade dos recursos, tempos de atendimento, tempos entre falhas e tempos entre reparos. O modelo de desempenho, por exemplo, permite definir 13 parâmetros. O modelo de disponibilidade, permite definir 48 parâmetros. Isto é, ele permite avaliar um grande número de cenários. Além disso, este trabalho apresenta uma análise de sensibilidade que identifica os componentes mais críticos da arquitetura. Portanto, este trabalho tem o potencial de ajudar administradores de sistemas hospitalares a planejarem arquiteturas mais otimizadas de acordo com suas necessidades.

Palavras-chaves: Hospital Inteligente, Redes de Petri Estocásticas, Desempenho, Disponibilidade, Internet das Coisas.

29.1 Introdução

A Internet das Coisas (IoT) está inovando em todas as áreas da nossa sociedade, conectando milhões de aparelhos e tornando o dia a dia das pessoas mais fácil. O setor de saúde também pode se beneficiar da IoT. O uso de dispositivos e sensores inteligentes pode revolucionar a saúde das pessoas em qualquer ambiente, como em casa ou na rua. Em um hospital não é diferente, sinais vitais de pacientes captados por sensores podem ser transmitidos para Unidades de Terapia Intensiva (ICU), taxa de ocupação de leitos, índice de produtividade dos funcionários, entre outros. Portanto, existem muitas possibilidades da IoT na área da saúde (MAIA, 2017). A crise de saúde causada pela pandemia com a COVID-19 destacou a fragilidade dos sistemas de saúde em todo o mundo e a necessidade de otimizar o uso dos recursos existentes (escassos). Para se preparar para uma pandemia como a que estamos enfrentando, os hospitais precisam de uma estratégia para gerenciar seu espaço, equipe e suprimentos a fim de fornecer o melhor atendimento aos pacientes (??).

A IoT na saúde tem se destacado como uma tendência que permite a criação de novos tratamentos, mais precisos e com mais riqueza de informações (RAHMANI et al., 2015; RAHMANI et al., 2018; ZHANG et al., 2018). Com esses tratamentos, é possível obter uma transmissão permanente de dados fornecendo um quadro completo da doença do paciente. A agilidade dos serviços permite que os médicos possam responder imediatamente a qualquer alteração no quadro do paciente, visto que eles contam com informações contínuas em tempo real. As informações são capturadas não apenas por um, mas por uma rede de sensores sem fio (WSN). A WSN integra uma série de sensores autônomos espacialmente distribuídos em uma rede e transmite seus dados de forma cooperativa por meio de comunicação sem fio. Esses sensores (ver Figura 29.1) captam os dados do paciente e os encaminham por meio de comunicação sem fio para um *Gateway* que é responsável pelo processamento dos dados. Posteriormente, o *gateway* pode rotear os dados para servidores distribuídos, locais (computação de borda) ou em nuvem (RAHMANI et al., 2018). A computação de borda é uma arquitetura distribuída que apresenta poder de processamento descentralizado, permitindo computação móvel e tecnologias da IoT. Na computação de borda, os dados são processados por dispositivos próximos ao usuário final (computador ou servidor local) em vez de serem transmitidos para a nuvem (BUTLER, 2017; SHI et al., 2016a), o que é bastante atraente considerando hospitais inteligentes.

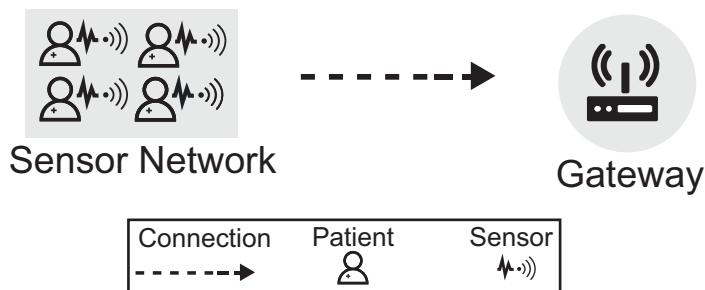


Figura 316 – Conexão entre sensores e gateway.

As redes de sensores são responsáveis por monitorar pacientes e notificar médicos em casos de emergência. No entanto, esses sistemas não podem enfrentar falhas de solicitações ou alta latência, pois podem comprometer a saúde de uma pessoa específica. Além de ser caro/demorado realizar experimentos e nem sempre se tem uma infraestrutura disponível. Redes de Petri estocásticas (SPNs) são modelos analíticos que podem representar sistemas complexos com diversas características, incluindo paralelismo e simultaneidade. As SPNs possuem fundamentos probabilísticos e com resultados bastante confiáveis se bem aplicados. Este trabalho se propõe a avaliar, por meio de modelos SPN, o uso de recursos, o tempo médio de resposta (MRT), a disponibilidade e o tempo de inatividade de uma arquitetura de um hospital inteligente, levando à otimização. Em resumo, as principais contribuições deste trabalho são:

1. Um modelo SPN para avaliação de desempenho de hospitais inteligentes.
2. Uma validação do modelo SPN para avaliação de desempenho.
3. Um modelo SPN para avaliar a disponibilidade de hospitais inteligentes.
4. Estudos de caso usando os modelos SPN propostos que possam servir como guias para os administradores de sistema planejarem infraestruturas hospitalares inteligentes.

29.1.1 Definição do Problema

Os hospitais inteligentes são a solução para a fragilidade dos sistemas de saúde que se pode observar com o COVID-19. No entanto, esses sistemas não podem enfrentar falhas de solicitações ou alta latência, pois podem comprometer a saúde de uma pessoa específica.

29.1.2 Objetivos

29.1.2.1 Objetivo Geral

O objetivo geral deste trabalho é avaliar o desempenho e a disponibilidade de uma arquitetura de um Hospital Inteligente sob a perspectiva de planejamento de arquiteturas mais eficientes, de modo a auxiliar na alocação dos recursos a fim de obter o melhor desempenho e alta disponibilidade.

29.1.2.2 Objetivos Específicos

1. Diminuir o tempo médio de resposta.
2. Reduzir o nível de utilização de recursos, evitando ociosidade e a perda (descarte) de dados.
3. Auxiliar administradores no planejamento de arquiteturas mais otimizadas.

29.1.3 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma: O capítulo 29.2 apresenta alguns conceitos para a compreensão deste trabalho; O capítulo 29.3 discute os principais trabalhos relacionados; O capítulo 29.4 apresenta a arquitetura de um hospital inteligente que foi usado como base para a criação dos modelos; O capítulo 29.5 apresenta um modelo SPN para avaliação de desempenho de hospitais inteligentes; O capítulo 29.6 apresenta um modelo SPN para avaliar a disponibilidade de hospitais inteligentes; O capítulo 29.7 traça algumas conclusões e trabalhos futuros.

29.2 Background

Este capítulo apresenta os principais conceitos necessários para melhor entendimento deste trabalho. Os seguintes tópicos são discutidos: IoT, hospitais inteligentes, computação em nuvem, computação de borda e Redes de Petri estocásticas.

29.2.1 IoT e Hospitais Inteligentes

A IoT está inovando todas as áreas da nossa sociedade e descreve um cenário no qual diversos objetos estão conectados e se comunicando. A IoT é formada por uma rede de objetos físicos capazes de coletar e transmitir dados. Essa inovação tecnológica tem como objetivo principal conectar os objetos que utilizamos no dia-a-dia à Internet, aproximando o mundo físico do digital. A conexão desses objetos busca trazer mais comodidade e agilidade no processamento das informações. Esses objetos podem variar de *smartwatches* (NGU et al., 2018) a geladeiras (TORRALBO-MUÑOZ et al., 2018) que solicitam produtos automaticamente do supermercado.

A IoT ainda tem um longo caminho a evoluir, mas a principal força da IoT é o alto impacto que terá em vários aspectos da vida cotidiana e no comportamento de usuários em potencial, não apenas na saúde. Para o usuário privado, os efeitos mais aparentes da introdução serão visíveis tanto no campo de trabalho quanto no doméstico. Neste contexto, vida assistida, e-saúde e aprendizagem aprimorada são apenas alguns exemplos de possíveis cenários de aplicação em que o novo paradigma em breve terá um papel de liderança. Sob a perspectiva dos negócios, as consequências mais aparentes serão igualmente visíveis em áreas como automação e manufatura industrial, logística, gestão de negócios/processos, transporte inteligente de pessoas e mercadorias (ATZORI; IERA; MORABITO, 2010).

Na área de *eHealth*, existem vários exemplos de conexão de dispositivos médicos na Internet para realizar diferentes serviços médicos remotos, como monitoramento remoto de pacientes, supervisão de idosos, assistência médica on-line, consultas, ou mesmo controle de braço robótico para intervenções cirúrgicas (SEBESTYEN; SAPLACAN; KRUCZ, 2010; YANG; YANG; PLOTNICK, 2013). Um verdadeiro impulso nesta área foi dado pela mais recente disseminação de aparelhos e dispositivos médicos portáteis miniaturizados. Esses dispositivos podem ser usados para medição contínua de parâmetros médicos (ex.: ECG, pressão arterial, temperatura), bem como reconhecimento e monitoramento de atividades ou avaliações médicas feitas remotamente (SEBESTYEN; TIREA; ALBERT, 2012). Existem muitas soluções particulares presentes no mercado, mas com acessibilidades muito restritas.

Hospital Inteligente é o conceito usado para se referir a hospitais digitalizados que contam com processos automatizados e otimizados baseados em IoT. Esses dispositivos servem para melhorar os procedimentos existentes de atendimento ao paciente, além de introduzir novos serviços. Dentre essas capacidades, podemos destacar o monitoramento contínuo dos pacientes. Os objetivos de um hospital inteligente incluem fornecer atendimento aprimorado ao paciente, atendimento médico remoto e melhorar os recursos de diagnóstico para garantir a segurança do paciente (GONTZES, 2017). Os Hospitais Inteligentes permitem a criação de novos tratamentos, os quais podem ser precisos e com maior riqueza de informações. Com hospitais inteligentes, será possível fornecer um quadro completo da doença do paciente, permitindo que os médicos respondam prontamente a qualquer mudança.

29.2.2 Cloud e Edge Computing

A computação em nuvem é um paradigma em contínua evolução que se originou da combinação de várias tecnologias diferentes. Esse paradigma é como “um tipo de sistema paralelo e distribuído que consiste em uma coleção de computadores interconectados e virtualizados que são provisionados dinamicamente e apresentados como um ou mais recursos de computação unificados com base em acordos de nível de serviço estabelecidos por meio de negociação entre o provedor de serviços e os consumidores” (MELL; GRANCE et al., 2011; BUYYA; YEO; VENUGOPAL, 2008).

O princípio básico da computação em nuvem é atribuir a computação a um grande número de computadores distribuídos, em vez de serviços locais ou remotos. Também são caracterizados pela utilização eficiente de recursos, empregando virtualização, monitoramento de recursos e mecanismos de平衡amento de carga (SARANYA; VIJAYALAKSHMI, 2011). Nas sociedades modernas, a maioria dos serviços essenciais são disponibilizados de forma transparente. O abastecimento de água, energia elétrica, gás e telefone, bens essenciais em nosso dia a dia, têm essa característica. Esses modelos de mercado seguem o conceito de pagar pelo que usar: o valor pago pelo serviço é flexível de acordo com a necessidade da organização a qualquer momento (GOMES, 2012). A computação em nuvem fornece um modelo de pagamento semelhante para a utilização de serviços de computação.

Existem três modelos de implementação de Cloud Computing (HUANG et al., 2010). Uma nuvem privada é uma infraestrutura em nuvem fornecida para uso exclusivo por uma única organização composta por vários consumidores. No modelo de nuvem pública, a infraestrutura de nuvem é fornecida para uso aberto pelo público em geral, que permanecem como entidades únicas, mas são unidas por tecnologia padronizada ou proprietária que permite a portabilidade de dados e aplicativos. O modelo de nuvem híbrida é composto de duas ou mais infraestruturas de nuvem distintas (privada, comunidade ou pública). No entanto, as nuvens híbridas apresentam complexidade adicional, a distribuição de aplicações por ambos os modelos (WANG et al., 2013). Resumidamente, entre os benefícios associados à utilização dos serviços

em nuvem, podemos destacar a gestão centralizada, a redução do consumo energético e a diminuição dos custos inerentes à manutenção de infraestruturas tradicionais. A nuvem oferece uma diversidade de serviços que favorece a agilidade do mercado (MILLER, 2008).

Trabalhos anteriores, como micro *datacenter* (GREENBERG et al., 2008), *cloudlet* (CUERVO et al., 2010; SILVA et al., 2017) e *fog computing* (MUTLAG et al., 2019), foram apresentados à comunidade porque a computação em nuvem nem sempre é eficiente para o processamento de dados devido aos elevados *delays* por conta da distância.

Colocar todas as tarefas de computação na nuvem provou ser uma forma eficiente de processamento de dados. O poder de computação na nuvem geralmente supera a capacidade de processamento na borda. Com a crescente quantidade de dados gerados na borda, a velocidade do transporte de dados está se tornando o gargalo para o paradigma da computação baseada em nuvem. Por exemplo, cerca de 5 Gigabytes de dados serão gerados por um Boeing 787 a cada segundo (XIA et al., 2019), mas a distância entre o avião e o satélite ou a estação base no solo causa um *delay* na transmissão de dados. Considere um veículo autônomo como outro exemplo. Dados de um Gigabyte serão gerados pelo carro a cada segundo, e isso requer processamento em tempo real para que o veículo tome decisões corretas (HOLUBČÍK et al., 2019). Se todos os dados precisassem ser enviados para a nuvem para processamento, o tempo de resposta seria muito longo. Nesse caso, os dados precisam ser processados na borda para um tempo de resposta mais curto, processamento mais eficiente e menor utilização da rede.

Portanto, a computação de borda refere-se às tecnologias que permitem que a computação seja realizada na borda da rede, em dados *downstream* para serviços em nuvem e dados *upstream* para serviços IoT. Aqui, definimos borda como quaisquer recursos de computação e rede ao longo do caminho entre as fontes de dados e os data centers em nuvem. Por exemplo, um *smartphone* é uma fronteira entre o corpo e a nuvem, um *gateway* em uma casa inteligente é a fronteira entre as coisas domésticas e a nuvem, um micro *data center* e um *cloudlet* é uma fronteira entre um dispositivo móvel e a nuvem (SHI et al., 2016b).

29.2.3 Rede de Petri Estocástica

Redes de Petri (MURATA, 1989b) são uma família de formalismos muito adequados para modelar vários tipos de sistema, uma vez que a simultaneidade, sincronização e mecanismos de comunicação são representados naturalmente. Este trabalho adota uma extensão particular, nomeada, Rede de Petri Estocástica (MARSAN, 1995), que permite a associação de atrasos estocásticos a transições temporizadas, e o respectivo espaço de estado pode ser convertido em Cadeia de Markov de Tempo Contínuo (CTMC) (TRIVEDI, 2002). Os modelos SPN apresentam uma base matemática sólida e são adequados para representar e analisar sistemas paralelos com vários componentes e que exibem aspectos de simultaneidade e sincronização. Portanto, esse formalismo representa uma escolha importante para modelar sistemas de computação em nuvem. Em SPNs, os locais são representados por círculos, enquanto as transições são representadas como retângulos preenchidos (transições imediatas) ou retângulos vazios (transições temporizadas).

Arcos (arestas direcionadas) conectam lugares a transições e vice-versa. Os tokens (pequenos círculos preenchidos) podem residir em locais que denotam o estado (ou seja, marcação) de uma SPN. Um arco inibidor é um tipo de arco específico que representa um pequeno círculo branco em uma borda, em vez de uma seta, e geralmente são usados para desativar as transições se houver tokens presentes em um local. O comportamento de uma SPN é definido em termos de um fluxo de token. Os tokens são criados e destruídos de acordo com os disparos de transição. As transições imediatas representam atividades instantâneas e têm maior prioridade de disparo do que as transições temporizadas. Além disso, tais transições podem conter uma condição de guarda, e um usuário pode especificar uma prioridade de disparo diferente, entre outras transições imediatas. As SPNs também permitem a adoção de técnicas de simulação para obtenção de métricas do sistema como alternativa à geração de um CTMC. A definição de rede de Petri estocástica estendida adotada neste trabalho é:

Seja $\mathcal{N} = (P, T, I, O, H, \Pi, M_0, Atts)$ uma rede de Petri estocástica (SPN), onde:

- $P = \{p_1, p_2, \dots, p_n\}$ é o conjunto de lugares, que pode conter tokens e formar as variáveis de estado discretas de uma rede de Petri. $ordp_{\mathcal{N}}$ corresponde a uma função bijetiva ($ordp_{\mathcal{N}} : P \rightarrow \{1, 2, \dots, n\}$)

que mapeia cada lugar para um número natural único.

- $T = \{t_1, t_2, \dots, t_m\}$ é o conjunto de transições, que modelam os componentes ativos. $ordt_{\mathcal{N}}$ é uma função bijetiva ($ordt_{\mathcal{N}} : T \rightarrow \{1, 2, \dots, m\}$) que mapeia cada transição para um número natural único.
- $I \in (\mathbb{N}^n \rightarrow \mathbb{N})^{n \times m}$ é uma matriz de multiplicidades de arcos de entrada dependentes de marcação, onde $I[ordp_{\mathcal{N}}(p_j), ordt_{\mathcal{N}}(t_k)]$ dá a multiplicidade de arco (possivelmente dependente da marcação) de arcos de entrada do local p_j para transição t_k [$A \subseteq (P \times T) \cup (T \times P)$ — conjunto de arcos]. Uma transição só é habilitada se houver tokens suficientes em todos os locais de entrada.
- $O \in (\mathbb{N}^n \rightarrow \mathbb{N})^{n \times m}$ é uma matriz de marcação de multiplicidades dependentes de arcos de saída, onde $O[ordp_{\mathcal{N}}(p_k), ordt_{\mathcal{N}}(t_j)]$ especifica a multiplicidade de arco possivelmente dependente da marcação de arcos de saída da transição t_j para o local p_k . Quando uma transição é disparada, ela remove o número de tokens especificados pelos arcos de entrada dos locais de entrada e adiciona o número de tokens fornecidos pelos arcos de saída a todos os locais de saída.
- $H \in (\mathbb{N}^n \rightarrow \mathbb{N})^{n \times m}$ é uma matriz de multiplicidades dependentes de marcação que descreve os arcos do inibidor, onde $H[ordp_{\mathcal{N}}(p_j), ordt_{\mathcal{N}}(t_k)]$ retorna a multiplicidade de arco possivelmente dependente da marcação de um arco inibidor do local p_j para transição t_k . Na presença de um arco inibidor, uma transição é ativada para disparar apenas se cada lugar conectado por um arco inibidor contiver menos tokens do que a multiplicidade do arco.
- $\Pi \in \mathbb{N}^m$ é um vetor que atribui um nível de prioridade a cada transição. Sempre que houver várias transições disparáveis em um determinado momento, aquela com a prioridade mais alta dispara primeiro e leva a uma mudança de estado.
- $M_0 \in \mathbb{N}^n$ é um vetor que contém a marcação inicial para cada local (estado inicial). Neste trabalho, $M(p_n)$ denota o número de tokens do lugar p_n na marcação M .
- $Atts : (Dist, W, G, Policy, Concurrency)^m$ compreende um conjunto de atributos para m transições, onde:
 - $Dist \in \mathbb{N}^m \rightarrow \mathcal{F}$ é uma função de distribuição de probabilidade de disparo possivelmente dependente de marcação. Em uma rede de Petri estocástica temporizada, o tempo deve decorrer entre a ativação e o disparo de uma transição. O tempo real de disparo é uma variável aleatória, para a qual a distribuição é especificada por \mathcal{F} . Nós diferimos entre as transições imediatas ($\mathcal{F} = 0$) e transições temporizadas, para as quais o domínio de \mathcal{F} é $(0, \infty)$.
 - $W \in \mathbb{R}^+$ é a função de peso, que representa um peso de disparo w_t para transições imediatas ou uma taxa λ_t para transições temporizadas. O último é significativo apenas para o caso padrão de transições temporizadas com atrasos de disparo distribuídos exponencialmente. Para transições imediatas, o valor especifica uma probabilidade relativa de disparar a transição quando há várias transições imediatas ativadas na marcação e todas têm a mesma probabilidade. Uma escolha aleatória é então aplicada usando as probabilidades w_t .
 - $G \in \mathbb{N}^n \rightarrow \{\text{true}, \text{false}\}$ é uma função que atribui uma condição de guarda relacionada às marcações de lugar para cada transição. Dependendo da marcação atual, as transições podem não disparar (estão desativadas) quando a função de guarda retorna falso.
 - $Policy \in \{\text{prd}, \text{prs}\}$ é a política de preempção (prd — *repetição preemptiva diferente* significa que quando uma transição preemptiva é habilitada novamente, o tempo de disparo decorrido anteriormente é perdido; prs — *retomada preventiva*, em que o tempo de disparo relacionado a uma transição preemptiva é retomado quando a transição é habilitada novamente).
 - $Concurrency \in \{\text{ss}, \text{is}\}$ é o grau de simultaneidade de transições, onde ss representa a semântica de servidor único e is descreve a semântica do servidor infinito no mesmo sentido que nos modelos de enfileiramento. Transições com política is pode ser entendido como tendo uma transição individual para cada conjunto de tokens de entrada, todos rodando em paralelo.

Em muitas circunstâncias, pode ser adequado representar a marcação inicial como um mapeamento do conjunto de lugares para números naturais ($m_0 : P \rightarrow \mathbb{N}$), onde $m_0(p_i)$ denota a marcação inicial do lugar p_i e $m(p_i)$ denota uma marcação alcançável (estado alcançável) do lugar p_i . Neste trabalho, a notação $\#p_i$ também foi adotado para representar $m(p_i)$. Para obter mais detalhes sobre os conceitos e semântica do SPN, consulte (MARSAN, 1995).

29.2.3.1 Exemplo de modelagem SPN

Um processo simples de carregamento/descarregamento de celular pode ser modelado por meio de SPN. Inicialmente, a bateria do celular está descarregada e leva algum tempo para iniciar o processo de carregamento (duração média $1/\lambda$). Em seguida, o celular é conectado a um carregador e o processo de carregamento é iniciado. A duração do processo de cobrança é considerada em média $1/r$. Após o processo de carregamento, o carregador é removido e o processo de descarga é iniciado (duração média $1/\mu$). Podemos modelar o comportamento do processo de carga/descarga usando a SPN representado na Figura 317. A Figura 318 mostra o gráfico de alcançabilidade correspondente para aquele modelo SPN.

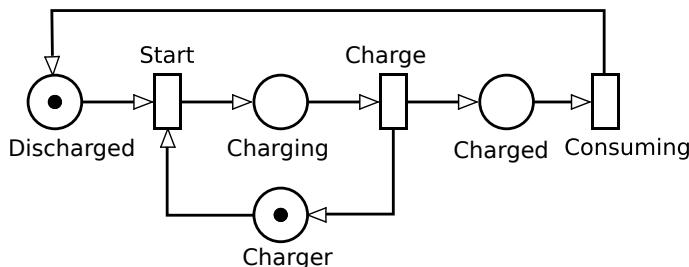


Figura 317 – SPN representando um processo de carregamento/descarregamento de um telefone celular.

- o lugar *Discharged* representa o estado do celular quando ele é descarregado;
- o lugar *Charging* corresponde ao estado do celular quando está em processo de carregamento;
- colocar *Charged* é o estado que modela a situação quando o celular está carregado;
- coloque *Charger* representa o estado do carregador quando não está em uso;
- transição *Start* modela a ação iniciando o processo de carregamento do celular. A taxa dessa transição é λ ;
- transição *Charge* representa o próprio processo de carregamento do celular. A taxa dessa transição é r ;
- transição *Consuming* representa o processo de descarga do celular. A taxa dessa transição é μ .

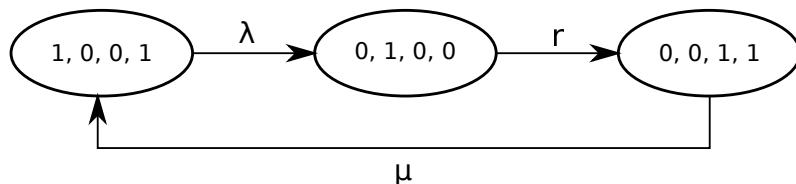


Figura 318 – Gráfico de acessibilidade do modelo SPN da Figura 317.

Nos modelos SPN de avaliação de disponibilidade, existe o conceito de componentes ativos ou inativos. A Figura 319 mostra um pequeno modelo de disponibilidade com dois componentes (A e B). Ambos têm parâmetros de tempos médios para falhas (MTTF) e reparos (MTTR). O componente A, por exemplo, está

ativo quando tem um token no lugar *A_U* (*up*) e inativo quando tem um token no lugar *A_D* (*down*). Neste exemplo, para o componente B estar ativo, A também deverá estar ativo. O arco de inibição garante que, se o componente A mudar o estado *up* para o estado *down*, a transição *T0* será disparada e o componente B também estará no estado *down*.

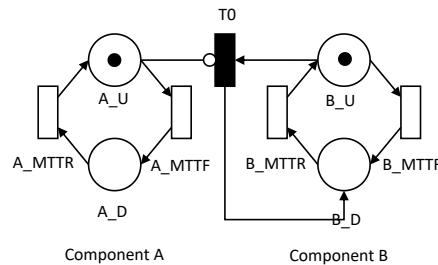


Figura 319 – Exemplo de dependência entre componentes que representam estados de disponibilidade.

29.2.4 Análise de Sensibilidade

A análise de sensibilidade é uma medida do efeito de um determinado dado de entrada sobre os dados de saída, com o objetivo de delinear os elos fracos dos sistemas computacionais, e a partir daí, buscar adotar um conjunto de técnicas que visam melhorar esses sistemas em diferentes cenários (CAMPOLONGO; TARANTOLA; SALTELLI, 1999). De certa forma, a análise de sensibilidade pode trazer a segurança necessária, e encaminhar os resultados dentro da perspectiva pré-estabelecida pelos administradores do sistema.

Existem muitas maneiras de executar a análise de sensibilidade. Vários métodos estão à disposição, como: análise de regressão, análise de perturbação, variação um por um, simulação de Monte Carlo, análise diferencial paramétrica, análise de correlação e diferença percentual. A distinção de qual método utilizar é uma etapa difícil, se fazendo necessário estabelecer quais recursos computacionais são disponíveis, e as características dos problemas abordados (CAMPOLONGO et al., 2004) (PIANOSI et al., 2016).

O método de diferença percentual foi escolhido para realizar a análise de sensibilidade deste trabalho, sendo uma abordagem que não utiliza um domínio contínuo para os valores de entrada de parâmetros. Este processo se baseia no cálculo da diferença percentual, ao variar um parâmetro de entrada de valor mínimo para o seu valor máximo. É necessário a utilização de todo o intervalo de valores possíveis de cada parâmetro para calcular as sensibilidades dos parâmetros (HOFFMAN; GARDNER, 1983).

A Equação 29.1 mostra como o índice de sensibilidade é calculado pelo método de diferença percentual. A expressão $\max \{Y(\theta)\}$ e $\min \{Y(\theta)\}$ são, respectivamente, os valores máximo e mínimo de saída, medidos ao variar o parâmetro θ sobre um intervalo de seus n possíveis valores de interesse. Se $Y(\theta)$ é conhecido por variar monotonicamente, somente os valores extremos de θ (ou seja, θ_1 e θ_n) podem ser usados para calcular $\max \{Y(\theta)\}$, $\min \{Y(\theta)\}$ e como resultado $S_\theta\{Y\}$ (JÚNIOR, 2016).

$$S_\theta\{Y\} = \frac{\max \{Y(\theta)\} - \min \{Y(\theta)\}}{\max \{Y(\theta)\}} \quad (29.1)$$

29.3 Trabalhos Relacionados

Este capítulo apresenta alguns trabalhos relacionados. Três aspectos são analisados: IoT em saúde, utilização de recursos e computação de borda em hospitais inteligentes.

IoT no contexto *healthcare* é um tópico comum no campo de pesquisa atualmente. Alguns artigos consideram **utilização de recursos como parâmetro**. (OUEIDA et al., 2018) propõem um framework de rede de preservação de recursos (RPN) usando rede de Petri. O framework proposto é projetado para modelar recursos não consumíveis e é teoricamente descrito e validado. O RPN se aplica a um cenário da vida real onde os principais indicadores de desempenho, como tempo de permanência do paciente, taxa de utilização de recursos e tempo médio de espera do paciente, são modelados e otimizados. (GRECO; RITROVATO;

XHAFÁ, 2019) propõem uma solução tecnológica e arquitetônica, baseada em tecnologias de *big data Open Source*, para realizar análises em tempo real de fluxos de dados de sensores vestíveis. A arquitetura proposta é composta por quatro camadas distintas: uma camada de detecção, uma camada de pré-processamento (Raspberry Pi), uma camada de processamento de *cluster* (corretor de Kafka e mini-cluster de Flink) e uma camada de persistência (banco de dados Cassandra). A avaliação de desempenho de cada camada foi realizada considerando o uso de CPU e memória para realizar uma tarefa simples de detecção de anomalias usando o conjunto de dados REALDISP. (CHEN et al., 2018) propõem o sistema de saúde inteligente baseado em *Edge Cognitive Computing* (ECC). Este sistema pode monitorar e analisar a saúde física dos usuários por meio da computação cognitiva. Ele também ajusta a alocação de recursos de computação de toda a rede de computação de borda de forma abrangente, de acordo com o grau de risco à saúde de cada usuário. Os experimentos mostram que o sistema de saúde baseado em ECC fornece uma melhor experiência do usuário e otimiza os recursos de computação de maneira razoável, além de melhorar significativamente as taxas de sobrevivência de pacientes em uma emergência repentina.

Os artigos mais próximos a este trabalho abordam o tópico **computação de borda em hospitais inteligentes**. (ZHANG et al., 2018) propõem uma arquitetura para conectar coisas inteligentes em hospitais inteligentes com base em NB-IoT e apresenta a computação de borda para lidar com o requisito de latência no processo de cura. (ZHANG et al., 2018) desenvolveram um sistema de monitoramento de infusão para monitorar a taxa de queda em tempo real e o volume do fármaco restante durante a infusão intravenosa. Além disso, eles discutem os desafios e direções futuras para a construção de um hospital inteligente conectando coisas inteligentes. (RAHMANI et al., 2018) exploram a posição estratégica dos *gateways* na extremidade da rede para oferecer vários serviços de nível superior, como armazenamento local, processamento de dados em tempo real, mineração de dados incorporada, etc. (RAHMANI et al., 2018) em seguida, propõem explorar o conceito de *Fog Computing* em sistemas IoT em Saúde, formando uma camada intermediária geo-distribuída entre nós sensores inteligentes e a nuvem.

Alguns artigos relacionados abordam modelos SPN para avaliar a disponibilidade de sistemas de computação em saúde com suporte para sensores IoT. (PANTELOPOULOS; BOURBAKIS, 2009) apresentam um modelo da Rede de Petri Estocástica de um Sistema de Monitoramento de Saúde Vestível com vários sensores, incluindo uma estrutura de simulação correspondente implementada em Java. (ARAUJO et al., 2014b) propõem um modelo de alto nível que caracteriza o comportamento de um sistema m-Health com o objetivo de identificar a probabilidade de uma mensagem ser entregue no tempo t, enquanto vários componentes envolvidos podem falhar. (TIGRE et al., 2018) e (SANTOS et al., 2018) propõem modelos estocásticos para analisar como as falhas afetam a disponibilidade do sistema eletrônico de saúde.

Diferente de nosso trabalho, os estudos mencionados acima não exploram o desempenho, a disponibilidade e o uso de recursos em conjunto. Alguns estudos também carecem de avaliação de desempenho e análise de disponibilidade no contexto de hospitais inteligentes. A tabela 200 apresenta uma comparação dos trabalhos relacionados.

29.4 Visão Geral da Arquitetura

A Figura 320 apresenta a arquitetura de um hospital inteligente que usa um sistema de monitoramento de saúde baseado em IoT. Sensores conectados ao corpo coletam informações vitais do paciente. Esses dados também podem ser complementados com informações contextuais, como data, hora, local, temperatura, etc. Conhecer o contexto nos permite identificar padrões incomuns e fazer inferências sobre a situação. Outros sensores também podem ser conectados aos sistemas de transmissão de dados para a equipe médica.

Essa arquitetura consiste de um WSN, um *gateway*, um *switch*, um supervisor e dois servidores (local e remoto). A WSN é responsável por coletar sinais biomédicos e contextuais de sensores localizados no corpo dos pacientes e no próprio ambiente. Esses dados podem ser usados para tratamento ou diagnóstico e são transmitidos ao *gateway* por meio de protocolos de comunicação sem fio. O *gateway* deve suportar protocolos de comunicação distintos e atua como um ponto de contato entre os três elementos: o WSN, o supervisor e o *switch*. O *gateway* pode receber dados de diferentes sub-redes e deve realizar a conversão do protocolo de comunicação e fornecer outros serviços de nível superior, como agregação de dados, filtragem

Tabela 200 – Comparaçao de trabalhos relacionados.

Trabalho Relacionado	Métrica	Análise de Sensibilidade Um por Um	Modelo	Dependência de Componente
(OUEIDA et al., 2018)	Desempenho e Recursos	Não	Redes de Petri	Não
(CHEN et al., 2018)	Recursos	Não	Não Aplicado	Não
(ZHANG et al., 2018)	Recursos	Não	Block Diagram	Não
(RAHMANI et al., 2018)	Desempenho e Recursos	Não	Não Aplicado	Não
(GRECO; RITROVATO; XHAFA, 2019)	Desempenho	Não	Modelo em Camadas	Não
(ARAUJO et al., 2014b)	Desempenho e Disponibilidade	Não	Redes de Petri e RBD	Não
(TIGRE et al., 2018)	Desempenho	Sim	Redes de Petri	Sim
(SANTOS et al., 2018)	Desempenho e Disponibilidade	Sim	Redes de Petri e RBD	Sim
Este Trabalho	Desempenho, Disponibilidade e Recursos	Sim	Redes de Petri	Sim

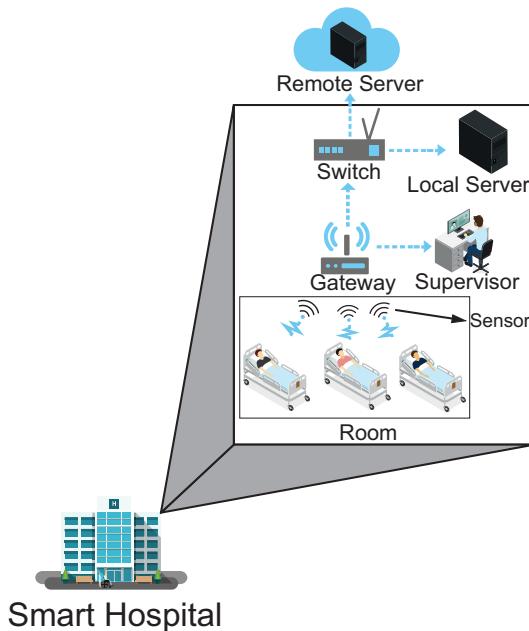


Figura 320 – Visão geral da arquitetura avaliada.

e assim por diante.

O supervisor pode ser um *smartphone* ou *tablet* por meio do qual o médico ou enfermeiro monitora o estado dos pacientes. O *switch* é responsável por criar rotas para o servidor local e servidor remoto. O servidor local armazena os dados para pesquisas posteriores. O servidor remoto garante a escalabilidade desses dados, proporcionando mais segurança e redundância.

29.5 Avaliação de Desempenho de um Hospital Inteligente

Esta capítulo apresenta o modelo SPN para um hospital inteligente, bem como métricas de desempenho e análises numéricas, para exemplificar a aplicabilidade da proposta.

29.5.1 Modelo SPN para Arquitetura de um Hospital Inteligente

A Figura 321 apresenta o modelo SPN para a arquitetura proposta com os seguintes componentes: (i) *Admission* que trata da chegada dos dados; (ii) *Gateway* que encaminha os dados para o servidor supervisor e para o switch; (iii) *Supervisors* que recebem os dados para realizar o monitoramento do paciente. (iv) *Switch* gerencia a rota para o servidor local e remoto. (v) *Local Server* armazena dados do paciente para análise futura e para evitar perda de dados. (vi) *Remote Server* processa e armazena os dados na nuvem.

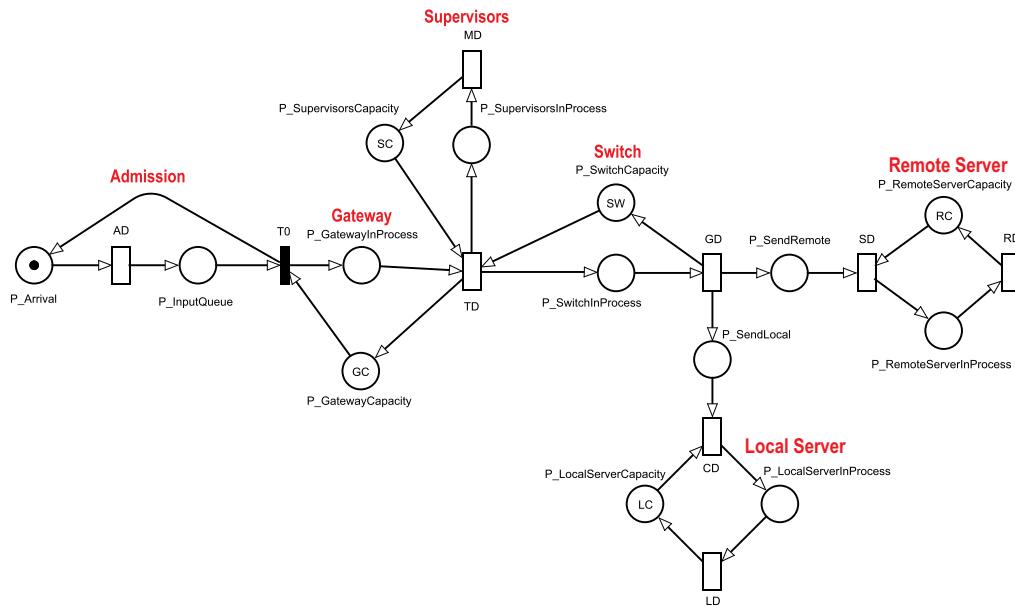


Figura 321 – Modelo SPN para uma arquitetura hospitalar inteligente.

Os componentes são representados por gráficos como lugares (círculos), transições temporizadas (barras vazias) e marcações de lugares (círculos preenchidos). A Tabela 201 descreve todos os elementos do modelo. As transições temporizadas são parametrizadas com distribuições de probabilidade. O administrador do sistema deve definir essas distribuições de acordo com a literatura ou realizar medições e caracterizações do sistema.

Dada a visão geral do modelo, agora descrevemos o fluxo de tokens entre os componentes do modelo. A sub-rede de Admissão consiste em dois locais $P_{Arrival}$ e $P_{InputQueue}$, que representam o atraso entre a entrada e aceitação destes dados na fila. Os tokens gerados em $P_{Arrival}$ representam qualquer tipo de solicitação que envolva entrada de dados a serem processados ou armazenados.

A transição T_0 representa o recebimento da solicitação, note que é uma transição imediata, não tendo nenhum atraso associado. T_0 dispara assim que um token existe em $P_{InputQueue}$ e há pelo menos um token em $P_{GatewayCapacity}$. A capacidade do gateway pode ser interpretada como canais de distribuição disponíveis, dada pela marcação GC .

Quando T_0 dispara, a sub-rede do gateway é alcançada. Um token é obtido de $P_{InputQueue}$ e $P_{GatewayCapacity}$. Um token é retornado para $P_{Arrival}$, permitindo um novo gatilho. Um token é então adicionado ao local $P_{GatewayInProcess}$. O número de tokens em $P_{GatewayInProcess}$ representa a fila de solicitações no gateway. O enfileiramento ocorre quando não há capacidade disponível para atender à solicitação recém-chegada. Se houver capacidade disponível no supervisor e switch (locais $P_{SupervisorsCapacity}$ e $P_{SwitchCapacity}$), a transição TD é disparada para ambos simultaneamente. TD representa o atraso para o gateway enviar uma solicitação o supervisor e para um switch.

Se houver recurso suficiente no supervisor, $P_{SupervisorsInProcess}$ conterá o número de solicitações na fila para processamento. O tempo que as solicitações permanecem em processamento em um nó depende da transição MD . Essas transições têm a semântica *infinite server*, portanto, cada solicitação é processada de forma independente. É importante notar que o tempo de processamento depende significativamente da

Tabela 201 – Descrição de cada elemento do modelo, incluindo lugares, transições e marcas.

Tipo	Elemento	Descrição	Semântica do Servidor
Lugares	P_Arrival	Gera novas requisições	-
	P_InputQueue	Aguarde a disponibilidade da fila	-
	P_GatewayInProcess	Requisições na fila do gateway	-
	P_GatewayCapacity	Capacidade do gateway	-
	P_SupervisorsInProcess	Requisições na fila do supervisor	-
	P_SupervisorsCapacity	Capacidade do servidor supervisor	-
	P_SwitchInProcess	Requisições na fila do switch	-
	P_SwitchCapacity	Capacidade do switch	-
	P_LocalServerInProcess	Requisições na fila do servidor local	-
	P_LocalServerCapacity	Capacidade do servidor local	-
Transições temporizadas	P_RemoteServerInProcess	Requisições na fila do servidor remoto	-
	P_RemoteServerCapacity	Capacidade do servidor remoto	-
	AD	Tempo entre chegadas	Single Server
	TD	Tempo para o gateway enviar solicitações ao supervisor e switch	Infinite Server
	MD	Tempo de processamento da solicitação no servidor supervisor	Infinite Server
	GD	Tempo de envio para o servidor local e para o servidor remoto	Infinite Server
	CD	Atraso para que as requisições cheguem ao servidor local	Infinite Server
	LD	Tempo de processamento da solicitação no servidor local	Infinite Server
	SD	Atraso para que as requisições cheguem ao servidor remoto	Infinite Server
	RD	Tempo de processamento da solicitação no servidor remoto	Infinite Server
Marcações de lugar	GC	Capacidade máxima da fila de entrada do gateway	-
	SC	Capacidade máxima do servidor supervisor	-
	SW	Capacidade máxima da fila do switch	-
	LC	Capacidade máxima da fila do servidor local	-
	RC	Capacidade do servidor remoto	-

capacidade computacional dos nós do servidor supervisor. Assim, a transição *MD* deve ser configurada com tempo de processamento para um único trabalho em um tipo de recurso específico. Depois que o *GD* for acionado, a solicitação segue em ambos os lados, para o servidor local e o servidor remoto simultaneamente. A seguir, é necessário aguardar a hora de chegada dos dados no servidor local, e esta hora é representada pela transição *CD*. Depois que a solicitação chega, ela segue para o processamento.

O tempo que as solicitações permanecem em processamento no servidor local depende da transição *LD*. *LD* representa o tempo de serviço do servidor local. Tal transição possui a semântica *infinite server*, portanto cada requisição é processada independentemente se houver recurso disponível em *P_LocalServerCapacity*. Novamente, é essencial observar que o tempo de processamento depende significativamente da capacidade computacional do nó (VM ou contêiner, por exemplo) e do banco de dados usado para armazená-lo. Assim, a transição *LD* deve ser configurada com tempo de processamento para um único trabalho em um tipo de recurso específico.

Se não houver recurso no servidor remoto (*P_RemoteServerCapacity*), haverá enfileiramento no local *P_SendRemote*. Nesse caso, há um tempo associado ao envio de uma solicitação para o servidor remoto (transição *SD*) e tempo de serviço no servidor remoto (transição *RD*). Essas transições também utilizam a semântica de *infinite server*, portanto, cada solicitação é processada de forma independente.

O tempo entre as chegadas é atribuído à transição *AD*. Consideramos que os tempos entre os disparos são distribuídos exponencialmente; esta suposição pode ser modificada alterando esta distribuição. A

transição AD leva em consideração apenas o tempo que as requisições entram no sistema, ou seja, as perdas na rede não são levadas em consideração.

O modelo proposto permite avaliar diversos cenários, visto que o avaliador pode configurar até 13 parâmetros (8 transições e 5 marcações), conforme Tabela 201. Qualquer alteração em um desses parâmetros pode impactar significativamente no tempo médio de resposta do sistema e, consequentemente, no custo da infraestrutura. A variação dos cenários considerando um número significativo de fatores torna este modelo a principal contribuição deste trabalho.

29.5.2 Métricas de Desempenho

Esta seção define métricas para avaliar a arquitetura de um hospital inteligente com base no modelo proposto. Neste trabalho calculamos quatro métricas: tempo médio de resposta (MRT), probabilidade de descarte ($Discard_Probability$), número de solicitações descartadas em relação ao tempo ($Discard_Number$) e probabilidade de utilização de recurso ($Utilization<ServerName>$).

O MRT pode ser obtido na Lei de Little (LITTLE; GRAVES, 2008) que relaciona o número médio de solicitações em andamento em um sistema ($RequestsInProcess$) e a taxa de chegada de novas solicitações ($ArrivalRate$). A taxa de chegada é o inverso do atraso de chegada: $ArrivalRate = \frac{1}{AD}$. A Lei de Little exige um sistema estável, o que significa que a taxa de chegada é inferior ao tempo de serviço. O MRT é obtido pela Equação 29.2.

$$MRT = RequestsInProcess \times AD \quad (29.2)$$

Portanto, é possível também calcular o MRT através da equação:

$$MRT = \frac{RequestsInProcess}{ArrivalRate} \quad (29.3)$$

A Equação 29.4 define $RequestsInProcess$. O número de solicitações em andamento no sistema é calculado somando o número de tokens em cada um dos locais que representam uma solicitação em andamento. Na Equação 29.4, $Esp(Place)$ representa o valor estatístico esperado dos tokens em um “Place” (local), onde $Esp(Place) = (\sum_{i=1}^n P(m(Place) = i) \times i)$. Em outras palavras, $Esp(Place)$ indica quantos tokens ocupam aquele Place.

$$\begin{aligned} RequestsInProcess = & Esp(P.GatewayInProcess) + \\ & Esp(P.SupervisorsInProcess) + Esp(P.SwitchInProcess) + \\ & Esp(P.LocalServerInProcess) + Esp(P.RemoteServerInProcess) \end{aligned} \quad (29.4)$$

A equação 29.5 define a probabilidade de descarte ($Discard_Probability$). Deve haver um token na fila de entrada ($P_ArrivalQueue$) e não haver mais recursos disponíveis no gateway para calcular o descarte. $P(Place = n)$ calcula a probabilidade de n nós no “Place”.

$$Discard_Probability = (P((P_InputQueue \Rightarrow 1) \wedge (P.GatewayCapacity = 0))) \times 100 \quad (29.5)$$

A equação 29.6 define o número de requisições descartadas em um determinado período de tempo T . Para obter o número de descarte no período T , basta multiplicar a probabilidade de descarte por $ArrivalRate$ e o tempo T .

$$Discard_Number = Discard_Probability \times ArrivalRate \times T \quad (29.6)$$

Por fim, calculamos também a probabilidade de utilização de recursos, onde a equação é dada pelo número de tokens no local correspondente ao momento de execução dividido pela capacidade total desses recursos. Fazemos isso para cada componente de recurso do modelo. Note que a capacidade em questão é dada pela marcação do local correspondente a esse recurso.

A equação de utilização do gateway é dada por:

$$UtilizationGateway = \frac{Esp(P.GatewayInProcess)}{GC} \times 100 \quad (29.7)$$

A equação para a utilização do supervisor é dada por:

$$\text{UtilizationSupervisors} = \frac{\text{Esp}(P_SupervisorsInProcess)}{SC} \times 100 \quad (29.8)$$

A equação para a utilização do *switch* é dada por:

$$\text{UtilizationSwitch} = \frac{\text{Esp}(P_SwitchInProcess)}{SW} \times 100 \quad (29.9)$$

A equação para a utilização do servidor local é dada por:

$$\text{UtilizationLocalServer} = \frac{\text{Esp}(P_LocalServerInProcess)}{LC} \times 100 \quad (29.10)$$

A equação para utilização de servidor remoto é dada por:

$$\text{UtilizationRemoteServer} = \frac{\text{Esp}(P_RemoteServerInProcess)}{RC} \times 100 \quad (29.11)$$

29.5.3 Análise Numérica

Esta seção apresenta oito análises numéricas do modelo. A tabela 202 exibe os valores atribuídos às transições temporizadas e aos lugares do modelo SPN. Novamente, variamos o valor da transição correspondente ao atraso de chegada (AD) de 1,0 ms a 10,0 ms com incrementos de 0,5 ms. Os outros parâmetros permaneceram fixos. Todos os resultados da análise com o modelo são apresentados nos gráficos das Figuras 322(a), 322(b), 322(c), 322(d) e 322(e).

Tabela 202 – Valores atribuídos para transições e marcações do modelo SPN.

Tipo	Elemento	Valor
Transições Temporizadas	AD	[1,0-10,0] (Incremento de 0,5(ms))
	TD	3,5(ms)
	MD	20,3(ms)
	GD	2,5(ms)
	CD	1,0(ms)
	LD	20,3(ms)
	SD	4,0(ms)
	RD	20,3(ms)
Lugares	GC	8
	SC	8
	SW	8
	LC	16
	RC	16

29.5.3.1 Tempo Médio de Resposta (MRT)

A Figura 322(a) exibe os resultados do MRT. O MRT aumenta até AD = 3,0ms, com MRT = 140ms; depois disso, diminui. Há um ligeiro aumento em AD = 4,0 ms, e então começa a decair de forma constante. A Figura 322(d) mostra a utilização do *gateway*, onde podem ser notados requisições enfileiradas com 85% de utilização no AD = 3,0ms.

29.5.3.2 Probabilidade de Descarte

A Figura 322(b) apresenta a probabilidade de descarte de solicitações. No ponto mais crítico (1,0 ms), a probabilidade é de cerca de 75%. É possível notar que essa probabilidade diminui à medida que a DA aumenta (entre 1,0ms e 5,5ms) até estagnar a zero quando a DA atinge 6,0ms. Considerando o joelho (AD = 3,0ms), a probabilidade de descartar solicitações é inferior a 40%.

29.5.3.3 Número de Descartes

A Figura 322(c) exibe o número de descartes em um determinado tempo T. O período T considerado neste texto foi de 10,0ms. À medida que AD aumenta, o número de dados descartados diminui. A partir de 6,5ms, é possível observar que o número de descartes chega a zero. No ponto mais crítico (AD = 1,0), o número de

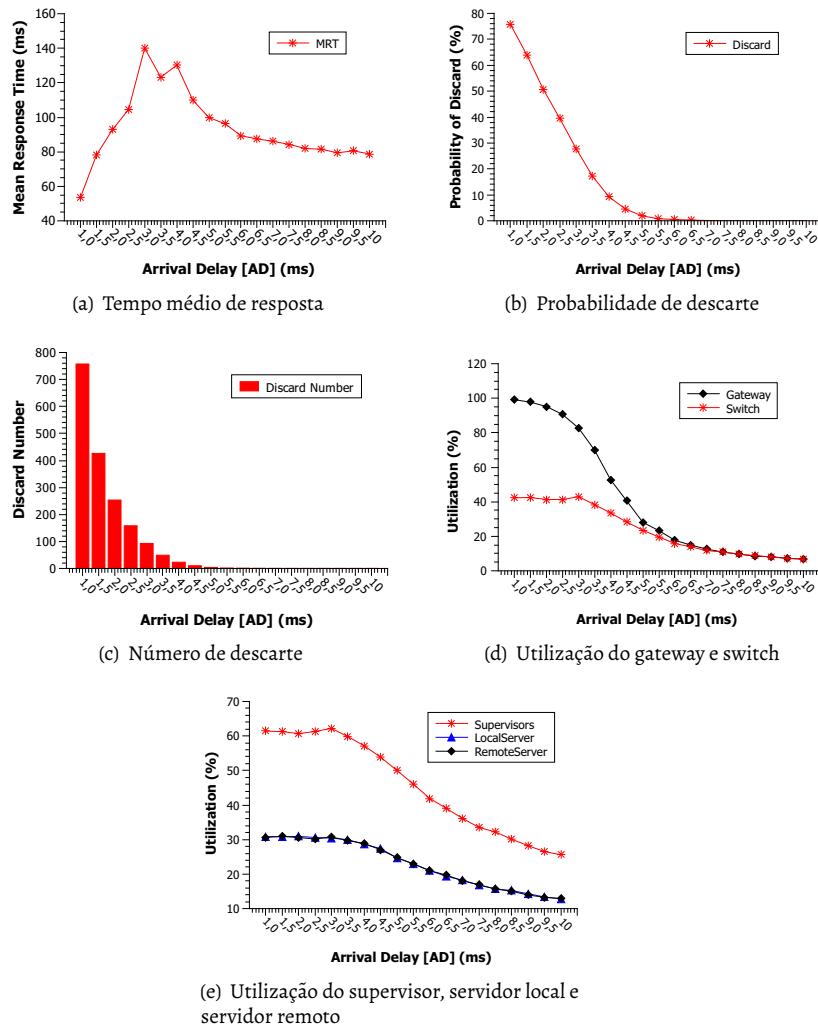


Figura 322 – Resultados da avaliação de desempenho do modelo

descartes é de aproximadamente 760 requisições. Para tal cenário, não haverá perdas se a DA ultrapassar 6,0ms. No joelho MRT ($AD = 3,0\text{ms}$), temos aproximadamente 100 solicitações perdidas.

29.5.3.4 Utilização

A Figura 322(d) mostra o nível de utilização do *gateway* e do *switch*. Para $AD = 1,0\text{ ms}$, a utilização do *gateway* atinge 100%. Com valores de AD acima de $1,5\text{ms}$, o valor cai continuamente, atingindo valores menores que 10% quando o AD atinge aproximadamente $10,0\text{ms}$. Conforme explicado anteriormente, o *switch* tem uma porcentagem menor de utilização, uma vez que o gargalo do *gateway* não permite que as solicitações cheguem na mesma frequência no *switch*. No intervalo de tempo de $7,5\text{ms}$ a $10,0\text{ms}$, o *gateway* e o *switch* possuem o mesmo nível de utilização.

A Figura 322(e) mostra o nível de utilização do supervisor, do servidor local e do servidor remoto. O supervisor com $AD = 1,0\text{ms}$ tem aproveitamento de aproximadamente 62%, com algumas variações até sua redução após o AD atingir $3,5\text{ms}$. Para $AD = 10,0\text{ms}$, sua utilização atinge aproximadamente 36%. Tanto o servidor local quanto o supervisor têm comportamentos semelhantes. Essas máquinas partem da utilização de 31% e, para $AD = 10,0\text{ms}$, sua utilização chega a aproximadamente 3%. A capacidade do servidor local (16 núcleos) e do servidor remoto (16 núcleos) é maior do que a capacidade do supervisor (8 núcleos).

29.5.4 Validação do modelo

Esta seção apresenta a validação do modelo SPN proposto. O objetivo desta validação foi comparar o MRT calculado pelo modelo e o MRT coletado por experimentos em cenários reais. A Figura 323 apresenta um esboço do experimento prático realizado. Assim como na arquitetura proposta neste trabalho, na validação simulamos a existência de vários componentes (gerador de requisições, *gateway*, *switch*, supervisor, servidor local e servidor remoto) que se comunicam. Cada requisição é processada por todos os componentes, portanto, se chegarem 5 requisições, as mesmas serão transmitidas para todos os componentes. Para realizar a validação, foi desenvolvido um sistema sintético que simula o envio de requisições obedecendo uma distribuição exponencial.

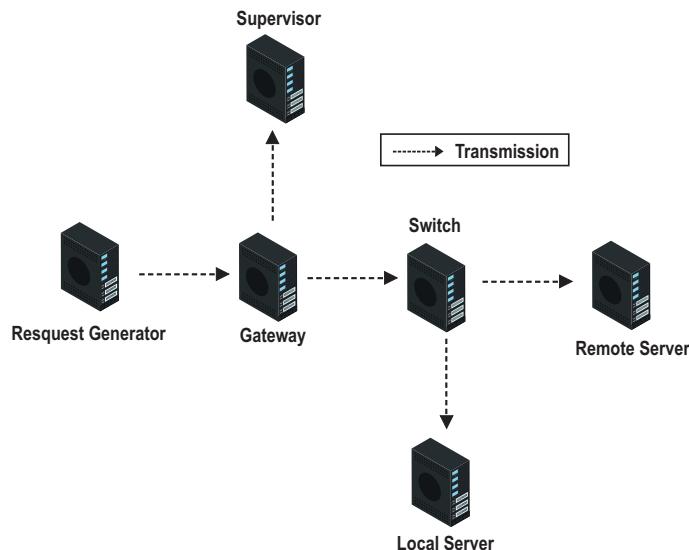


Figura 323 – Esquema de experimento prático para validação do modelo.

A Tabela 203 apresenta as configurações dos computadores utilizados para realizar a validação juntamente com os seus respectivos perfis e funções. Os computadores podem ser divididos em categorias sendo elas: os que são responsáveis por encaminhar dados (*gateway* e *switch*); os que são responsáveis por processar dados (supervisor) e os que são responsáveis por salvar os dados (servidor local e servidor remoto).

Tabela 203 – Computadores utilizados no experimento

Configuração do computador	Perfil	Função
Intel Core i5 2.5Ghz 12GB	Gerador de requisições	Gera solicitações para serem processadas nos servidores
Intel Core i3 3.7Ghz 8GB	Gateway	Recebe, agrupa e transmite os dados
Intel Core i3 3.7Ghz 4GB	Supervisor	Recebe e processa as requisições
Intel Core i3 3.7Ghz 4GB	Switch	Recebe e encaminha os dados
Intel Core i5 3.4Ghz 8GB	Servidor Local	Recebe e salva os dados localmente
Intel Core i5 3.4Ghz 8GB	Servidor Remoto	Recebe e salva os dados remotamente

O sistema sintético é um algoritmo de classificação de API de coleções Java simples (`java.util`) que recebe e processa um vetor de números. Porém, o sistema sintético é capaz de realizar qualquer serviço. No início da validação, é necessário um experimento inicial para coletar os tempos médios de cada etapa para alimentar as transições do modelo. O sistema sintético gera *logs* durante o processo.

A Figura 324 apresenta um diagrama de sequência do experimento, com as etapas de envio e processamento dos dados. Cada solicitação utilizada no experimento consiste em um arquivo contendo uma matriz de tamanho 4x4 preenchida com valores numéricos aleatórios. O gerador de solicitação simula x número de solicitações que são enviadas ao *gateway* dentro de um determinado intervalo (AD). O *gateway* recebe as solicitações e as encaminha ao *switch* e ao *supervisor*. No *supervisor*, essa matriz é processada, mostrando

o resultado ao usuário. O switch encaminha a matriz para servidores locais e remotos. A matriz é salva em um banco de dados nos servidores local e remoto.

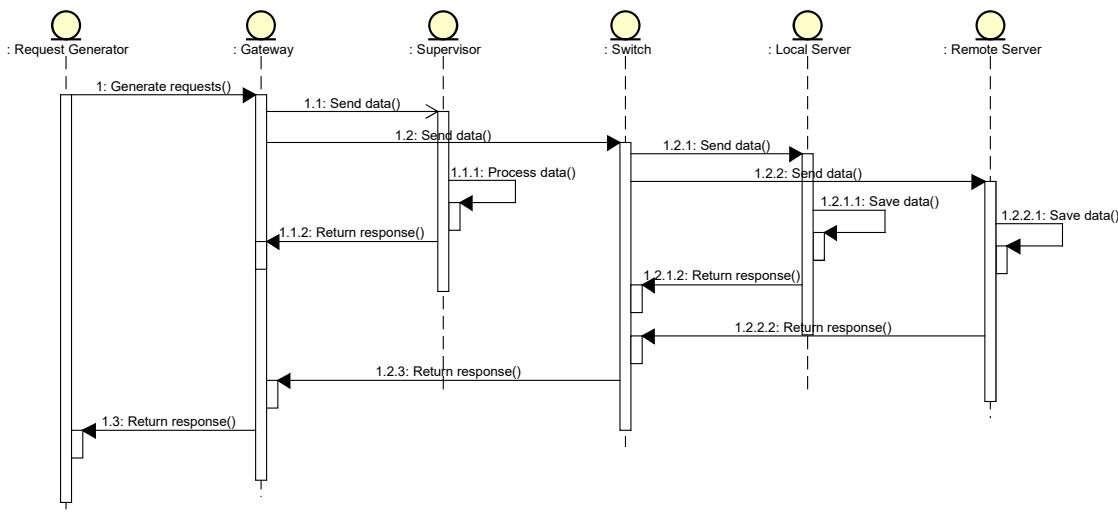


Figura 324 – Diagrama de sequência do experimento prático.

O *switch* espera por confirmações do servidor local e do servidor remoto. Ao chegarem, o *switch* envia uma confirmação para o *gateway*, que então envia outra confirmação para o gerador da solicitação, gerando uma espécie de verificação. A verificação garante que os dados foram transmitidos a todos os componentes e nenhum problema de processamento foi encontrado.

A partir dos experimentos iniciais mencionados acima, foram obtidos os tempos de transição para alimentar o modelo. Os parâmetros usados no modelo que espelham o experimento real foram relacionados às transições e ao número de tokens nos lugares. Para as transições, os seguintes valores foram configurados: TD = 4,5ms, MD = 4,5ms, GD = 2,0ms, CD = 2,5ms, SD = 80,0ms, LD = 42ms, RD = 42ms. Para a capacidade dos componentes, os seguintes valores foram definidos: P_GatewayCapacity = 8, P_SupervisorsCapacity = 8, P_SwitchCapacity = 8, P_LocalServerCapacity = 16, P_RemoteServerCapacity = 16. Essa configuração representa que cada componente tem apenas quatro threads para lidar com a transmissão de dados e quatro threads para lidar com o processamento de solicitações.

29.5.4.1 Resultados da Validação

A Tabela 204 apresenta os MRTs e ADs utilizados no experimento, onde os MRTs estão relacionados a cada solicitação liberada. Para cada AD, 30 solicitações foram acionadas. O modelo também foi configurado para acionar apenas 30 solicitações. Diferentes quantidades de solicitações afetariam o MRT devido a longas filas.

A Figura 325 apresenta os resultados comparando os MRTs do experimento real com os do modelo. Para cada AD, os valores MRT do experimento e modelo foram muito próximos.

Após verificar que as amostras apresentavam distribuição normal, optou-se por aplicar o teste T de uma amostra às três comparações. Adotamos o teste T de uma amostra para comparar o MRT gerado pelo modelo com o MRT obtido no experimento. Para verificar se a hipótese nula de que as médias são iguais, foi observado o valor p. Considerando um intervalo de confiança de 95 %, os valores de p foram: AD = 50ms, p = 0,135; AD = 60 ms, p = 0,677; AD = 70ms, p = 0,683; AD = 80 ms, p = 0,186; AD = 90ms, p = 0,301; AD = 100ms, p = 0,066. Em todos os casos, o valor de p é maior que 0,05, indicando que os resultados gerados pelo modelo são estatisticamente equivalentes ao experimento.

Tabela 204 – Tempos obtidos no experimento prático.

Requisição	MRT (ms)											
	AD 100ms	=	AD 90ms	=	AD 80ms	=	AD 70ms	=	AD 60ms	=	AD 50ms	=
1	345		228		234		225		217		249	
2	318		235		117		199		200		255	
3	259		230		180		237		130		270	
4	226		155		164		281		170		365	
5	180		118		145		164		136		418	
6	226		129		226		228		156		498	
7	196		134		291		213		111		569	
8	117		169		276		249		138		640	
9	105		233		292		163		174		663	
10	162		139		312		203		224		696	
11	112		146		335		258		192		732	
12	112		124		359		156		203		779	
13	143		74		250		218		252		808	
14	119		85		287		279		250		839	
15	106		142		389		379		305		864	
16	140		201		170		395		359		882	
17	109		202		103		350		409		860	
18	109		200		160		416		498		895	
19	185		125		157		311		491		848	
20	177		118		84		341		521		713	
21	110		159		108		353		475		769	
22	131		210		103		398		474		795	
23	156		324		154		470		528		722	
24	209		253		225		134		603		726	
25	145		301		235		224		648		805	
26	198		334		277		280		677		756	
27	254		367		208		370		758		713	
28	175		387		205		402		745		739	
29	216		253		275		399		788		757	
30	215		160		336		466		812		826	

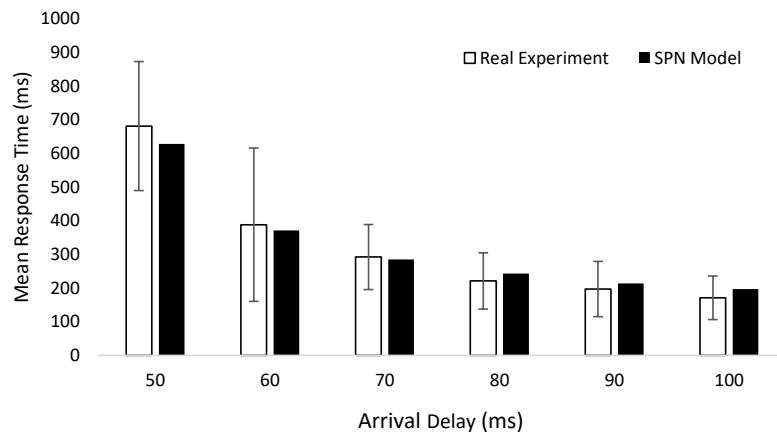


Figura 325 – Comparaçāo entre modelo e experimento.

29.6 Avaliando a Disponibilidade de um Hospital Inteligente

Este capítulo apresenta um modelo SPN para avaliar a disponibilidade de um sistema hospitalar inteligente, considerando a arquitetura apresentada anteriormente. No entanto, este capítulo detalha a arquitetura um pouco mais em termos de camadas do servidor.

29.6.1 Arquitetura em Camadas

A Figura 326 apresenta a arquitetura em camadas dos servidores (remotos e locais), incluindo (*hardware, operating system, hypervisor, virtual machine, data process e storage*). Devemos destacar que os componentes de software *data process* e *storage* são responsáveis pelo processamento e armazenamento dos dados do sensor, respectivamente. As camadas superiores só estarão disponíveis se as camadas inferiores estiverem funcionando.

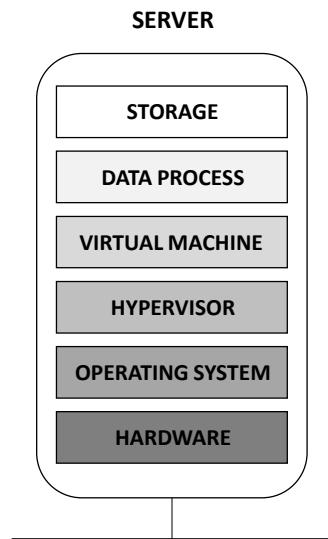


Figura 326 – Arquitetura dos servidores baseada em camadas.

29.6.2 Modelo SPN

A Figura 327 apresenta um segundo modelo SPN para a arquitetura de um hospital inteligente com foco na disponibilidade. O modelo SPN é composto por *switch* (SW), *gateway* (GA), *sensors* (SE), *supervisor* (SU), *local server* (LS) e *remote server* (RS). O *local server* e o *remote server* possuem vários componentes interconectados para o sistema funcionar. O *hardware* (HW), *operating system* (OS), *hypervisor* (HV), *virtual machine* (VM), *data process* (DP) e *storage* (S), fazem parte da composição de LS e RS. Cada componente tem um MTTF (tempo médio entre falhas) e um MTTR (tempo médio para reparo). Transições imediatas são usadas para conectar os componentes que compõem o *local server* e *remote server*. Essas transições garantem que, se um componente falhar, os componentes dependentes também falharão.

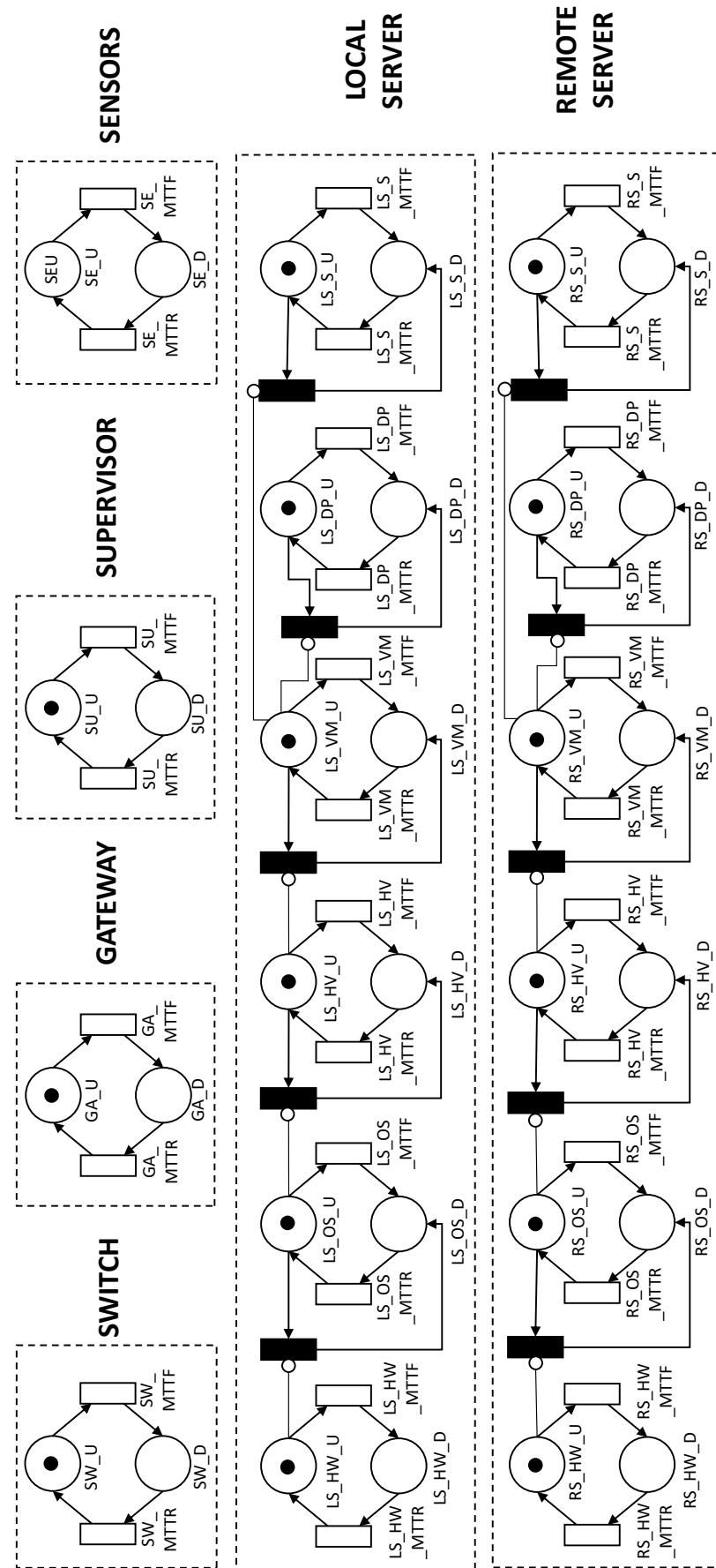


Figura 327 – Modelo SPN de disponibilidade de um hospital inteligente.

O número de sensores ativos é dado pelo número de tokens em SE_U (*sensors up*). O número de tokens no lugar SE_D (*sensors down*) representa o número de sensores inativos. A mudança entre o estado ativo/inativo para cada sensor é acionada pelas transições: SE_MTTF (MTTF dos *sensors*) e SE_MTTR (MTTR dos *sensors*). Consideramos que o *gateway* está funcionando quando possui um token no local GA_U (*gateway up*) e não está funcionando quando possui um token no local GA_D (*gateway textitdown*). GA_MTTF (MTTF do *gateway*) e GA_MTTR (MTTR do *gateway*) alteram os estados do *gateway*. Semelhante ao *gateway*, consideramos que *switch* está funcionando quando tem um token no local SW_U (*switch textitup*); e não está funcionando quando possui token em SW_D (*switch textitdown*). A mudança ativo/inativo do *switch* é acionada pelas transições: transições SW_MTTF (MTTF do *switch*) e SW_MTTR (MTTR do *switch*). O *supervisor* está funcionando quando tem um token no lugar SU_U (*supervisor up*) e não está funcionando com um token no SU_D (*supervisor down*). A mudança ativo/inativo do *supervisor* é acionada pelas transições: SU_MTTF (MTTF do *supervisor*) e SU_MTTR (MTTR do *supervisor*).

O *local server* estará operacional quando tiver tokens nos locais LS_DP_U (*data process up*) e LS_S_U (*storage up*). Consideramos que o *local server* não está funcionando quando tem um token em um dos seguintes locais: LS_HW_D (*hardware down*) ou LS_OS_D (*operating system down*) ou LS_HV_D (*hypervisor down*) ou LS_VM_D (*virtual machine down*) ou LS_DP_D (*data process down*) ou LS_S_D (*storage down*). Alterações ativas/inativas são acionadas por: LS_HW_MTTF, LS_OS_MTTF, LS_HV_MTTF, LS_VM_MTTF, LS_DP_MTTF e LS_S_MTTF — para tempo médio entre falhas — e LS_HW_MTTR, LS_OS_MTTR, LS_HV_MTTR, LS_VM_MTTR, LS_DP_MTTR e LS_S_MTTR — para tempo médio para reparo.

O *remote server* estará funcionando quando tiver tokens nos locais: RS_DP_U (*data process up*) and RS_S_U (*storage up*). Consideramos que o *remote server* não está funcionando quando possui um token em um dos seguintes componentes: RS_HW_D (*hardware down*) ou RS_OS_D (*operating system down*) ou RS_HV_D (*hypervisor down*) ou RS_VM_D (*virtual machine down*) ou RS_DP_D (*data process down*) ou RS_S_D (*storage down*). As alterações ativas/inativas são acionadas por: RS_HW_MTTF, RS_OS_MTTF, RS_HV_MTTF, RS_VM_MTTF, RS_DP_MTTF e RS_S_MTTF — para tempo médio entre falhas — e RS_HW_MTTR, RS_OS_MTTR, RS_HV_MTTR, RS_VM_MTTR, RS_DP_MTTR e RS_S_MTTR — para tempo médio para reparo.

A Tabela 205 apresenta as condições de guarda para as principais transições. As condições de proteção garantem que as transições sejam acionadas apenas por uma condição específica. Em nosso modelo, as condições são adotadas para permitir a dependência dos componentes. Por exemplo, o sistema operacional pode ser reparado apenas se o hardware estiver ativo.

Tabela 205 – Condições de guarda que restringem o disparo de transições.

Servidor	Transição	Condição
Local	LS_OS_MTTR	P{#LS_HW_U>0}
	LS_HV_MTTR	P{#LS_OS_U>0}
	LS_VM_MTTR	P{#LS_HV_U>0}
	LS_DP_MTTR	P{#LS_VM_U>0}
	LS_S_MTTR	P{#LS_VM_U>0}
Remote	RS_OS_MTTR	P{#RS_HW_U>0}
	RS_HV_MTTR	P{#RS_OS_U>0}
	RS_VM_MTTR	P{#RS_HV_U>0}
	RS_DP_MTTR	P{#RS_VM_U>0}
	RS_S_MTTR	P{#RS_VM_U>0}

29.6.3 Estudo de Caso

Esta seção apresenta um estudo de caso baseado no modelo proposto, incluindo uma análise de disponibilidade e uma análise de sensibilidade.

29.6.3.1 Definindo Cenários e Métricas

A Figura 328 apresenta os servidores em três cenários propostos (A, B, C). O cenário A descreve a métrica de disponibilidade de todo o sistema com um servidor local e remoto no modo ativo. O cenário B descreve a métrica de disponibilidade do sistema apenas com o servidor local em execução. O cenário C representa a métrica de disponibilidade do sistema apenas com o servidor remoto em execução.

	Local	Remote	Local	Remote
Up				
Down				
	Scenario A	Scenario B	Scenario C	

Figura 328 – Estados do servidor em cada cenário.

Duas métricas foram adotadas para esta avaliação: disponibilidade e tempo de inatividade por ano. A equação de disponibilidade representa a soma das probabilidades do número de tokens em cada estado. Assim, P representa probabilidade e # representa o número de tokens em um determinado local. A Tabela 206 apresenta as métricas usadas para obter a disponibilidade do sistema. O tempo de inatividade (D) pode ser obtido por $D = (1 - A) \times 8760$, onde A representa a disponibilidade do sistema e 8760 o número de horas no ano.

Tabela 206 – Métricas utilizadas.

Cenário	Métrica da Disponibilidade
A	$A = P\{((#RD_HW_U>0)\text{AND}(\#RD_OS_U>0)\text{AND}(\#RD_HV_U>0))\text{AND}((#RD_VM_U>0)\text{AND}(\#RD_DP_U>0)\text{AND}(\#RD_S_U>0))\text{OR}\\((#LD_HW_U>0)\text{AND}(\#LD_OS_U>0)\text{AND}(\#LD_HV_U>0))\text{AND}((#LD_VM_U>0)\text{AND}(\#LD_DP_U>0)\text{AND}(\#LD_S_U>0))\text{AND}\\((#GA_U>0)\text{AND}(\#SW_U>0)\text{AND}(SE_U=SEU)\text{AND}(\#SUP_U>0)\}$
B	$A = P\{((#LD_HW_U>0)\text{AND}(\#LD_OS_U>0)\text{AND}(\#LD_HV_U>0))\text{AND}((#LD_VM_U>0)\text{AND}(\#LD_DP_U>0)\text{AND}(\#LD_S_U>0))\text{AND}\\((#GA_U>0)\text{AND}(\#SW_U>0)\text{AND}(SE_U=SEU)\text{AND}(\#SUP_U>0)\}$
C	$A = P\{((#RD_HW_U>0)\text{AND}(\#RD_OS_U>0)\text{AND}(\#RD_HV_U>0))\text{AND}((#RD_VM_U>0)\text{AND}(\#RD_DP_U>0)\text{AND}(\#RD_S_U>0))\text{AND}\\((#GA_U>0)\text{AND}(\#SW_U>0)\text{AND}(SE_U=SEU)\text{AND}(\#SUP_U>0)\}$

29.6.3.2 Resultados

Alguns parâmetros de entrada são necessários para avaliar o modelo proposto. Os tempos de falha e reparo de componentes foram extraídos de (ARAUJO et al., 2019) e (LISBOA et al., 2018). Os parâmetros do *switch* e do *gateway* foram extraídos do site da Cisco. O *gateway* adotado foi o roteador Cisco 4461¹ e o *switch* foi o *switch* Cisco Catalyst 1000 Series para o modelo de 24 portas². A Tabela 207 apresenta todos os valores de MTTF e MTTR adotados.

A Figura 329 apresenta os resultados de disponibilidade e tempo de inatividade para um hospital inteligente considerando os três cenários mencionados acima. Os resultados foram obtidos por simulação

¹ Available at: <https://www.cisco.com/c/en/us/products/collateral/routers/4000-series-integrated-services-routers-isr/data-sheet-c78-732542.html>

² Available at <https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-1000-series-switches/nb-06-catlk-ser-switch-ds-ce-en.html?oid=otren019232>

Tabela 207 – Parâmetros de entrada (MTTR/MTTF) adotados no estudo de caso.

Componente	MTTF (horas)	MTTR (horas)
Switch	698220	8
Gateway	480770	8
Sensors	300000	1
Supervisor	44957	1
Hardware (LS)	4765,793684	3,4702988846
Hardware (RS)	8760	8
Operational System (LS, RS)	2800	1
Hypervisor (LS, RS)	2900	1
Virtual Machine (LS, RS)	2880	0,0958333333
Device Manager (LS, RS)	700	1
Data Base (LS, RS)	1440	1

utilizando a ferramenta Mercury (SILVA et al., 2015). O cenário A apresenta a melhor disponibilidade (99,9199%), seguido pelo cenário B (99,3515%) e o cenário C (99,2932%). O cenário A tem 7,01 h/ano de inatividade, enquanto o cenário B tem 56,8 h/ano, e o cenário C, 61,91 h/ano. O cenário A tem a maior disponibilidade porque tem redundância de servidores (local server e remote server), portanto, se um dos dois servidores falhar, o outro pode suprir as necessidades do sistema. O cenário B estava mais disponível do que o C, influenciado pelo MTTR do hardware (ver Tabela 207). O MTTR no local server é cerca de 3h e 8h no remote server.

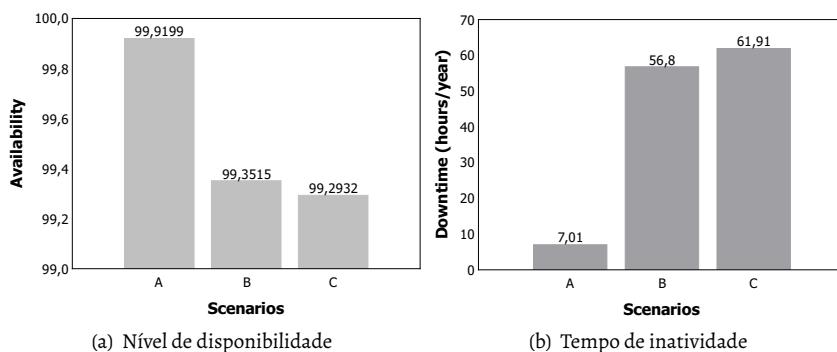


Figura 329 – Níveis de disponibilidade e tempo de inatividade em relação ao sistema de monitoramento de saúde para os cenários propostos.

A próxima etapa foi realizar a análise de sensibilidade. O MTTF e o MTTR foram variados em intervalos de 10%, até atingir 50% a mais e menos o valor base. A Tabela 208 apresenta os índices de sensibilidade para cada cenário em ordem decrescente. As transições que têm o impacto mais significativo sobre a disponibilidade aparecem no topo. O componente de software *data process* (DP) apresentou o maior índice de sensibilidade nos três cenários, o que significa que tal componente deve ser replicado para evitar a indisponibilidade do sistema.

Considerando o cenário A, algumas análises adicionais foram realizadas. Para melhor ilustrar a influência dos parâmetros, a Figura 330 representa a variação na disponibilidade com base na variação dos parâmetros. As figuras 330(b), 330(d), 330(e) e 330(f) mostram os parâmetros MTTF, onde a disponibilidade tende a aumentar proporcionalmente ao MTTF. As Figuras 330(a) e 330(c) mostram que aumentando o MTTR a disponibilidade diminui. A variação dos parâmetros fornece uma visão geral do comportamento do sistema. A disponibilidade é significativamente influenciada pelas taxas de falha e reparo dos componentes *data process*, *storage*, e *operating system*. Como todos esses componentes são software e geralmente são menos caros para replicar do que hardware, é um resultado interessante.

Tabela 208 – Índices de sensibilidade para todos os cenários.

Arquitetura base		Arquitetura com servidor local		Arquitetura com servidor remoto	
Variável	Índice	Variável	Índice	Variável	Índice
RS_DP_MTTR	1.52×10^{-5}	LS_DP_MTTR	2.34×10^{-3}	RS_DP_MTTR	2.26×10^{-3}
LS_DP_MTTF	1.51×10^{-5}	LS_HW_MTTF	1.95×10^{-3}	RS_DP_MTTF	1.88×10^{-3}
LS_S_MTTR	1.11×10^{-5}	LS_DP_MTTF	1.90×10^{-3}	RS_OS_MTTF	1.70×10^{-3}
RS_DP_MTTF	1.09×10^{-5}	LS_OS_MTTF	1.70×10^{-3}	RS_S_MTTR	1.52×10^{-3}
RS_OS_MTTF	1.06×10^{-5}	LS_S_MTTR	1.61×10^{-3}	RS_HV_MTTF	1.20×10^{-3}
LS_OS_MTTF	1.03×10^{-5}	LS_HV_MTTF	1.20×10^{-3}	RS_HW_MTTF	1.16×10^{-3}

29.7 Conclusões e Trabalhos Futuros

Este trabalho adotou redes de Petri estocásticas para avaliar o desempenho e a disponibilidade de um sistema hospitalar inteligente sem ter que investir em equipamentos reais previamente. Os modelos são bastante paramétricos, permitindo calibrar a capacidade dos recursos, os tempos de atendimento, os tempos entre falhas e os tempos entre reparos.

O primeiro modelo permite a configuração de 13 parâmetros, permitindo avaliar um grande número de cenários diferentes. A análise mostrou que o tempo de chegada é um parâmetro essencial no sistema. Em diferentes cenários, foi possível observar a estreita relação entre MRT, utilização de recursos e taxa de descarte, principalmente para altas taxas de chegada. Outro aspecto importante foi a validação que demonstrou que o modelo representa o mesmo comportamento do sistema real. Isso pode ser observado a partir dos resultados do modelo e do experimento real que estão dentro do intervalo de confiança.

Três cenários foram explorados considerando o segundo modelo. Os maiores resultados de disponibilidade foram observados no cenário A, composto por redundância de servidores (local e remoto). Tal cenário — com redundância — apresentou uma disponibilidade de 99,9199%, ou seja, 7,01 h/ano de inatividade. A análise de sensibilidade também foi explorada para identificar os componentes que têm o impacto mais significativo na disponibilidade do sistema. Assim, o impacto de diferentes valores de MTTF e MTTR para cada componente foi medido. A análise de sensibilidade mostrou que componentes específicos têm um impacto maior na disponibilidade do sistema. O componente de software de processamento de dados, por exemplo, apresentou alto índice de sensibilidade nos três cenários, o que significa que tais componentes devem ser replicados para evitar indisponibilidade do sistema.

Portanto, este trabalho pode auxiliar os administradores de sistema na identificação do equipamento mais adequado em termos de custo e eficiência. Como trabalhos futuros, pretendemos realizar mais análises numéricas, avaliando diferentes aplicações. Além disso, pretendemos combinar os dois modelos (desempenho e disponibilidade) no mesmo modelo e estender os cenários possíveis.

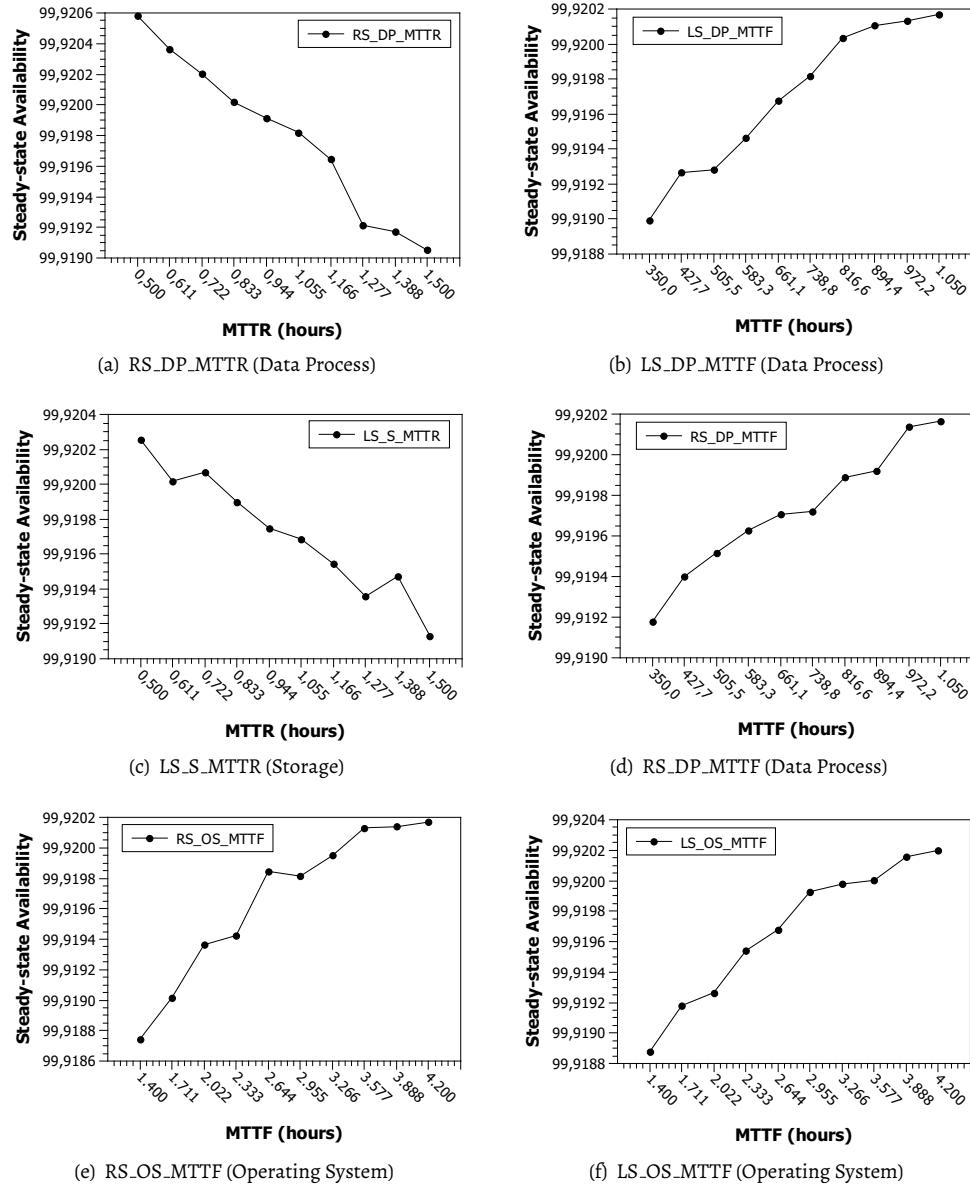


Figura 330 – Resultados de disponibilidade variando MTTR/MTTF.

30

CAPÍTULO

Metodologia para Recomendação de Rotas de Evacuação em Cenário de Desastre Natural

Autores: Orrana Lhaynher Veloso de Sousa; Deborah Maria Vieira Magalhães.

Resumo: Devido a geodinâmica terrestre, desastres naturais são fenômenos recorrentes que evidenciam a fragilidade da vida humana e não humana. Dentre estes, podemos citar incêndios, furacões, terremotos e outros, que destroem diretamente ou indiretamente infraestruturas, causando danos sociais que afetam indivíduos, cidades e países. O homem desenvolve métodos e tecnologias, como planejamento adequado e sistemas de notificação, a fim de prevenir ou reduzir os impactos de desastres. Como resultado desses fenômenos, indivíduos são levados à miséria, ficando sem moradia, saneamento básico e alimentos. Assim, a manutenção da vida é certamente o principal propósito de residentes, que buscam fugir de suas localidades através de rotas de evasão. Contudo, a definição de uma rota de fuga não é uma tarefa trivial devido a, por exemplo, congestionamentos, bloqueio de vias, abrigos indisponíveis, entre outros. Então, este trabalho propõe um método para a sugestão de rotas e simulação de demanda de evacuação baseada em agente. Essa sugestão será feita com a definição dos pontos de origem - através de coordenadas na área afetada, e pontos de destino - através da seleção do abrigo mais próximo e com disponibilidade de vagas de cada evacuado. Após isso, a recomendação será feita levando em consideração a disponibilidade de vias na rede rodoviária e a redução de possíveis congestionamentos, recomendando assim a melhor rota para cada evacuado nas circunstâncias dadas. Para testar a eficácia do método, é apresentada a simulação de um estudo de caso para o incêndio florestal *Camp Fire* no condado de *Butte* na Califórnia que aconteceu em novembro de 2018. Os resultados do estudo de caso demonstram que o método pode fornecer rotas de evasão que impactam positivamente no tempo de evacuação global.

Palavras-chaves: desastre natural, simulação baseada em agente, método de evacuação.

30.1 Introdução

Desastres naturais são fenômenos que ocorrem em função das mudanças climáticas que acontecem em nosso planeta. Dentre estes, desastres climáticos e geofísicos, como terremotos e tsunamis, mataram 1,3 milhão de pessoas nos últimos 20 anos e deixaram mais 4,4 bilhões de feridos, desabrigados ou em necessidade de ajuda de emergência, segundo [ONUBR \(2018\)](#). Estes são números estratégicos, se forem

considerados o que eles significam em termos de oportunidades destruídas, de pessoas que perderam suas vidas, casas, escolas e hospitais.

Em cenários de desastre, o principal objetivo da população é escapar da área afetada, consequentemente poupando vidas. De acordo com Mattar (2011), o deslocamento humano causado por efeitos climáticos adversos tem sido por muito tempo uma estratégia natural de adaptação à variação do meio ambiente. Dentre os tipos de deslocamentos populacionais, a migração intra-urbana se destaca, sendo feita através de estradas, na maioria dos casos, que podem levar para destinos seguros, fora da área de risco, e/ou para abrigos de segurança.

Entre as escolhas de destino de evacuação, deve-se destacar a indicação de abrigos como rota final. Esta recomendação tem como fatores propícios a localização em uma área segura e a disponibilidade de suprimentos, refeições, serviços médicos e outros serviços de apoio. Para aqueles que são incapazes de chegar a suas casas ou cujas casas são impossíveis de viver, esta opção mitiga a preocupação de onde alojar-se e oferece uma direção do que fazer após um desastre.

Com isso, deve-se considerar a rota a ser tomada e qual abrigo a ser recomendado, pois estes podem influenciar diretamente no tempo gasto na evasão. Em casos assim, a escolha errônea da rota e do abrigo terá grande impacto no número de vítimas em desastres naturais. Além disso, em áreas densamente povoadas aumenta-se também a probabilidade de que não haja vagas e, consequentemente, recursos disponíveis para todos, fazendo com que a escolha do abrigo seja um fator de extrema importância. Desse modo, um algoritmo de sugestão de rotas em que se considere vários fatores se faz imprescindível, já que considerar apenas um aspecto em uma evasão num cenário de desastre pode afetar a segurança dos evacuados.

Congruente com a importância da recomendação, a simulação deste tipo de cenário traz importantes benefícios, visto que possibilita a avaliação de métodos de evacuação e seus possíveis resultados. Em função disso, diversas ferramentas de simulação têm sido desenvolvidas, como o Simulador de Transporte Multiagente, também conhecido como MATSim. Este simulador contém características relevantes para a simulação de evacuação em desastres naturais, tais como a modelagem baseada em agentes, a capacidade de suportar cenários em larga escala e a criação de uma solução ideal para o cenário em tempo aceitável (MATSIM, 2019).

30.1.1 Objetivos

Este trabalho tem como objetivo desenvolver um método de recomendação de rotas que auxilie na evacuação de residentes em desastres naturais, tendo como fator de tomada de decisão a lotação e distância de abrigos, a disponibilidade de vias e a redução de congestionamentos.

Os objetivos específicos são:

1. Possibilitar a evacuação em um tempo considerável;
2. Recomendar a rota adequada para o usuário considerando vias e abrigos disponíveis;
3. Reduzir o tempo gasto em congestionamentos;
4. Avaliar o desempenho da metodologia utilizada através da simulação do cenário de desastre para a região do Condado de Butte, Califórnia.

30.1.2 Metodologia

A metodologia do presente trabalho reside na implementação de um método de sugestão de rota de evacuação considerando os parâmetros lotação e distância de abrigos, disponibilidade de vias e redução de congestionamentos, com o intuito de minimizar o número de vítimas. Ela está dividida em três etapas: definição de ponto de origem, definição de ponto de destino e recomendação de rota. Em complemento, uma simulação de demanda de evacuação foi construída para o incêndio florestal *Camp Fire*, Califórnia, com o propósito de analisar e validar o método.

30.1.3 Contribuições

As contribuições desse trabalho se dividem em duas categorias: contribuições para a criação de cenários de desastre e contribuições para o simulador MATSim. Na primeira categoria, as contribuições foram:

1. Modelagem de um cenário com base em um desastre real;
2. Realização de um cronograma de evacuação;
3. Definição de lotação em pontos de abrigos;
4. Recomendação de rota considerando estradas disponíveis, congestionamentos e disponibilidade de abrigos.

Já na segunda categoria, a contribuição feita foi:

1. Criação de um algoritmo que remove rotas indisponíveis na rede rodoviária.

30.1.4 Estrutura do trabalho

Este capítulo está estruturado da seguinte forma:

- **Seção 2** - são apresentados os conceitos importantes sobre este trabalho;
- **Seção 3** - são mencionados os trabalhos correlatos a este;
- **Seção 4** - está exposta em detalhes a metodologia de desenvolvimento deste trabalho;
- **Seção 5** - é apresentado a conclusão deste projeto.

30.2 Referencial teórico

Nesta seção serão descritos conteúdos relevantes para o entendimento do trabalho. Na subseção 2.1 serão discutidos os métodos de roteamento de evacuação. Já na subseção 2.2, será apresentada a Modelagem baseada em agentes. Na subseção 2.3, será abordado o *OpenStreetMap*, um mapa livre e editável do mundo. Na subseção 2.4 e 2.5, o algoritmo de Dijkstra e o algoritmo A* são comentados, respectivamente. Enquanto que na subseção 2.6, o Simulador de Transporte Multiagente que será a *framework* de simulação é abordado. Por fim, na subseção 2.7, a ferramenta de visualização utilizada no projeto é discutida.

30.2.1 Métodos de roteamento de evacuação

De acordo com [Shahabi e Wilson \(2014\)](#), os métodos de roteamento de evacuação podem ser divididos em métodos descritivos e prescritivos. Os métodos descritivos são soluções que simulam visualmente uma dada situação de emergência. O objetivo é reproduzir a realidade o mais precisamente possível. Alguns dos métodos que se enquadram nessa categoria são: simulações baseadas em fluxo, simulações baseadas em agentes e simulações baseadas em atividades. O uso desse método traz benefícios à pesquisadores e desenvolvedores na área, pois permite a avaliação de políticas de evacuação antes de serem colocadas em prática em um ambiente similar ao real. Com isso, o maior número de pessoas pode ser salvo ao se adotar uma política que satisfaça objetivos escolhidos para uma evacuação em particular.

De outro modo, os métodos prescritivos determinam as estratégias ideais de roteamento de evacuação para alcançar algum objetivo de evacuação sem obrigatoriamente executar uma simulação. Ou seja, este tipo de método fornece um plano de evacuação em vez de visualizar um plano. Esses métodos usualmente utilizam técnicas matemáticas para coletar dados e recomendar rotas e/ou horários. O propósito é minimizar o tempo de evacuação, reduzir o congestionamento do tráfego ou diminuir a exposição ao perigo. Consequentemente, essas soluções são menos dependentes da geografia e geralmente são mais escaláveis.

30.2.2 Simulação baseada em agentes

Na Simulação Baseada em Agentes, componentes ativos ou tomadores de decisão são modelados e implementados usando conceitos e tecnologias relacionados a agentes. Ela foi fundamentada a partir de um método de modelagem de sistemas complexos que define regras e comportamentos para componentes individuais (agentes), bem como o ambiente em que eles estão presentes. Cada agente então avalia individualmente sua situação e toma decisões com base no seu conjunto de regras definido. Com isso, o curso e comportamento de eventos podem ser observados através das interações entre eles.

Nesse tipo de simulação, agentes podem ser pessoas, animais, veículos, equipamentos, células, componentes de um sistema, etc. Eles são capazes de evoluir, permitindo que comportamentos imprevistos surjam, podendo exibir padrões de comportamento complexos e fornecendo informações valiosas sobre a dinâmica do sistema do mundo real que ele emula. A modelagem em que se baseia este tipo de simulação às vezes incorpora redes neurais, algoritmos evolutivos ou outras técnicas de aprendizado para permitir aprendizado e adaptação realistas.

30.2.3 OpenStreetMap

O *OpenStreetMap* (OSM) é um projeto de mapeamento colaborativo para criar um mapa livre e editável do mundo, [OpenStreetMap \(2019\)](#). É constituído por dados abertos: qualquer pessoa tem a liberdade de usar os dados para qualquer fim. Ele fornece dados a centenas de sites na *Internet*, aplicações de celular e outros dispositivos. Os mapas foram desenvolvidos e são mantidos com rigor por sua comunidade voluntária de mapeadores, que inserem e revisam dados de receptores GPS portáteis, fotografias aéreas, imagens de satélite e outras fontes livres.

A grande disponibilidade de informações de localização de alta qualidade propiciou o mapeamento do mercado de massa com base em receptores GPS acessíveis, computadores domésticos e a *Internet*. Embora uma série de projetos baseados em mapeamento criado pelo usuário tenha aparecido, "o OSM é provavelmente o projeto mais extenso e efetivo atualmente em desenvolvimento", [Haklay e Weber \(2008, p 2\).](#)

30.2.4 Algoritmo Dijkstra

O algoritmo de Dijkstra, concebido pelo cientista da computação holandês Edsger Dijkstra em 1956 e publicado em 1959¹, soluciona o problema do caminho mais curto num grafo dirigido ou não dirigido com arestas de peso não negativo, em tempo computacional $O(m + n \log n)$ ², onde m é o número de arestas e n é o número de vértices . Este algoritmo calcula o caminho mais curto de acordo com o custo mínimo entre as vértices de um grafo. Escolhido um vértice como raiz da busca, o algoritmo calcula o custo mínimo deste vértice para todos os demais vértices do grafo.

30.2.5 Algoritmo A*

O A*, ou "A-star", é um algoritmo de busca de caminho e de grafo, que é frequentemente usado em ciência da computação devido ser um algoritmo completo e com eficiência ideal. Uma grande desvantagem prática é sua $O(b^d)$ complexidade do espaço, pois armazena todos os nós gerados na memória. Assim, em sistemas práticos de roteamento de viagem, geralmente é superado por algoritmos que podem pré-processar o gráfico para obter melhor desempenho, bem como abordagens limitadas à memória; no entanto, A* ainda é a melhor solução em muitos casos, de acordo com [Zeng e Church \(2009\)](#).

30.2.6 Simulador de transporte multiagente

O Simulador de Transporte Multiagente, em inglês *Multi-Agent Transportation Simulator* (MATSim), é uma ferramenta que oferece uma estrutura para realizar simulações de transporte baseadas em agentes em larga

¹ **Algoritmo de Dijkstra.** Disponível em <http://bit.ly/334RISE>. Acesso em 20 de novembro e 2019.

² **Algoritmo de Dijkstra!**; eximiaCo.tech. Disponível em <http://bit.ly/337ggDX>. Acesso em 20 de novembro de 2019.

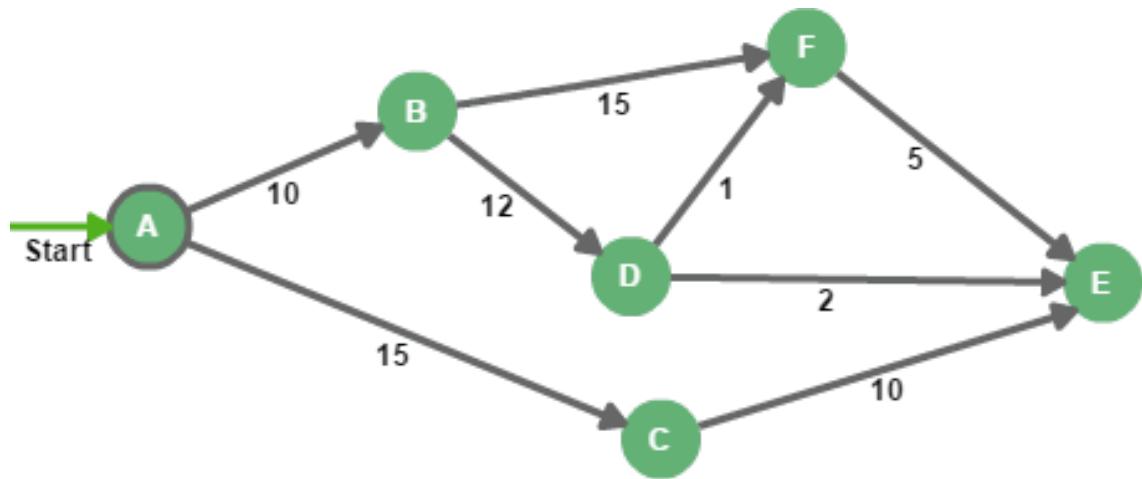


Figura 331 – Algoritmo de Dijkstra. Fonte: [Baeldung \(2019\)](#).

escala. Cada agente (pessoa) otimiza repetidas vezes seu cronograma diário de atividades ao mesmo tempo que compartilha espaços na infraestrutura de transporte com os demais agentes ao longo da simulação. Tal cronograma representa uma lista de atividades, como, por exemplo, estar em *casa* ou no *trabalho* e se deslocar de *carro* até o *shopping*). O cronograma conta ainda com informações temporais, por exemplo, sair de casa às 7:00, e informações adicionais, como a rota detalhada de casa pro shopping. Em uma simulação típica do MATSim, os dados referentes a trajetória a qual se deseja simular são otimizados em uma rede de transporte definida, esse é um dos recursos fundamentais do simulador, tornando-o adequado para uso em estudos de políticas de evacuação.

30.2.6.1 Modelo de fluxo de tráfego

O MATSim oferece dois mobsim internos: Simulação Baseada em Fila (QSim - *Queue Simulation*) e Simulação de Fila de Eventos Discretos Java (JDEQSim - *Java Discrete Event Queue Simulation*); além disso, simulações de mobilidade externa podem ser adicionadas. Ele utiliza a abordagem baseada em fila computacionalmente eficiente, como mostrado na Figura 332. Um carro que entra em um *link* de rede (ou seja, um segmento de estrada) a partir de um cruzamento é incluído no fim da fila de espera. Ele mantém-se lá até que o tempo para percorrer o *link* com fluxo livre tenha passado, ele esteja no início da fila de espera e o próximo *link* permita entrada.

O modelo de fluxo de tráfego do MATSim é vigorosamente baseado nos dois atributos de *link*: capacidade de armazenamento e capacidade de fluxo. A capacidade de armazenamento define o número de carros que cabem em um *link* de rede, enquanto a capacidade de fluxo especifica a capacidade de saída de um *link*, ou seja, quantos viajantes podem deixar o respectivo *link* por etapa de tempo, sendo assim um atributo individual do *link*.

30.2.6.2 Algoritmo co-evolutivo do MATSim

O equilíbrio do MATSim é obtido através de um algoritmo co-evolutivo. Esses algoritmos co-evoluem espécies distintas sujeitas a interação (por exemplo, competição). No MATSim, os indivíduos são descritos por seus planos, onde uma pessoa simboliza uma espécie. Com o algoritmo co-evolutivo, a otimização é realizada em termos de planos dos agentes, ou seja, em todo o plano diário de atividades e viagens. Ele obtém mais do que os equilíbrios de tráfego padrão, que desconsideram as atividades. Eventualmente, é obtido um equilíbrio, sujeito a restrições, onde os agentes não podem melhorar ainda mais seus planos unilateralmente.

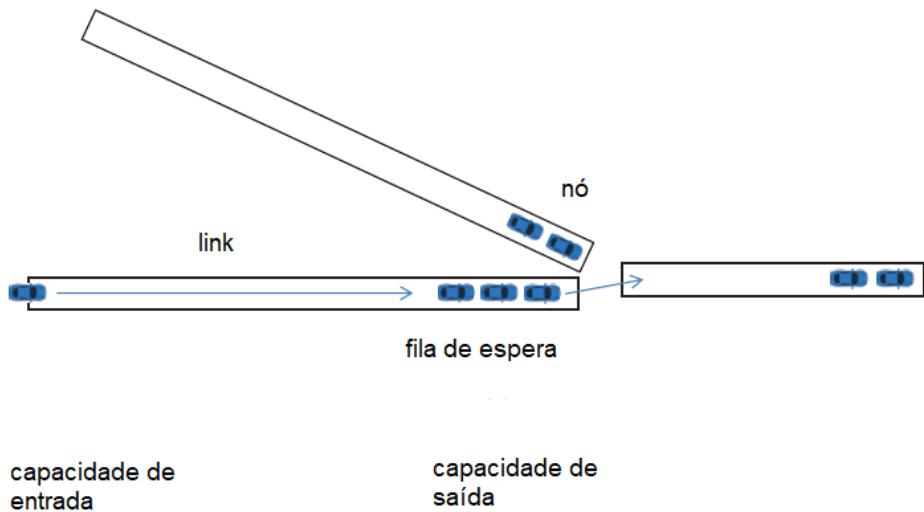


Figura 332 – Modelo de fluxo de tráfego do MATSim. Adaptado de [Horni e Axhausen \(2019\)](#).

O MATSim não utiliza um algoritmo evolutivo ao invés de um algoritmo co-evolutivo, pois o primeiro levaria a um sistema ideal, já que a otimização seria empregada com uma função de adequação global (ou populacional). Em vez disso, o algoritmo co-evolutivo provoca um equilíbrio (estocástico) do usuário, pois a otimização é realizada em termos de funções de pontuação individuais e dentro do conjunto de planos de um agente.

30.2.6.3 Etapas principais do processo de simulação e otimização

Conforme ilustrado na Figura 334, as cinco etapas principais são: demanda inicial, execução, pontuação, replanejamento e análise.

Na primeira etapa, a *Demanda* inicial descreve o comportamento de mobilidade a ser simulado. Abrange uma lista completa de agentes com seus respectivos cronogramas diários. Na etapa de execução, também conhecida como Simulação de Mobilidade (mobsim) todos os planos dos agentes são executados juntos. Isto é, agentes e veículos são simulados na rede rodoviária do cenário. Desse modo, um agente pode impactar em outros agentes, podendo ocasionar, por exemplo, engarrafamentos se vários agentes decidem transitar pela mesma estrada.

Após a execução, dá-se início à etapa de *Pontuação*, onde cada plano é avaliado e recebe um determinado número de pontos. A função de pontuação é customizável mas, por padrão, o tempo gasto nas atividades aumenta a pontuação, enquanto o tempo gasto na viagem o diminui. Desse modo, agentes em engarrafamentos perdem pontos, enquanto agentes que têm viagens rápidas e curtas acumulam mais pontos, executando atividades por um maior período. Para a função padrão, a utilidade de um plano (S_{plan}) é calculada como a soma de todas as utilidades de atividade ($S_{act,q}$) mais a soma de todas as (des)utilidades de viagem ($S_{trav,mode(q)}$) com N como o número de atividades, como mostrado na Equação 30.1. A $(trip)$ q é a viagem que segue a atividade q . Para a pontuação, a última atividade é mesclada com a primeira para produzir um número igual de viagens e atividades.

$$S_{plan} = \sum_{q=0}^{N-1} S_{act,q} + \sum_{q=0}^{N-1} S_{trav,mode(q)}. \quad (30.1)$$

Como discutido anteriormente, agentes podem ser influenciados por outros, provocando congestionamentos em vias. Portanto, na etapa de *Planejamento*, os agentes podem modificar seus planos visando

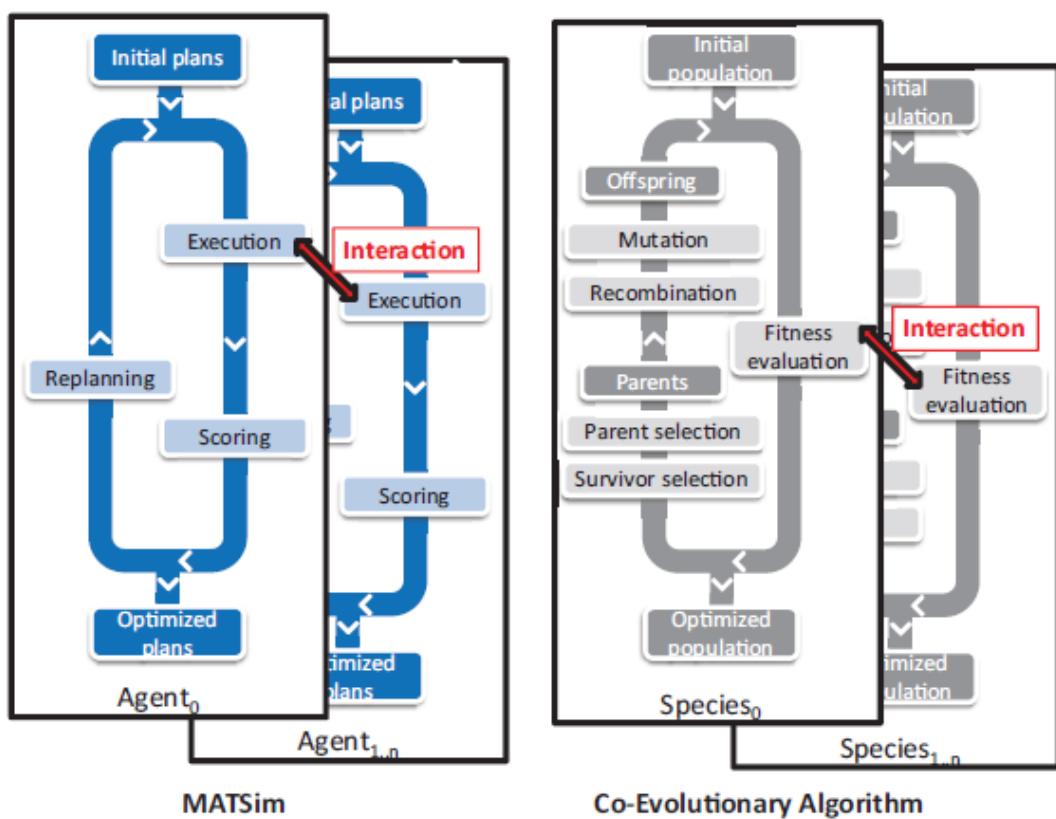


Figura 333 – Algoritmo co-evolucionário do MATSim. Fonte: [Horni e Axhausen \(2019\)](#).

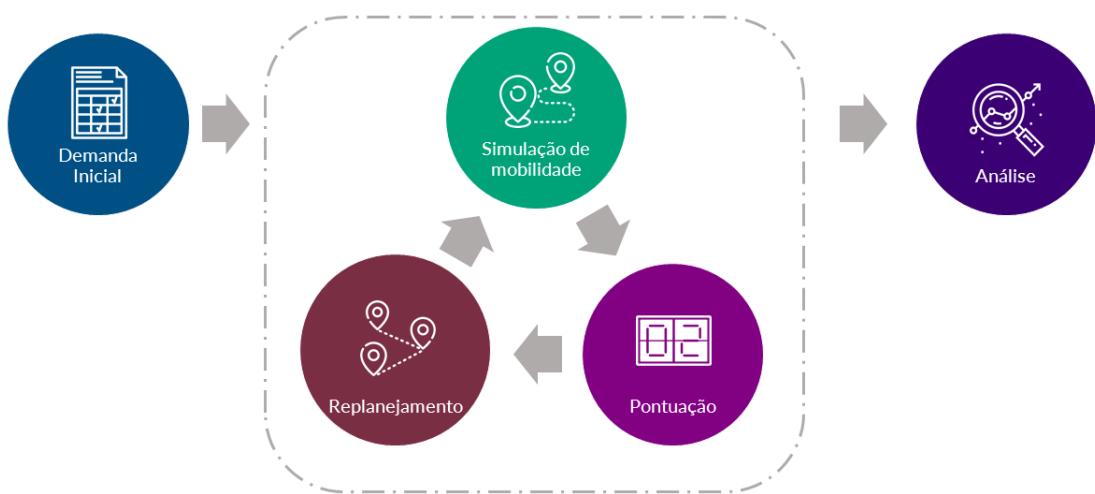


Figura 334 – O ciclo de vida da simulação no MATSim. Adaptado de [Horni e Axhausen \(2019\)](#).

melhorar as suas respectivas pontuações. Dessa forma, horários de inicio e fim de atividades podem ser alterados na simulação de mobilidade a fim de evitar situações que levem a pontuações ruins.

Na etapa de *Análise*, as ações ou eventos realizados na simulação são agregados para avaliar qualquer medida na resolução desejada, como compartilhamentos de modo (carro, táxi, bicicleta), total de milhas percorridas pelos agentes, duração média das viagens e distância por modo e hora. Tais análises podem ser executadas automaticamente no final ou em uma etapa separada qualquer de pós-processamento.

30.2.6.4 Arquivos de entrada

Para realizar uma simulação, os seguintes arquivos precisam necessariamente serem configurados no MATSim:

- o arquivo de configuração, que descreve o comportamento da simulação;
- o arquivo de rede, que descreve a rede rodoviária da área a ser simulada;
- e o arquivo de população, que descreve as demandas de viagem de acordo com o cenário que se deseja simular.

O MATSim utiliza o arquivo de configuração para a construção de uma ligação entre o simulador e o usuário, controlando como a simulação será feita. Cada parâmetro faz parte de exatamente um grupo de configurações, como, por exemplo, o parâmetro *inputNetworkFile* mostrado no Código 30.1. Esse grupo coleta parâmetros semelhantes ou diferentes para uma parte específica do MATSim. Configurações como sistema de coordenadas, número de iterações e tipos de atividades são definidas neste arquivo.

```
;config
  ;module name="network"
    ;param name="inputNetworkFile" value="network.xml" /
  ;/module
  ;module name="plans"
    ;param name="inputPlansFile" value="plans.xml" /
  ;/module
  ;module name="controler"
    ;param name="firstIteration" value="0" /
    ;param name="lastIteration" value="0" /
  ;/module
  ;module name="planCalcScore"
    ;parameterset type="activityParams"
      ;param name="activityType" value="h" / ;!-- home --
      ;param name="priority" value="1" /
      ;param name="typicalDuration" value="12:00:00" /
      ;param name="minimalDuration" value="08:00:00" /
    ;/parameterset
  ;/module
;/config
```

Código Fonte 30.1 – Exemplo de arquivo de configuração.

O arquivo *network.html* descreve a rede em que agentes ou veículos se locomovem, como apresentado no Código 30.2. Consiste de nós e *links*, ou vértices e arestas na teoria dos grafos. Cada nó é representado por valores de coordenadas x e y, enquanto *links* têm um número maior de características. Dentre essas características podem-se citar atributos como *from* e *to* que descrevem a geometria da rede. Além disso, todos os links são unidirecionais. Se uma via puder ser percorrida nas duas direções, dois links devem ser estipulados com os atributos *from* e *to* contrários.

```
;network
  ;nodes
    ;node id="0" x="-200.0" y="200.0" /
    ;node id="1" x="0.0" y="200.0" /
    ;node id="2" x="200.0" y="200.0" /
    ;node id="3" x="400.0" y="400.0" /
  ;/nodes
  ;links cpperiod="01:00:00" effectivelcelsize="7.5" effectivelanewidth="3.75" /
    ;link id="0_1" from="0" to="1" length="200.0" freespeed="222.222222222223"
```

```

    capacity="3600.0" permlanes="1.0" oneway="1" modes="car" i
  i/link{
  i/link id="1'2" from="1" to="2" length="200.0" freespeed="222.2222222222223"
  capacity="3600.0" permlanes="1.0" oneway="1" modes="car" i
  i/link{
  i/link id="2'3" from="2" to="3" length="200.0" freespeed="3.33889816360601"
  capacity="1800.0" permlanes="1.0" oneway="1" modes="car" i
  i/link{
  i/links{
  i/network{

```

Código Fonte 30.2 – Exemplo de arquivo de *network*.

A demanda de viagens do MATSim é descrita pelos planos diários dos agentes presentes no arquivo de população, como exposto no Código 30.3. Por isso, o arquivo de população contém essencialmente uma lista de planos diários em uma estrutura hierárquica. A população contém uma lista de pessoas, cada pessoa tem uma lista de planos, e cada plano tem uma lista de atividades. Apenas um plano é escolhido. O plano selecionado de cada agente é executado pelo mobsim, e pode ser substituído por outro durante a fase de replanejamento.

```

ipopulation{
  iperson id="car'home1'work1'0"{
    iplan selected="yes"{
      i/activity type="home" x="-1188391.227582054" y="114456.09067770051"
      end_time="09:00:00" i /activity{
      i/leg mode="car" i /leg{
      i/activity type="work" x="4663945.2848932855" y="0.0"
      end_time="17:00:00" i
      i/activity{
      i/leg mode="car" i /leg{
      i/activity type="home" x="-1188391.227582054" y="114456.09067770051"
      end_time="09:00:00" i /activity{
    ipлан{
    iperson{
  ipopulation{

```

Código Fonte 30.3 – Exemplo de arquivo de população.

30.2.6.5 Equilíbrio de *Nash*

O equilíbrio de *Nash* é calculado através da repetição do comportamento de melhor resposta: em todas as iterações, uma fração dos agentes recalcula uma rota ou um destino com base no que seria melhor na iteração anterior, assumindo que o comportamento de todos os outros agentes permaneça inalterado. Após esse replanejamento, os planos resultantes de todos os agentes são executados simultaneamente na simulação de mobilidade e novas medidas de desempenho são computadas. Este processo é repetido várias vezes. Uma vez estabilizado, nenhum agente pode melhorar substancialmente através de um replanejamento de rota ou destino e um equilíbrio aproximado de *Nash* é obtido. Com o destaque para simulações de evacuação, o equilíbrio de *Nash* faz com que tempos mais curtos de evacuação sejam alcançados, onde toda a população tenta encontrar uma rota de evacuação ideal por meio do aprendizado iterativo.

30.2.7 Via

O Via é um *software* independente que possibilita a visualização e análise de dados do MATSim, bem como conjuntos de dados espaciais e temporais genéricos. Os dados visualizados no Via são estruturados em camadas. Cada camada é responsável por visualizar um conjunto de dados. Assim, ele fornece camadas para visualizar redes, instalações, atividades e viagens de agentes, paradas de transporte público ou apenas pontos genéricos no espaço, como uma coordenada qualquer.

30.3 Trabalhos relacionados

Nesta seção são apresentados trabalhos relevantes e relacionados a este projeto ordenados cronologicamente. Como fator de comparação foram utilizadas as seguintes métricas: objetivos, simulação, ferramenta

utilizada caso tenha sido feita simulação, recomendação de abrigos, se leva em consideração lotação dos abrigos e controle de congestionamentos. A comparação é feita na Tabela 209.

30.3.1 Trabalhos selecionados

Há várias maneiras de implementar um método de recomendação de rotas. Assim, serão citados trabalhos referentes à métodos de recomendação de rotas com foco em cenários de desastres naturais.

Com o propósito de abordar o uso de algoritmos evolutivos multiobjetivo (MOEA) e o sistema de informações geográficas (GIS) para o planejamento de evacuação, [Saadatseresht e Taleai \(2009\)](#) propuseram uma abordagem em três etapas para o planejamento de evacuação. No primeiro passo, locais seguros adequados para evacuação são selecionados/projetados. Estes locais foram adotados do plano abrangente para a gestão de desastres de Teerã projetado pela *Japan International Cooperation Agency* (JICA) e a *Tehran Disaster Mitigation and Management Center* (TDMMC). Na segunda etapa, as áreas seguras candidatas apropriadas são escolhidas para cada bloco de construção. Esta seleção é baseada na proximidade das áreas seguras disponíveis aos blocos. Na terceira etapa, é feita uma distribuição ideal das pessoas considerando-se capacidade de estradas e distância para as áreas seguras. Seguindo essas três etapas, a área segura adequada e o caminho ideal para entrar na área segura são determinados para cada bloco de construção.

No artigo de [Stepanov e Smith \(2008\)](#), o projeto e a análise ótima das rotas de evacuação em redes de transporte são examinados. Uma metodologia para projetar políticas de roteamento ótimas para o planejamento de evacuação de emergência (EEP) é sugerida. Uma formulação do modelo de programa inteiro (IP) foi apresentada para gerar uma atribuição ótima de rota para um EEP estocástico. Este modelo de IP utiliza modelos de fila M/G/c/c de taxa de serviço de decaimento dependente do estado para capturar funções de atrasos de tempo em ligações rodoviárias. As medidas de desempenho para a política de evacuação gerada, como o tempo de liberação, a distância percorrida total e o nível de congestionamento, são avaliadas com o software de simulação MGCCSimul. Embora a política derivada com o modelo EEP IP leve a uma maior distância total percorrida, ela minimiza simultaneamente o nível de bloqueio e congestionamento, bem como o tempo de liberação. Em comparação com a política de IP original, a política de IP do EEP resulta em uma redução de 22% no tempo de liberação. O modelo EEP IP sugerido fornece solução analítica rápida e requer apenas uma avaliação com o programa de simulação.

O artigo de [Lämmel e Flötteröd \(2009\)](#) compara duas estratégias de roteamento diferentes em uma simulação com vários agentes de um grande cenário de evacuação no mundo real. A primeira abordagem aproxima o equilíbrio de Nash, onde cada evacuado adota uma estratégia de roteamento ideal individualmente, independentemente do que essa solução impõe aos outros. A segunda abordagem minimiza aproximadamente o tempo total de viagem no sistema, o que exige impor o comportamento cooperativo dos evacuados. Ambas as abordagens são analisadas em termos da dinâmica de evacuação global e em um nível geográfico detalhado. Os resultados deste artigo mostram que soluções de roteamento cooperativo matematicamente motivadas podem ser obtidas com uma sobrecarga computacional aceitável, mesmo em um sistema puramente baseado em simulação. A abordagem de roteamento cooperativo apresentada, que é uma aproximação do "roteamento ótimo do sistema" bem conhecido no campo da atribuição dinâmica de tráfego, gera uma taxa de evacuação substancialmente mais alta do que uma estratégia de roteamento não-cooperativa alternativa.

[Coutinho-Rodrigues e Alçada-Almeida \(2012\)](#) propuseram um problema de otimização multiobjetivo para encontrar caminhos de evacuação e a localização de abrigos para planejamento de evacuação urbana. Os autores consideraram muitos objetivos para otimização, como comprimentos de caminho, riscos de caminho, tempos de evacuação, comprimentos de caminhos de *backup* e número de abrigos. O conjunto de rotas primárias entre o local do desastre e os locais do abrigo foi gerado com uma abordagem bi-objetiva do caminho mais curto considerando os tamanhos dos caminhos e os riscos do caminho. Seu algoritmo para gerar os pares de caminhos primários e de *backup* para cada setor foi implementado no ambiente GIS. O modelo foi testado simulando uma situação de incêndio no centro da cidade (Baixa) de Coimbra, Portugal, uma área com muitas ruas estreitas, poucas áreas abertas e edifícios muito antigos. Dez soluções foram geradas e comparadas usando tabelas, gráficos e mapas codificados por cores em um ambiente GIS. A solução ideal foi determinada minimizando cada objetivo individualmente. Soluções adicionais

foram obtidas usando o método de ponderação, minimizando a distância para a solução ideal e usando um procedimento hierárquico.

Com o objetivo de avaliar e otimizar a eficiência de uso espaço-temporal de um plano de evacuação, Fang et al. (2013) propõem uma métrica de avaliação e a utiliza para a otimização de planos de evacuação. Primeiro, foi definida uma métrica de eficiência de uso espaço-tempo com base em trajetórias em segmentos de estradas e interseções. Em segundo lugar, foi proposto um algoritmo híbrido de otimização multiobjetivo de duas camadas para planejar as direções do movimento de conversão de veículos e pedestres em uma rede integrada de rodovias e construção de interiores com a finalidade de tomar decisões sobre planos de evacuação. Foram analisados os resultados otimizados de três perspectivas diferentes, ou seja, soluções de Pareto, curvas de evacuação e eficiências de uso espaço-tempo em interseções e *links*. Os experimentos computacionais deste estudo demonstram que as interseções em uma rede de evacuação precisam ser atribuídas a direções razoáveis de movimento de rotação de veículos e pedestres. Uma comparação entre as soluções propostas calculadas e uma solução TRANSIMS (Sistema de Simulação de Análise de Transportação) mostra que a abordagem proposta é adequada e eficaz para organizar veículos e pedestres no espaço e no tempo em um processo de evacuação.

No artigo de Kim e Lee (2017), uma rede de evacuação é projetada para desenvolver um plano de evacuação com base em um problema hierárquico de design de rede usado em redes de comunicação, transporte público e projetos de redes de organizações sociais. Análises experimentais foram realizadas para encontrar planos de evacuação eficazes em Haeundae Beach, Coréia, usando simulação de transporte de agentes múltiplos (MATSim). Para projetar uma rede de evacuação hierárquica eficaz, um projeto de rede de evacuação hierárquica é desenvolvido usando os conceitos de localização do *hub*, *cluster* e design de rede. Além disso, é considerado o contrafluxo na rede de evacuação, bem como a determinação das linhas do corredor pela regra de distância mínima, regra de fluxo máximo, regra de acesso máximo e regra de tempo de viagem mínimo. Como resultado da execução de cada cenário usando o MATSim, é avaliado um critério para estabelecer a linha do corredor. Uma rota de evacuação ideal é encontrada para minimizar as filas superlotadas de evacuados, quando o tempo total de evacuação da área do tsunami é de aproximadamente 3 horas e 30 minutos.

Benchekroun e Mouraouah (2018) propõem um plano de evacuação de tsunami para a cidade de Tânger-Marrocos para evacuação horizontal (quando os evacuados se deslocam das zonas de risco para áreas mais seguras ou áreas mais altas (colinas) localizadas fora da área de inundação). Este plano foi elaborado considerando a ameaça de tsunami de fonte tsunamigênica localizada na margem da Península Ibérica e usando os mapas de inundação do pior cenário para definir a área de inundação. O plano de evacuação é elaborado por meio da modelagem de rotas de evacuação usando a ferramenta *Closest Facility* implementada no software ArcMap 10.2 do sistema de informações geográficas (GIS) do ESRI. A modelagem de evacuação foi aplicada para encontrar as rotas mais eficazes, levando em consideração o menor tempo de viagem para evacuar as zonas de risco em caso de tsunami durante a baixa temporada. A modelagem foi alcançada usando dados de entrada detalhados da rede viária, abrigos de construção de evacuação ou áreas de montagem e distribuição espacial da população na zona de risco. Os resultados são apresentados em um plano de evacuação em que as rotas de fuga, locais seguros e pontos de montagem estão claramente marcados. Durante a resolução da análise, foram escolhidos atributos de restrição, incluindo limites de capacidade dos caminhos ou altos graus da encosta, para proibir alguns evacuados de atravessar determinadas estradas, a fim de evitar congestionamentos de pedestres.

Veeraswamy et al. (2018) estenderam o modelo de evacuação *buildingEXODUS* para modelar evacuações urbanas em larga escala, incluindo a rede de estradas e espaços abertos (por exemplo, parques, espaços verdes e praças) junto com os edifícios. Os resultados da simulação de evacuação foram acoplados aos resultados de um modelo de propagação de incêndios florestais e aplicados ao incêndio florestal de Swinley que ocorreu em Berkshire, Reino Unido em maio de 2011. Foram avaliados quatro procedimentos de evacuação diferentes nas rotas percorridas pelos pedestres, fornecendo estatísticas importantes sobre a evacuação como tempo para chegar ao local da montagem, a distância percorrida, o congestionamento experimentado pelos agentes e as margens de segurança associadas ao uso de cada rota de evacuação. Uma das principais conclusões deste trabalho é a importância de formular procedimentos de evacuação que identifiquem a população ameaçada, forneça um aviso de evacuação oportuno, identifique rotas apropriadas que mantêm

uma distância segura da frente de perigo, maximizando assim as margens de segurança mesmo com o custo de seguir rotas de evacuação mais longas. A simulação de evacuação oferece um meio de atingir esses objetivos.

No trabalho de [Lujak e Giordani \(2018\)](#) foi estudado a agilidade das rotas de evacuação em relação à mudança dinâmica de condições perigosas imprevisíveis em redes espaciais inteligentes. Com isso, eles propuseram duas novas métricas de importância do nó: evacuação entre centralidade de evacuação e centralidade de evacuação, ambas inspiradas na centralidade de evacuação. A primeira métrica representa a fração de rotas de evacuação eficientes entre todos os pares de origem-destino diferentes do nó especificado que passa por esse nó, enquanto a segunda representa a importância do nó especificado para a evacuação, considerando a disponibilidade de caminhos alternativos de evacuação eficientes (rotas) desse nó para saídas seguras. Além disso, dado um conjunto de posições e saídas seguras dos evacuados, eles encontraram rotas de evacuação ágeis mais curtas, onde, nesse trabalho, rota ágil representa a capacidade de redirecionar de forma eficiente e segura dos nós intermediários no caso de quedas imprevisíveis de segurança, maximizando o valor da centralidade da evacuação dos nós intermediários da rota. Além disso, propuseram um algoritmo para esse problema e discutiram sua capacidade de reagir às mudanças nas circunstâncias de segurança ao longo das rotas recomendadas.

Tabela 209 – Trabalhos relacionados.

Trabalhos Relacionados	Objetivos	Simulação?	Sim, qual a ferramenta?	Recomenda abrigos?	Considera lotação dos abrigos?	Controle de congestionamento
Stepanov e Smith (2008)	Projetar e analisar algoritmos de atribuição de rotas	Sim	MGCCSimul	Não	Não	Sim
Saadatseresh e Taleai (2009)	Abordar o uso de algoritmos para o planejamento de evacuação	Não	-	Não	Não	Não
Lämmel e Flötteröd (2009)	Comparar estratégias de roteamento em simulação de grande cenário de evacuação	Sim	MATSim	Não	Não	Sim
Coutinho-Rodrigues e Alçada-Almeida (2012)	Identificar caminhos de evacuação e a localização de abrigos	Não	-	Sim	Não	Não
Fang et al. (2013)	Fornecer método prático de medir eficácia de planos de evacuação	Sim	FIFO e TRANSIMS	Não	Não	Não
Kim e Lee (2017)	Projetar rede de evacuação para desenvolver plano de evacuação com base em problema de design de rede hierárquico	Sim	MATSim	Não	Não	Sim
Benchekroun e Mouraouah (2018)	Propor plano de evacuação de tsunami para a cidade de Tanger-Marrocos para evacuação horizontal	Não	-	Sim	Não	Sim
Veerawamy et al. (2018)	Estender modelo de evacuação buildingEXODUS para modelar evacuações urbanas em larga escala	Sim	Prometheus	Sim	Não	Sim
Lujak e Giordani (2018)	Fornecer método prático de medir eficácia de planos de evacuação.	Não	-	Não	Não	Sim
Este projeto	Recomendar rotas e abrigos disponíveis, reduzir tempo e minimizar congestionamentos	Sim	MATSim	Sim	Sim	Sim

A metodologia adotada neste trabalho difere dos trabalhos selecionados, fornecendo a recomendação de rotas para evacuados considerando-se a distância e disponibilidade de vagas em abrigos, a disponibilidade de vias no cenário e a redução de congestionamentos. Além disso, a simulação de evacuação do tráfego de veículos é realizada utilizando o simulador MATSim em um estudo de caso real. Essas contribuições terão então os seus desempenhos analisados e o impacto no tempo de evacuação total avaliado.

30.4 Proposta

Este trabalho consiste em desenvolver uma metodologia para a recomendação de rotas junto à simulação de um cenário de desastre com o software MATSim. Simultaneamente, para ilustrar a aplicação da metodologia desenvolvida, é feita sua aplicação na simulação da área do desastre *Camp Fire*, *Butte County* (CA), que aconteceu em 2018. Em ordem, apresentaremos o cenário de desastre utilizado como estudo de caso, discutiremos o método proposto e, por fim, Por fim, a simulação é comentada.

30.4.1 Cenário de desastre

No dia 08 de novembro de 2018 ocorreu o incêndio mais destrutivo na história da Califórnia, conhecido como *Camp Fire*. Este foi o desastre natural mais caro em 2018 no mundo em termos de perdas seguradas. Em janeiro de 2019, o dano total foi estimado em US \$ 16,5 bilhões; um quarto dos danos não estava seguro, [LosAngelesTime \(2019\)](#). Além de perdas financeiras, o incêndio causou a morte de 85 civis e feriu outros 12 civis e 5 bombeiros, [FOX40 \(2019\)](#). Com isso, o *Camp Fire* também é o incêndio florestal mais mortal nos Estados Unidos desde o incêndio de Cloquet em 1918 e está no topo da lista dos incêndios florestais mais mortais do mundo.



Figura 335 – Incêndio *Camp Fire* visto do satélite Landsat 8, em 08 de novembro de 2018. Escala: 8 km por cm. Fonte: [SFGATE \(2018\)](#).

Este incêndio teve uma rápida propagação, que teve como força motriz os ventos extremos que assolaram a região. Com rajadas de até 100 quilômetros por hora, estes ventos empurraram o fogo com uma velocidade rápida para o sul da Califórnia, [WIRED \(2018\)](#). Rajadas fortes também direcionaram faíscas e fumaça para as comunidades a favor do vento. Esse foi um dos problemas para extinguir o *Camp Fire*, que estava situado de modo que os ventos vindos do Nordeste soprassem a fumaça para a área da baía, a 160 quilômetros de distância.

Além da destruição e mortes causadas, essa fumaça atingiu uma grande área, como visto nas Figuras 335 e 336. A fumaça resultou em poluição do ar difundido em toda a Área da Baía de São Francisco e Vale Central, o que levou ao fechamento de escolas públicas em cinco municípios da área da baía e dezenas de bairros da área metropolitana de Sacramento em 16 de novembro. A fumaça era visível em lugares distantes como a cidade de Nova York, depois que nuvens de fumaça percorreram uma distância de mais de 3.000 milhas (aproximadamente 4.828,032 quilômetros). Além disso, este incêndio cobriu uma área de 153.336 acres, como mostrado na Figura 337 e destruiu 18.804 estruturas, com a maior parte dos danos ocorrendo nas primeiras quatro horas.

30.4.2 Método

A Figura 338 descreve a metodologia seguida para alcançar os resultados obtidos. Nas subseções abaixo é explicado o passo a passo dessa metodologia.

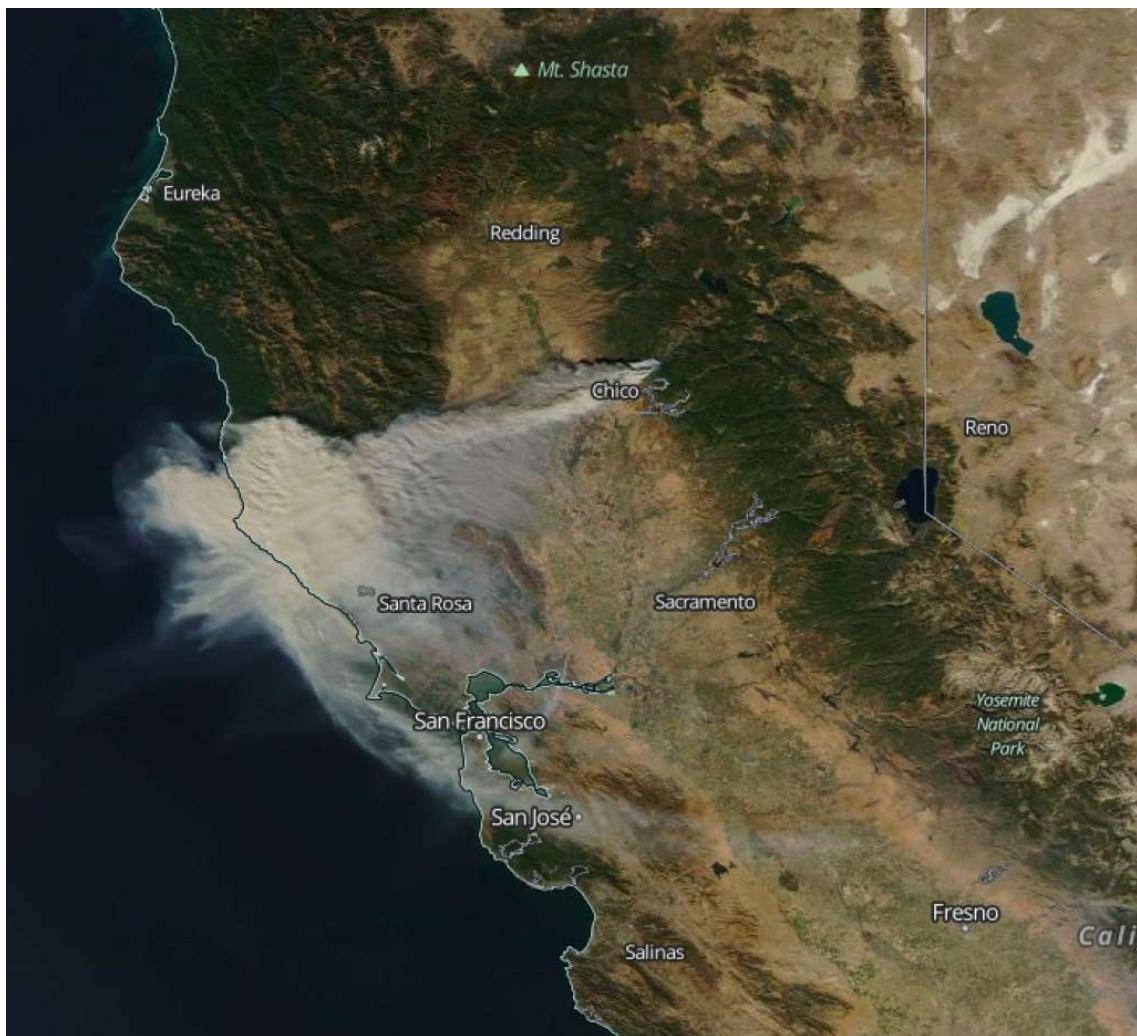


Figura 336 – Extensão da fumaça provocada pelo *Camp Fire*. Fonte: [SFGATE \(2018\)](#).

30.4.2.1 Definição do ponto de origem

Como discutido anteriormente, o fogo se espalhou rapidamente na região do Condado de Butte. Isso causou grandes danos a cabos de comunicação e torres de celular³, fazendo com que algumas localidades não tivessem avisos de evacuação. Porém, as cidades de Pulga e Paradise foram as únicas que receberam os avisos, tendo suas ordens de evacuação, via Twitter, às 07:23⁴ e 08:00⁵, respectivamente. Como a área afetada era extensa e o aumento de pessoas evacuando congestionaria as estradas e diminuiria o tráfego, os horários de evacuação foram diferentes. Portanto, houve a necessidade de fazer um cronograma de horas em que os residentes deveriam evacuar. Este cronograma foi então uma contribuição para este trabalho, já que em trabalhos anteriores os evacuados saiam no mesmo horário, como, por exemplo, Lämmel e Nagel (2008), Bakillah et al. (2012) e Benchekroun e Mouraouah (2018).

Assim, levando em consideração estes fatores, a área de evacuação foi dividida em quatro setores: o primeiro com Pulga e Starling City; o segundo com a região de Magalia e Concow, o terceiro com a área de

³ Ahead of Camp Fire Anniversary, New Details Emerge of Troubled Evacuation; Frontline. Disponível em <https://to.pbs.org/2XyxAR9>. Acesso em 20 de novembro de 2019.

⁴ Camp Fire Timeline of Terror: The evacuation of Butte County's Paradise from beginning to end; ABC7News. Disponível em: <https://abc7ne.ws/2OyK1rW>. Acesso em 20 de novembro de 2019.

⁵ As California's deadliest wildfire closed in, evacuation orders were slow to arrive; Los Angeles Times. Disponível em: <https://lat.ms/2s3BIUf>. Acesso em 20 de novembro de 2019.

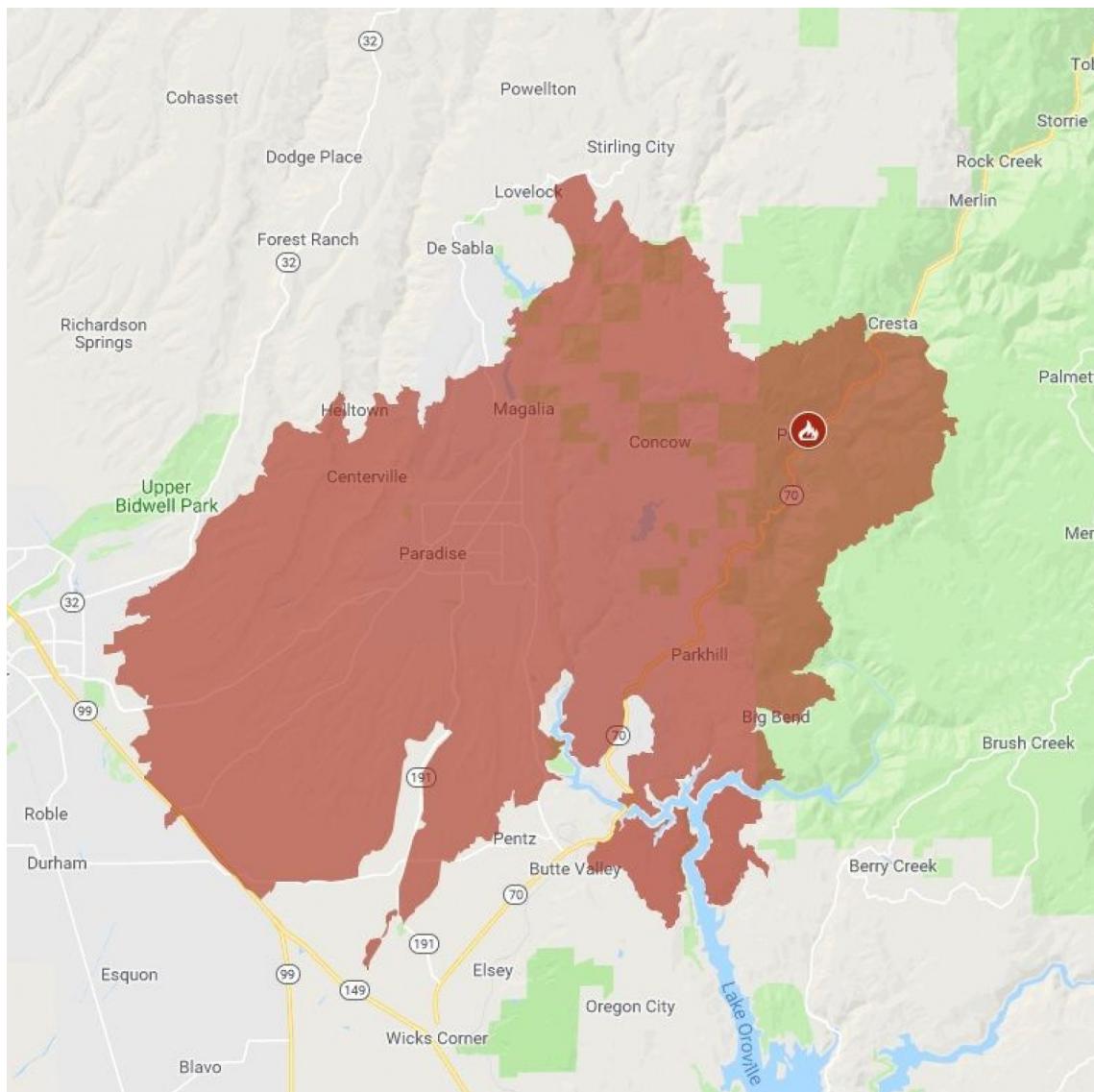


Figura 337 – Área queimada pelo incêndio *Camp Fire*. Escala: 2 km por cm. Fonte: [Newsweek \(2018\)](#).

Paradise, Centerville, Helltown e Parkhill e o quarto com as localidades próximas a Chico, como mostrado na Figura 339. Em relação a definição do horário de evacuação dessas áreas, foi considerado o tempo que o incêndio levou para chegar em Paradise após a ordem de evacuação de Pulga, ou seja, essas ordens de evacuação foram utilizadas como base para evacuar os outros dois setores. Logo, os horários de evacuação ficaram estabelecidos como 07:23, 07:42, 08:00 e 08:19, em ordem.

Após feito o cronograma, os pontos de origem dos evacuados foram definidos. Esta definição se deu por meio da obtenção de coordenadas aleatórias de cada setor e seus respectivos horários através da geração de um código. Estes pontos foram apresentados na Figura 340 a seguir.

30.4.2.2 Definição do ponto de destino

Inicialmente, para a definição do ponto de destino, nós realizamos um levantamento dos abrigos disponíveis próximos ao desastre e suas respectivas lotações. Assim, cinco abrigos foram selecionados⁶, como apresen-

⁶ **Camp Fire Resources**; Far Northern Regional Center. Disponível em: <http://bit.ly/2D2qRp9>. Acesso em 20 de novembro de 2019.

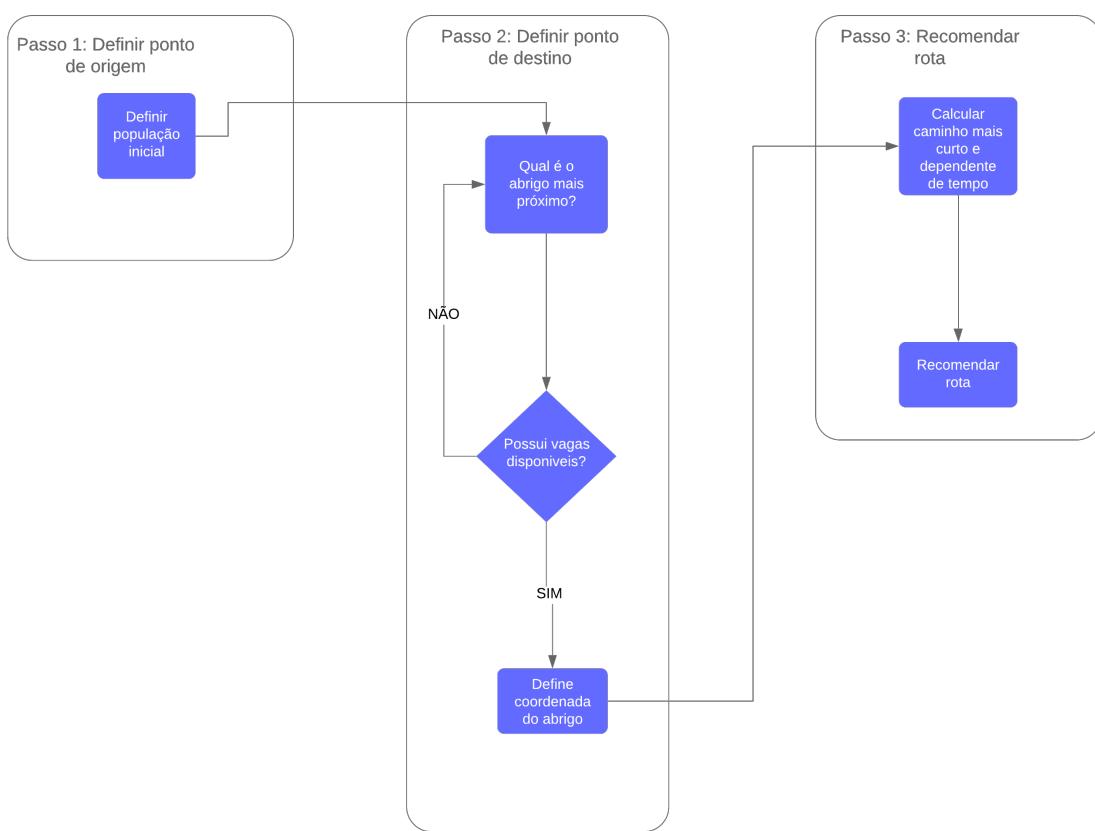


Figura 338 – Metodologia de recomendação de rotas.

tado na Tabela 210. Em seguida, suas coordenadas foram obtidas através do *Google Maps* e transformadas do Sistema Geodésico Mundial (WGS 84), norma usada pelo Sistema de Posicionamento Global (GPS), para o código criado pelo Grupo de Pesquisa Petrolífera Europeia (European Petroleum Survey Group), o código EPSG. Essa transformação se dá devido ao fato do simulador não suportar dados geográficos no sistema WGS 84.

Tabela 210 – Abrigos selecionados como destinos de evacuação.

Abrigos	Localização	Capacidade
East Avenue Church	Chico	250
Silver County Fairgrounds	Chico	620
Glenn County Fairgrounds	Orland	450
Yuba Sutter Fairgrounds	Yuba City	380
Butte County Fairgrounds	Gridley	100

Após isso, os nós mais próximos a essas coordenadas são escolhidos e assim é criada uma lista com os nós referentes as localizações dos abrigos, conforme é mostrado na Figura 341. Subsequentemente, é feito um cálculo utilizando o algoritmo de *Dijkstra* que recebe como parâmetros a lista de nós e o nó da primeira atividade dos agentes. Com isso, são retornados os menores caminhos entre o ponto de origem do evacuado e os abrigos. Assim, é possível criar uma lista com os abrigos mais próximos ao evacuado em ordem crescente. Depois, é checado se há vagas disponíveis no abrigo mais próximo. Se houver, este abrigo será escolhido como destino pro evacuado, se não, o abrigo seguinte na lista será checado. Essa é uma outra contribuição do nosso trabalho, já que normalmente é recomendado somente o abrigo mais próximo sem considerar a lotação, como visto nos trabalhos de Agafonova e Myasnikova (2017) e Zhu Kun Xie e Yang



Figura 339 – Setores do cronograma de evacuação. Adaptado de [Times \(2018\)](#).

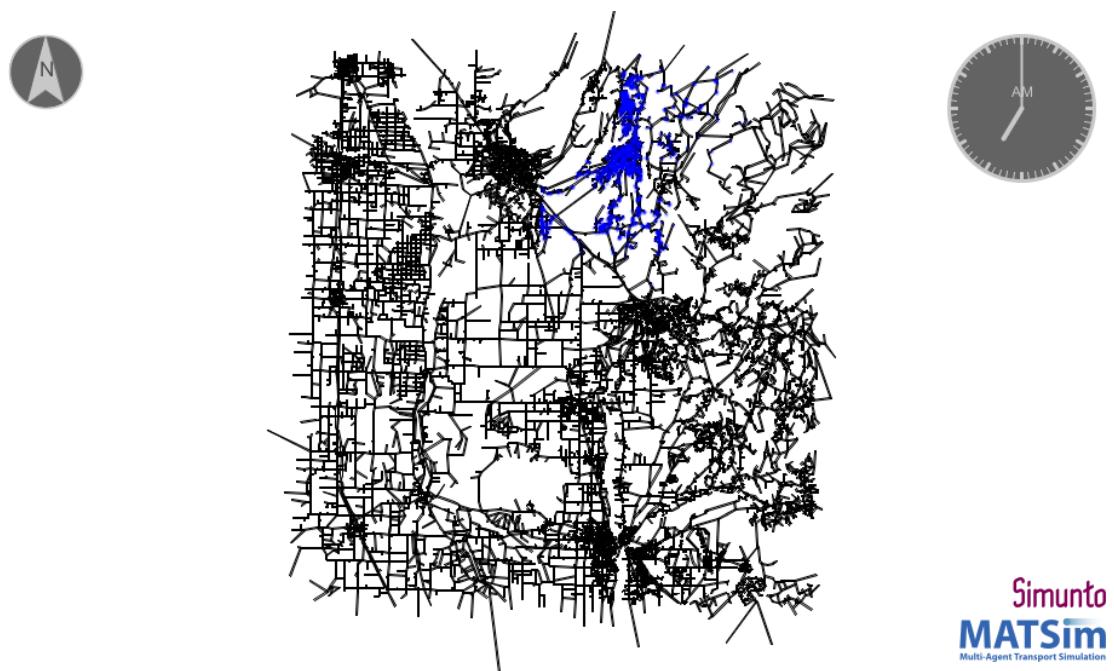


Figura 340 – Pontos de origem dos evacuados.

(2018), por exemplo.



Figura 341 – Localização de abrigos no mapa da área simulada.

30.4.2.3 Recomendação de rota

No MATSim, as rotas dos agentes são especificadas por sequências de *links* e são geradas por um algoritmo de caminho mais curto que requer uma função de custo. Esta função de custo pode ser personalizada. No cálculo da rota deste método, que acontece durante a simulação, o tempo de viagem do *link* foi utilizado como o custo e o algoritmo A* foi aplicado como roteador. Na construção dessa rota, foi considerado também o horário de partida da primeira atividade dos agentes e a disponibilidade de vias, já que no cenário real diversas rotas foram fechadas devido as chamas varrem perigosamente perto das rodovias. Para analisar essa disponibilidade, um algoritmo foi desenvolvido para gerar um arquivo de eventos de mudança de rede, fazendo com que fosse viável a remoção de vias indisponíveis em tempo de execução. Essa remoção foi feita de forma dependente do tempo, ou seja, foram definidos horários em que essas rotas deveriam ser removidas de qualquer sugestão. Com o cronograma de evacuação como base, foram estabelecidos horários diferentes para cada setor, cada um tendo uma hora para evacuar a área. Dessa forma, dependendo do horário, essas rotas não foram utilizadas na recomendação.

Além disso, como forma de recomendar melhores rotas, o equilíbrio de *Nash*, usado nativamente no simulador, foi aplicado. Ele calcula esse equilíbrio repetindo o comportamento de melhor resposta: em todas as iterações, uma fração dos agentes recalcula uma rota ou um destino com base no que seria melhor na iteração anterior, assumindo que o comportamento de todos os outros agentes permaneça inalterado. Após esse replanejamento, os planos resultantes de todos os agentes são executados simultaneamente na simulação de mobilidade e novas medidas de desempenho são computadas. Este processo é repetido várias vezes, de acordo com o número de iterações definidas no cenário, neste caso, 5 iterações. Uma vez estabilizado, nenhum agente pode melhorar substancialmente através de um replanejamento de rota ou destino e um equilíbrio aproximado de *Nash* é obtido.

Nesse módulo, outro fator que poderia influenciar na recomendação de rotas foi a minimização de congestionamentos na rede rodoviária. Esta minimização foi feita através da detecção de congestionamentos e, em seguida, a execução do algoritmo de roteamento novamente para recomendar rotas alternativas. Ou

seja, rotas congestionadas se tornaram menos atraentes, promovendo a busca por outras rotas. Com isso, decisões de viagem individuais dos agentes são alteradas durante as iterações. Esta implementação faz também com que a simulação passe do Equilíbrio de *Nash* para o Sistema Ideal, onde rotas mais rápidas são recomendadas, mas não são estáveis (rotas com maior risco de impactar negativamente no tempo de evacuação). Depois que o descongestionamento foi minimizado, os usuários de transporte não têm mais nenhum incentivo para se comportar de maneira ideal e retornam ao comportamento normal. Em consequência, o comportamento do deslocamento volta do Sistema Ideal para o equilíbrio de *Nash*.

30.4.3 Simulação

Após a descrição da metodologia, nesta seção, descrevemos como ocorre a criação dos arquivos necessários à simulação. Todos os códigos estão disponíveis em um repositório⁷ no *GitHub*.

30.4.3.1 Criação da rede de simulação

Como discutido anteriormente, um dos arquivos principais para a realização de uma simulação no MATSim é o arquivo de *network*. O método mais comum para a sua geração é através da exportação de dados do OpenStreetMap (OSM). Existem diversos serviços da Web que possibilitam a extração de dados para uma área de interesse, tais como: *GeoFabrik*, *BBBike* e *Overpass Turbo*. Além destes, existem outros softwares livres, como o *Quantum GIS* e *JOSM*, que têm *plugins* que são utilizados para este fim. Contudo, a forma mais comum é através do próprio site do *OpenStreetMap*, onde é possível buscar pelo mapa ou região de interesse e exportá-lo, conforme ilustrado na Figura 342. Com a exportação, um arquivo é gerado. Este arquivo é utilizado para salvar informações de mapas de ruas na forma de "nós"(pontos), "caminhos"(conexões) e "relações"(propriedades de ruas e objetos, como *tags*).

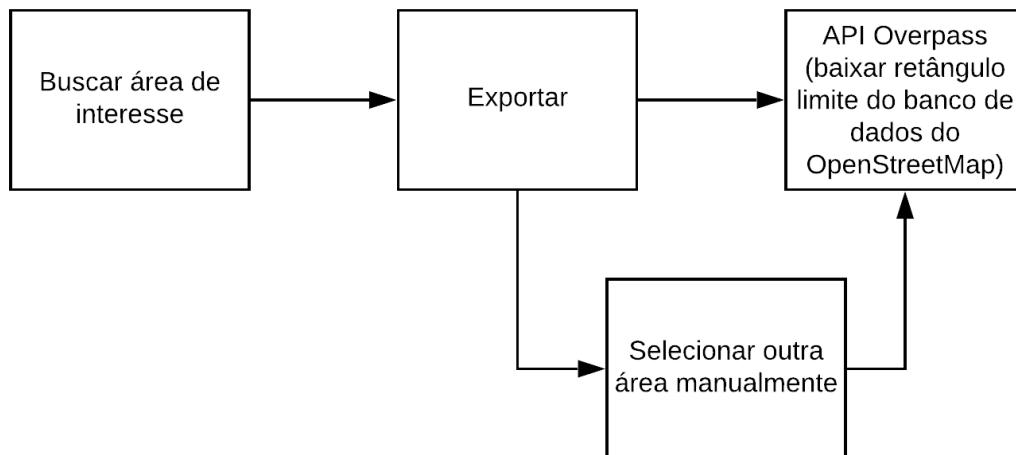


Figura 342 – Fluxo de extração de dados do *OpenStreetMap*.

Para a criação do cenário deste trabalho, foram exportados cinco arquivos: um arquivo com toda a rede rodoviária utilizada na simulação e outros quatro, correspondentes aos setores comentados anteriormente. Após isso, foi desenvolvido um código para a conversão de dados OSM em arquivos compatíveis com o simulador MATSim, resultando na rede mostrada na Figura 343.

⁷ Link do repositório. (<https://github.com/OrranaLhaynher/matsim-example-project>)



Figura 343 – Rede rodoviária MATSim após a conversão.

30.4.3.2 Criação da população MATSim

Para que o ciclo MATSim seja realizado é necessária uma demanda inicial, como mostrado na Figura 334. Existem comumente duas possibilidades para a geração dessa demanda: a forma prática e a exigente. A forma mais prática é usar um censo populacional completo na criação da demanda, enquanto que a exigente é gerar uma população sintética, que pode ser baseada em pesquisas de amostra ou estrutura.

Neste estudo de caso foi criada uma população sintética de 1800 agentes que foi dividida em 4 setores da área de evacuação, como comentado anteriormente. Então, cada agente teve uma atividade *home* (casa) formulada utilizando os dados de latitude e longitude definidos conforme as coordenadas do setor em que este foi designado, como mostrado na Figura 340. A atividade *shelter* (abrigos) foi definida a partir da seleção de qual abrigo seria enviado o evacuado.

30.4.3.3 Criação do arquivo de configuração

A criação do arquivo de configuração consistiu na alteração dos parâmetros de cada módulo do arquivo padrão gerado diretamente pelo simulador. Nesse estudo de caso, optou-se por modificar apenas 7 dos 25 módulos que compõem o arquivo de configuração padrão. No arquivo *config* utilizado como dado de entrada no software, só é preciso registrar os módulos que sofreram alteração. O MATSim assume os parâmetros padrão para os módulos que não são registrados no arquivo de configuração. Os tópicos a seguir detalham as configurações adotadas para a simulação deste estudo de caso.

1. Modulo Global - define o sistema de coordenadas.
2. Modulo Network - define o diretório do arquivo de rede.
3. Modulo Plans - define o diretório do arquivo de população.
4. Modulo Controller - o principal parâmetro definido nesse módulo é o número de iterações feitas na simulação, nesse caso adotado como 5. Outro parâmetro definido é a frequência com que os arquivos de saída são gerados (definido como 1 para este estudo) e o formato do *snapshot*.
5. Modulo Qsim - é definido o número máximo de horas do dia contidas em cada simulação. Isso é importante para evitar que o programa faça iterações infinitas caso haja algum erro de configuração.

6. Modulo *PlanCalcScore* - define os parâmetros da função de utilidade usada para calcular os scores de cada plano e atividade. Nesse estudo, foram mantidas as configurações gerais de pontuação do MATSim, e introduzidas apenas as características de cada tipo de atividade considerada.
7. Modulo *Strategy* - estabelece o que muda para os agentes entre cada iteração simulada. Um parâmetro importante nesse modulo é a quantidade de planos que pode haver na memória do agente.

30.5 Resultados

Neste seção, são apresentados os resultados da simulação feita no software MATSim utilizando os arquivos gerados pela metodologia descrita na seção anterior.

30.5.1 Visualização dos eventos de simulação

Após a importação do arquivo de configuração no *MATSim*, foram criados arquivos de *output*, possibilitando a visualização no *Via*. Três simulações foram planejadas para avaliar a metodologia de recomendação de rotas: (1) utilizando o cronograma de evacuação e as configurações nativas do simulador, (2) utilizando as modificações propostas sem o módulo de descongestionamentos e (3) com as modificações propostas. Além disso, arquivos de cada iteração foram gerados para um melhor entendimento do que acontece durante a simulação.

No primeiro cenário, foram simulados 1800 agentes, que tinham seus pontos de destinos escolhidos de forma randômica entre os cinco abrigos disponíveis. A área do desastre não foi removida durante a execução, fazendo com que rotas naquela área fosse recomendadas. Por isso, mesmo após o tempo seguro de utilização dessas rotas, agentes continuavam evacuando por elas. Notou-se também que os congestionamentos experienciados nessa simulação estavam concentrados ou na área de evacuação ou próximo a ela. Isso acontece devido a distribuição aleatória por abrigos.

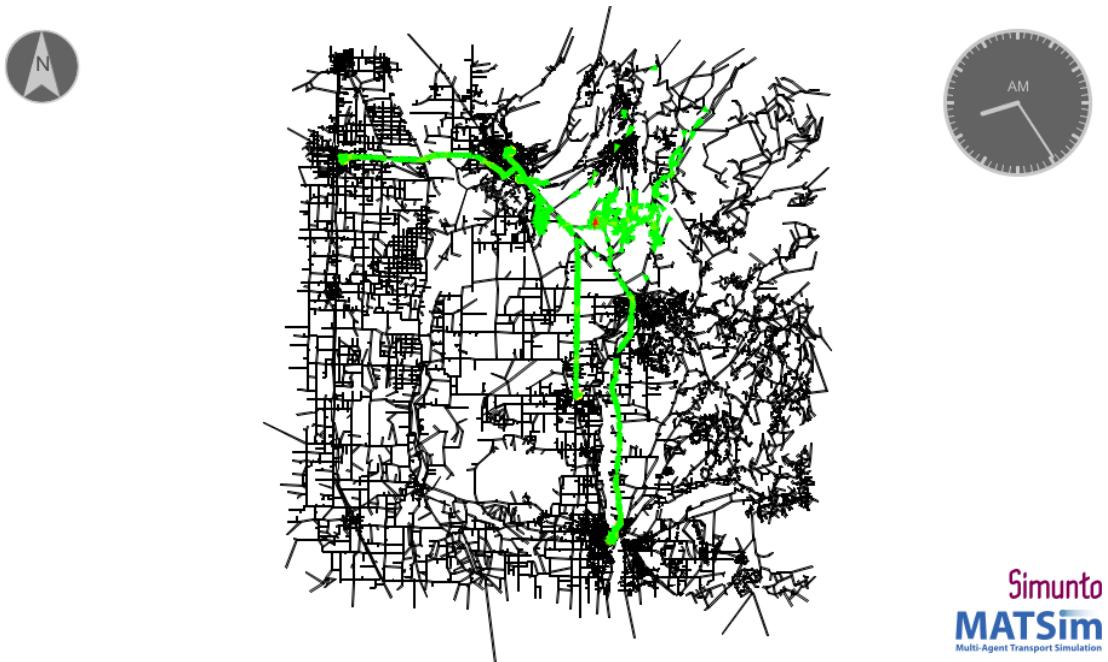


Figura 344 – Instante em que toda a população entra na rede rodoviária na simulação 01.

No segundo cenário, foram simulados 1800 agentes, que tinham como pontos de destinos recomendados o abrigo mais próximo e com vagas disponíveis. A área do desastre foi removida durante a execução, fazendo

com que rotas naquela área não fossem recomendadas após o horário definido. Desse modo, agentes não entram na região de perigo e acabam tendo rotas recomendadas mais seguras para evacuar. Com relação a congestionamentos, essa simulação foi a que teve o maior número deles. Porém, a maioria dos congestionamentos encontravam-se nas áreas em que os abrigos estavam localizados. Isso ocorre em virtude da rapidez com que os evacuados chegaram aos seus pontos de destino.

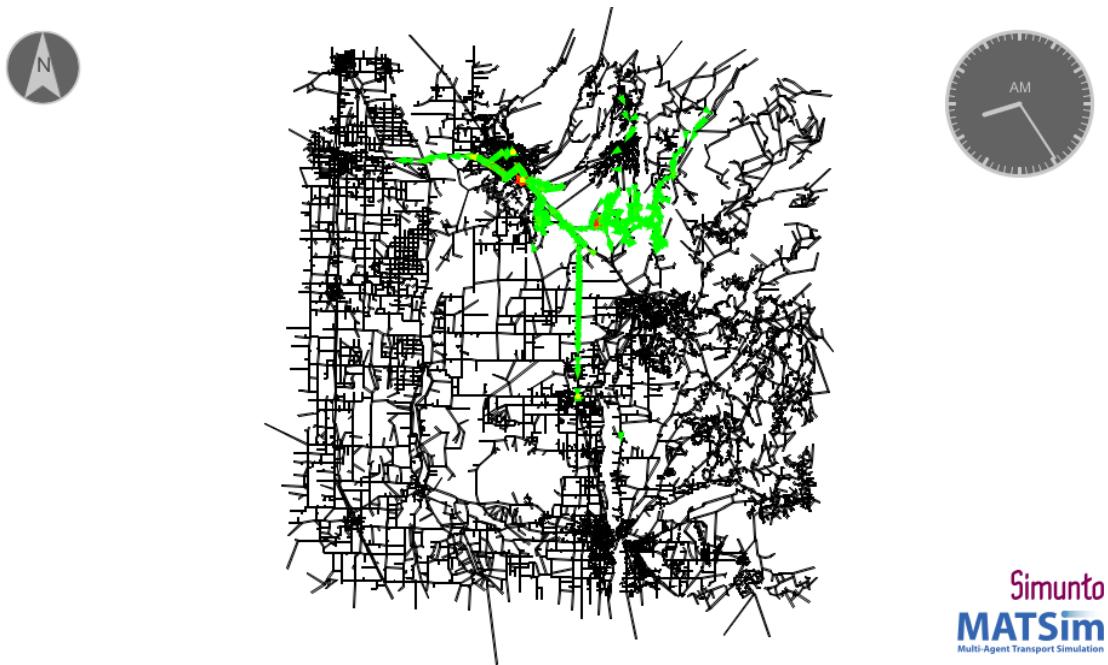


Figura 345 – Instante em que toda a população entra na rede rodoviária na simulação 02.

No terceiro cenário, os 1800 agentes, que tinham como pontos de destinos recomendados o abrigo mais próximo e com vagas disponíveis, possuiu outro fator no momento da recomendação da rota: a minimização de congestionamento. A área do desastre também foi removida durante esta execução, fazendo com que rotas nessa área não fossem recomendadas após o horário definido. No tocante dos congestionamentos, essa simulação foi a que teve o melhor desempenho. O número de congestionamentos foi menor que o da segunda simulação e maior do que o da primeira, como mostrado na Tabela 214. Porém, como dito anteriormente, os congestionamentos da primeira simulação estavam principalmente na área de desastre, enquanto que os congestionamentos desta simulação teve um número maior na área dos abrigos.

30.5.2 Análise

Uma análise das simulações foi feita para comprovar que as contribuições impactaram positivamente no tempo de evacuação individual e total do cenário. Na subseção 5.2.1 será mostrada como ficou a distribuição de evacuados nos abrigos. Enquanto que na subseção 5.2.2, foram apresentados uma amostra dos tempo de evacuação dos agentes em um dos abrigos escolhidos. Por fim, na subseção 5.2.3, o tempo de evacuação total foi exibido.

30.5.2.1 Distribuição de evacuados pelos abrigos

Na simulação com o cenário modelado de forma nativa, os evacuados tiveram o seu ponto de destino definido de forma aleatória, como mostrado na Figura 347. Mesmo com os horários de evacuação iguais, residentes eram distribuídos pelos cinco abrigos. Isso fez com que alguns tivessem um tempo de evacuação razoável, enquanto outros demoraram horas para chegar no seu destino. Na Figura 348, três evacuados

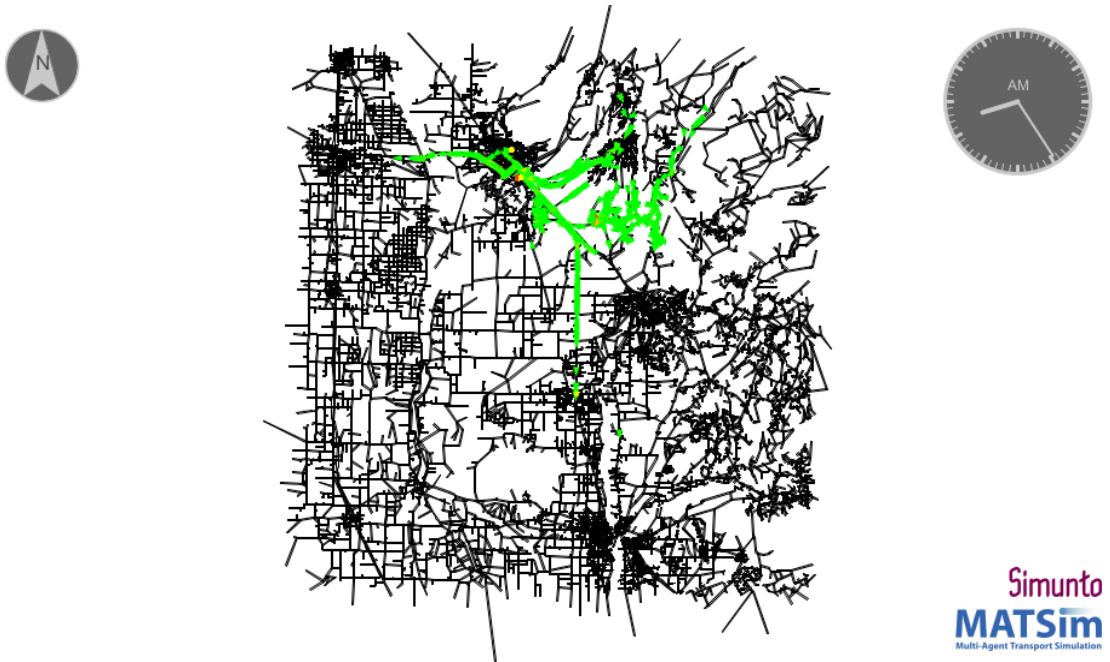


Figura 346 – Instante em que toda a população entra na rede rodoviária na simulação 03.

que saíram da mesma área estavam sendo encaminhados para três abrigos diferentes. Seus tempos de evacuação em minutos foram respectivamente 01:07:82, 57:25 e 45:40.

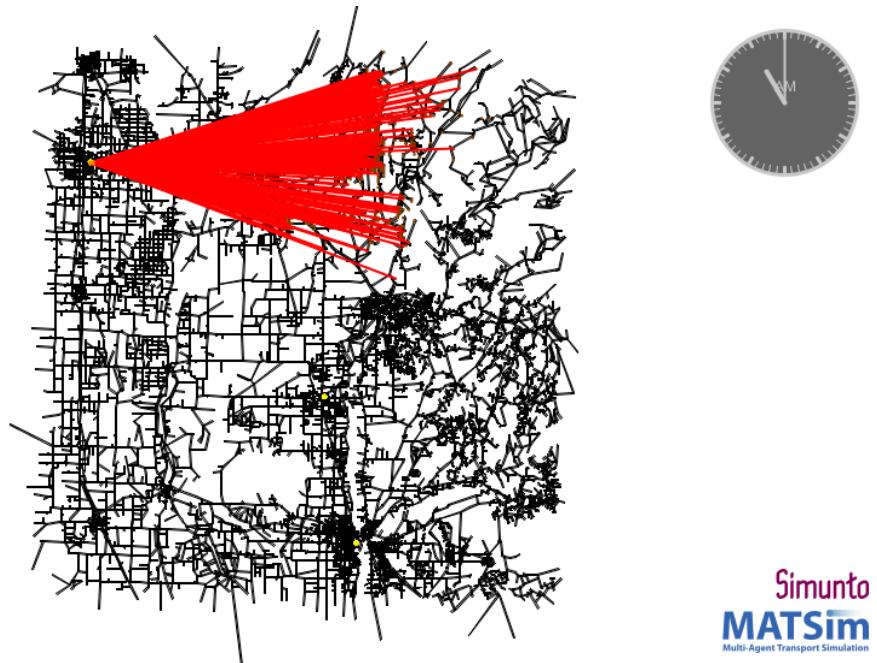


Figura 347 – Distribuição dos agentes no abrigo *Glenn County Fairgrounds* na simulação nativa do MATSim.

Em contrapartida, na simulação com o cenário modelado com as contribuições, os evacuados da mesma área foram encaminhados de seus pontos de origem para o abrigo mais próximo, como mostrado na Figura



Figura 348 – Três evacuados na mesma área sendo enviados para três abrigos diferentes.

349. Com isso o tempo de evacuação diminuiu, já que eles têm uma distância menor a percorrer. Na Figura 350, três evacuados que saíram da mesma área foram encaminhados para dois abrigos de acordo com a distância e número de vagas disponíveis. Seus tempos de evacuação em minutos foram respectivamente 31:07, 45:15 e 56:35.

30.5.2.2 Tempo de evacuação individual

Para fins de análise do tempo de evacuação dos agentes, foram exportados três arquivos CSV com os dados de todos os evacuados que tiveram como ponto de destino o abrigo Butte County Fairgrounds. Com esses arquivos, pode-se perceber uma diminuição no tempo de evacuação. Como amostra, os dez primeiros evacuados tiveram seus tempos de evacuação comparados.

Na Tabela 211, o maior tempo de evacuação foi de 59:55 minutos e o menor, de 22:03 minutos. Já na Tabela 212, o maior tempo de evacuação foi de 48:18 minutos e o menor, de 38:18 minutos. Por último, na Tabela 213, o maior tempo de evacuação foi de 01:04:35 hora e o menor, de 38:05 minutos. Com isso, pode se constatar que: (1) por definir aleatoriamente os pontos de destinos, a primeira metodologia faz com que aja uma discrepância entre os valores máximo e mínimo de evacuação; (2) a metodologia utilizada na segunda simulação consegue alcançar um equilíbrio no tempo de evacuação de todos os agentes por distribuir evacuados nos abrigos mais próximos a esses, já que as distâncias entre os abrigos e a área de desastre são parecidas; e (3) por fazer a transição do equilíbrio de *Nash* para o sistema ótimo, a metodologia utilizada na terceira simulação causa recomendações de rotas instáveis (rápidas, mas que podem não ser as melhores), provocando assim um aumento no tempo de evacuação.

30.5.2.3 Tempo de evacuação total

Na Tabela 214, o tempo de execução, o número de congestionamentos na rede, a duração destes congestionamentos e o tempo de evacuação total de cada simulação foi apresentado. Como pode-se observar, as simulações 02 e 03 tiveram tempos de evacuação melhores do que a simulação 01. Isso se dá principalmente pelo planejamento adequado do método de evacuação. Além disso, na simulação 01, com a recomendação aleatória de residentes pelos abrigos, agentes demoraram para chegar nos destinos, o que afeta o tempo de



Figura 349 – Distribuição dos agentes no abrigo *Glenn County Fairgrounds* na simulação da metodologia do trabalho.



Figura 350 – Três evacuados sendo enviados para os dois abrigos mais próximos.

Tabela 211 – Dados de dez evacuados da simulação nativa.

Id do agente	Horário de saída	Horário de chegada	Duração em horas
973	08:00:00	08:55:02	00:55:02
924	08:00:00	08:59:15	00:59:15
881	07:27:00	08:09:55	00:42:55
972	08:00:00	08:57:12	00:57:12
959	08:00:00	08:30:03	00:30:03
923	08:00:00	08:41:17	00:41:17
880	07:27:00	07:49:03	00:22:03
971	08:00:00	08:59:55	00:59:55
975	08:00:00	08:58:13	00:58:13
922	08:00:00	08:39:17	00:39:17

Tabela 212 – Dados de dez evacuados da segunda simulação.

Id do agente	Horário de saída	Horário de chegada	Duração em horas
973	08:00:00	08:38:28	00:38:28
924	08:00:00	08:45:03	00:45:03
881	08:00:00	08:40:05	00:40:05
972	08:00:00	08:39:27	00:39:27
959	08:00:00	08:39:10	00:39:10
923	08:00:00	08:46:40	00:46:40
880	08:00:00	08:48:18	00:48:18
971	08:00:00	08:38:18	00:38:18
975	08:00:00	08:43:12	00:43:12
922	08:00:00	08:39:37	00:39:37

Tabela 213 – Dados de dez evacuados da simulação da metodologia completa.

Id do agente	Horário de saída	Horário de chegada	Duração em horas
973	08:00:00	08:38:17	00:38:17
924	08:00:00	09:00:23	01:00:23
881	07:27:00	08:06:57	00:39:57
972	08:00:00	08:39:18	00:39:18
959	08:00:00	09:04:35	01:04:35
923	08:00:00	08:43:12	00:43:12
880	07:27:00	08:15:37	00:48:37
971	08:00:00	08:38:05	00:38:05
975	08:00:00	08:41:53	00:41:53
922	08:00:00	08:38:58	00:38:58

evacuação do cenário como um todo. Através da Tabela 214, nota-se também que o número de congestionamentos nas simulações 02 e 03 é maior do que na simulação 01. Esse aumento de congestionamentos

acontece devido a diminuição de rotas disponíveis na área de desastre durante essas simulações, já que com o tempo essas rotas vão sendo removidas da recomendação. Por isso, há um número menor de rotas e consequentemente um número maior de congestionamentos. Contudo, a maioria dos congestionamentos nessas simulações ocorre fora da área de risco, principalmente próximo aos abrigos. Esta particularidade dá-se devido a rapidez com que evacuados chegam aos seus destinos. Enquanto que na primeira simulação os congestionamentos concentram-se na área de desastre.

Tabela 214 – Tempo de evacuação total e congestionamentos.

Simulações	Tempo de Execução	Quantidade de Congestionamentos	Duração dos Congestionamentos em minutos	Tempo de Evacuação Total
Simulação 01	00:00:58	4	21:16, 30:06, 24:53, 07:45	02:50:57
Simulação 02	00:01:01	9	24:09, 30:53, 13:08, 31:14, 54:41, 17:02, 17:24, 39:50, 21:19	02:28:46
Simulação 03	00:02:52	5	15:29, 15:53, 32:09, 41:51 e 13:57	02:42:19

30.6 Conclusão

O planejamento e simulação de desastres através de métodos de evacuação traz benefícios a pesquisadores e desenvolvedores na área, pois permite a avaliação de políticas de evacuação antes de serem colocadas em prática em um ambiente real, facilitando assim a criação de estratégias eficazes. Neste trabalho, foi aplicada uma metodologia de recomendação de rotas e, consequentemente, a simulação desta no cenário do desastre natural *Camp Fire*. O objetivo foi recomendar rotas utilizando distribuição de evacuados em abrigos próximos com disponibilidade de vagas, remoção de vias indisponíveis e a minimização de congestionamentos.

Como forma de analisar o desempenho das contribuições feitas, foram utilizados os tempos de evacuação das três simulações realizadas. Com isso, resultados satisfatórios quanto aos gargalos previstos e os tempo de evacuação, tanto individual quanto global, foram obtidos. Assim, informações úteis quanto ao desempenho da metodologia foram obtidas através da comparação com uma metodologia simples de evacuação. Como resultado principal, o tempo de evacuação total diminuiu em 08:38 minutos com a adoção do método completo e 22:11 minutos com a adoção do método sem o módulo de congestionamento, possibilitando assim o uso deste na criação de políticas de evacuação.

Com o desenvolvimento de trabalhos futuros, essa metodologia será aplicada a simulação de um cenário com números diferentes de evacuados e vagas em abrigos. Além disso, a melhoria do modelo de cronograma de evacuação deve ser realizado, com isso a análise do impacto de diferentes cronogramas no tempo de evacuação e no número de congestionamentos da simulação poderá ser avaliado.

Parte III

Processamento de sinais

Extração Manual De Características Para Classificação De Sons Urbanos

Autores: Myllena Caetano de Oliveira; Deborah Maria Vieira Magalhães.

Resumo: O som é uma importante fonte de informações. Em relação aos sons urbanos, a perspectiva de conectividade, captura, análise e classificação automáticas dos sons podem auxiliar na redução da poluição sonora nas cidades, computação sensível ao contexto e vigilância urbana. Para que essas informações possam ser obtidas é necessário que o áudio passe por algumas etapas, dentre elas, a extração de características. Este trabalho propõe o desenvolvimento de um descritor, através da extração manual e seleção de características, capaz de realizar a classificação de sons urbanos. As características foram extraídas no domínio do tempo, frequência e cepstral, a seleção das mesmas foi feita com base na lista ranqueada gerada pelo *Information Gain*. O resultado obtido foi um descritor compacto e preciso capaz de caracterizar sons urbanos com 94,2% de acurácia para o conjunto UrbanSound8K e 78,7% para o conjunto ESC (urban noises). Em comparação aos estudos relacionados, o método proposto apresentou resultados promissores, apresentando um desempenho superior até mesmo que descritores baseados em abordagem de aprendizado profundo. Os resultados alcançados enfatizam o desempenho das características manuais para a representação dos sinais acústicos. Dessa forma, o método proposto mostra-se eficiente para integrar aplicações de classificação de eventos sonoros em espaços urbanos.

Palavras-chaves: Processamento de Áudio. Seleção de Características. Eventos Sonoros.

31.1 Introdução

Em 2018, cerca de 55,3% da população mundial vivia em espaços urbanos, esse número chegou a 80% quando tratamos da Europa e América do Norte. A China apresentou um crescimento de 40% em sua parcela de habitantes urbanos nos últimos 50 anos¹. Até 2030, as áreas urbanas devem abrigar 60% das pessoas em todo o mundo e uma em cada três pessoas viverá em cidades com pelo menos meio milhão de habitantes. O aumento da densidade populacional estressa a infraestrutura dos centros urbanos e traz consigo diversos desafios relacionados à mobilidade urbana, segurança e saúde pública (SOUZA et al., 2018).

¹ <https://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS>

Neste âmbito, as cidades inteligentes despontam como uma oportunidade para guiar políticas públicas visando oferecer melhores serviços e infraestrutura aos cidadãos (RATHORE et al., 2016). As cidades inteligentes se apoiam na expansão da conectividade com a Internet das Coisas (*IoT*, do inglês *Internet of Things*) para o monitoramento ubíquo e inteligente através de redes de sensores e *smartphones*. Sendo o som uma importante fonte de informação a respeito da vida urbana (EDWARDS, 2018), a perspectiva de conectividade, captura, análise e classificação automáticas dos sons urbanos pode auxiliar na redução da poluição sonora nas cidades e na tarefa de vigilância urbana.

Alguns fatores motivadores do desenvolvimento de novas pesquisas focadas na identificação de eventos sonoros a partir do processamento de *streams* de áudio são:

- O aumento da densidade populacional em centros urbanos projeta, entre outros desafios, a dificuldade de realização de policiamento efetivo e proteção de espaços públicos. Nos últimos anos, enfrenta-se no mundo o crescimento da criminalidade e terrorismo (BELLO; MYDLARZ; SALAMON, 2018). No Brasil, em 2012, as grandes cidades apresentaram altos níveis de criminalidade nas categorias de roubo, estupro, fraude e roubos residenciais (BORGES et al., 2017);
- O crescimento das redes de sensores acústicos aliado à ampla adoção dos *smartphones* impulsionou a utilização do microfone como dispositivo vantajoso para monitorar espaços urbanos, pois este é menor e mais barato que a câmera. Além disso, o microfone é mais robusto às condições ambientais, como neblina, poluição, chuva e mudanças de luminosidade, e realiza o monitoramento omnidirecional, sendo menos suscetível à oclusão (BELLO; MYDLARZ; SALAMON, 2018). No geral, a captura de som exige menos bateria dos dispositivos envolvidos (VIRTANEN; PLUMBLEY; ELLIS, 2018b) quando comparados a aplicações de monitoramento por vídeo.

31.1.1 Problema

O processamento e análise de fontes acústicas podem contribuir para localização e rastreamento de atos criminosos, terroristas e tumultos e, a partir dessa identificação, definir características como nível, duração e frequência de tais eventos. Isso pode, por sua vez, oferecer maior compreensão nas áreas de ciências sociais e políticas públicas sobre a relação entre o som urbano e a criminalidade em uma determinada região, melhorando a efetividade das intervenções tanto da polícia como dos órgãos que prestam socorro às vítimas.

Para a maioria dos humanos e animais, a habilidade de escutar eventos sonoros é uma tarefa trivial, mas desenvolver algoritmos que sejam capazes de automaticamente reconhecer um som é desafiador (VIRTANEN; PLUMBLEY; ELLIS, 2018b). Quando trata-se de ambientes urbanos, onde são encontrados sons naturais não biológicos (vento, chuva), sons naturais biológicos (animais em geral) e sons mecânicos (tráfego, construção, sinais, máquinas, instrumentos musicais), tem-se um ambiente altamente heterogêneo que pode possuir ilimitados sons.

Ainda, em ambientes realistas, o evento sonoro que se deseja classificar se sobrepõe aos demais presentes no ambiente. Portanto, o áudio capturado é uma superposição de todas as fontes presentes. Além disso, em várias aplicações de monitoramento de eventos sonoros, os microfones usados para capturar áudio costumam ficar significativamente mais distantes das fontes sonoras, o que aumenta a interferência no sinal capturado. Todos esses fatores previnem a correspondência do sinal com os modelos desenvolvidos para reconhecê-lo. Essa situação é bem diferente das aplicações de fala, entre elas, reconhecer a sequência de palavras na fala ou reconhecer a identidade da pessoa que está falando, onde comumente os microfones são utilizados em ambientes fechados e controlados. Portanto, a análise e compreensão de eventos acústicos em espaços urbanos é ainda mais desafiadora.

A realização de monitoramento sonoro exige custo financeiro e disponibilidade de tempo para instalação, configuração e monitoramento do som. Ao se adquirir uma base de dados pública, tais desafios são contornados, surgindo, no entanto, novos desafios. Sendo estes a impossibilidade de controlar a qualidade do sinal, número de amostras insuficientes para em algumas classes, dados provenientes de diferentes sensores e muito heterogêneos.

31.1.2 Objetivos

Diante do exposto, o objetivo geral deste trabalho é a construção de uma metodologia para a extração de características que ofereça uma boa descrição de eventos sonoros, contribuindo para detecção de situações de risco à segurança dos habitantes, melhoria da saúde e gerenciamento do tráfego em espaços urbanos. Portanto, os objetivos específicos são:

1. Investigar a viabilidade da utilização de características manuais para realizar a extração de atributos dos sinais acústicos nos domínios de tempo, frequência e cepstrais;
2. Aplicar uma etapa de seleção de características para utilizar na posterior etapa de classificação apenas as características mais representativas, reduzindo a dimensionalidade do descritor sem perder informações importantes;
3. Desenvolver um descritor preciso e compacto que possa ser empregado em soluções embarcadas, requisito relevante no contexto de cidades inteligentes; e
4. Aplicar algoritmos de classificação clássicos como, *Random Forest* e *Support Vector Machine*, a fim de avaliar o descritor proposto.

31.1.3 Organização do Trabalho

Este capítulo está organizado através das seguintes seções: a Seção 34.2 fundamenta os conceitos envolvidos no desenvolvimento do trabalho; a Seção 31.3 detalha as etapas realizadas na execução do método proposto; a Seção 31.4 apresenta e discute os resultados alcançados em todos os testes realizados com a abordagem proposta; a Seção 32.6 apresenta a conclusão deste trabalho, bem como os trabalhos futuros.

31.2 Referencial Teórico

Com o intuito de fornecer um melhor entendimento sobre os temas abordados neste trabalho, este capítulo apresenta uma breve revisão literária. Os conceitos explanados referem-se ao som (Seção 31.2.1), às técnicas de pré-processamento (Seção 31.2.2), às etapas de extração (Seção 31.2.3) e seleção (Seção 32.2.4) de características sonoras, à classificação das características (Seção 31.2.5) e às métricas de avaliação dos modelos de classificação (Seção 31.2.6).

31.2.1 Som

O som é o resultado de vibrações no ar ou na água, que se propaga através ondas chamadas de ondas sonoras (VIRTANEN; PLUMBLEY; ELLIS, 2018b). É possível concluir, então, que o som é uma onda do tipo mecânica, já que precisa de um meio para propagar-se, e que a mesma é tridimensional, já que pode ser percebida de qualquer direção. Existem 3 características principais que definem um som: intensidade, altura e timbre (OLIVEIRA, 2002). Na Figura 351, tem-se uma representação de onda sonora.

A intensidade do som está relacionada à energia de vibração da fonte que emite o som, ela é provocada pela pressão que a onda exerce sobre o ouvido, ou seja, quanto maior a pressão maior será a intensidade (IAZZETTA, 2010). A unidade de medida utilizada para representar essa propriedade é o bel (decibel). A altura é o que diferencia um som entre grave e agudo, sendo que o som grave é o que identificamos como mais “grosso” enquanto o agudo é o mais “fino”. Como demonstrado na Figura 351, essa característica é definida pela frequência da onda sonora. Um som com baixa frequência é dito som grave e o som com altas frequências é dito som agudo (OLIVEIRA, 2002). Finalmente, o timbre é o que nos permite diferenciar os sons de mesma frequência e mesma intensidade. Isso ocorre porque o timbre faz com que as ondas sonoras produzidas por esses sons tenham um formato completamente distinto (OLIVEIRA, 2002). Por exemplo, se dois instrumentos diferentes, como um piano e um saxofone, estiverem tocando uma mesma nota musical, a distinção desses instrumentos será possível graças ao timbre.

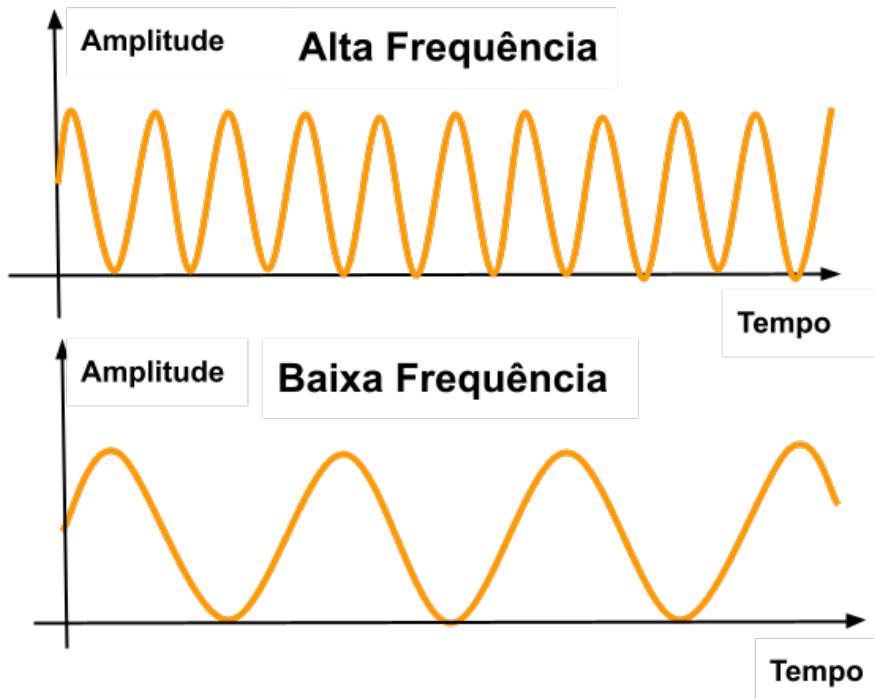


Figura 351 – Representação da onda sonora.

Uma onda sonora pode ser representada em um gráfico bidimensional, onde o eixo horizontal representa a passagem do tempo e o vertical a variação de pressão. Esse tipo de gráfico pode fornecer várias informações sobre o som ([IAZZETTA, 2010](#)), e é baseado nisso que as características sonoras são extraídas.

31.2.2 Pré-processamento

Para que a extração de características possa ser realizada é necessário que um áudio passe por alguns processos. O principal papel da etapa de pré-processamento é aprimorar determinadas características do sinal recebido, a fim de maximizar o desempenho da análise de áudio nas fases posteriores do sistema de análise. O pré-processamento também busca garantir a uniformização dos dados adquiridos, de modo que todos possuam as mesmas configurações de taxa de amostragem, quantização e números de canais.

A taxa de amostragem consiste no número de amostras por unidade de tempo medido em hertz (Hz), sendo uma amostra uma medida de amplitude do sinal em um intervalo de tempo. A taxa de amostragem influencia diretamente na semelhança do sinal digital com o analógico ([KOIZUMI et al., 2019](#)), dessa maneira, quanto maior for a taxa de amostragem, mais precisa é a representação do sinal. A quantização trata de discretizar um áudio em valores inteiros dentro de um intervalo de representações, sendo os comprimentos mais usados de 16-bit, 24-bit e 32-bit. A quantização interfere na qualidade do som, quanto maior o comprimento, maior a qualidade, e por consequência, maior o tamanho do arquivo ([PICHLMAIR; KAYALI, 2007](#)). O número de canais de áudio se refere a quantidade de canais que são responsáveis pela captura do som. Em um sistema monofônico, por exemplo, todas as informações do áudio são registradas em um mesmo canal ([EPPOLITO, 2008](#)).

Os valores típicos para a qualidade de um CD de áudio são uma taxa de amostragem de $f/s = 44.1$ Hz e uma quantização em 16 bits por amostra levando a uma taxa de bits de 705.600 bit/s para um sinal de áudio de canal único. Padrões de qualidade mais altos incluem taxas de amostragem de 48, 96 ou 192 Hz e quantização em 24 bits ([SERIZEL et al., 2018](#)).

31.2.3 Extração de Características

A fase de extração de características busca extrair de um som características que melhor o representam e que, através dessas, seja possível a um classificador distinguir esse som de acordo com a classe a qual ele pertence. Neste trabalho, foram extraídas características nos domínios de tempo, frequência e cepstral.

31.2.3.1 Características no domínio do tempo

As características de tempo estão diretamente ligadas com a onda temporal formada pelo áudio (SERIZEL et al., 2018). No domínio do tempo, foram extraídas as características: *Tempogram*, *Zero Crossing Rate* (ZCR) (SHARMA; UMAPATHY; KRISHNAN, 2020), *Spectral Rolloff* (XIE; ZHU, 2019), *Spectral Centroid* e *Root Mean Square* (RMS), sendo as características *Spectral Centroid* e RMS baseadas em energia (BARKER; VIRTANEN, 2016).

Entre elas, destaca-se o *Tempogram*, uma representação de tempo que está relacionada ao ritmo, matematicamente representado pela equação 31.1, onde t representa o tempo de duração em segundos e l o tempo de atraso (*lag*) (GROSCHÉ; MÜLLER; KURTH, 2010). A característica ZCR (Equação 31.2), assim como o nome sugere, irá contar a quantidade de vezes em que o sinal passa pelo ponto zero, ou seja, quando o valor do sinal muda de positivo para negativo ou de negativo para positivo. O *Spectral Rolloff* é definido como a frequência sob a qual uma porcentagem predefinida (normalmente entre 85% e 99%) da energia espectral total está presente.

$$A(t, l) = \frac{\sum_{n \in \mathbb{Z}} \Delta(n)\Delta(n+l).W(n-t)}{2N+1-l} \quad (31.1)$$

$$ZCR = \frac{1}{2} \sum_{N=1}^n |sign(x[n]) - sign(x[n-1])| \quad (31.2)$$

Para se conseguir o *Spectral Centroid*, cada quadro de um spectrograma de magnitude é normalizado e tratado como uma distribuição sobre *bins* de frequência, da qual a média (centroid) é extraída por quadro (Equação 31.4) (GIANNAKOPOULOS; SPYROU; PERANTONIS, 2019). O RMS (*root-mean-square*) é calculado a partir do sinal de som bruto no domínio do tempo. Matematicamente, o RMS é representado pela equação 31.3 (LIU; TSENG; TRAN, 2019).

$$RMS = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (31.3)$$

$$centroid(t) = \frac{\sum_k S(k, t).freq(k)}{\sum_j S(j, t)} \quad (31.4)$$

31.2.3.2 Características no domínio da frequência

No domínio da frequência, foram empregadas características relacionadas ao cromagrama, sendo elas: *Chroma STFT* (*Short-Time Fourier Transform* - transformada de Fourier de curto tempo) (ELLIS, 2007), *Chroma CQT* (*Constant-Q Transform* - transformada de constante-Q) (SCHÖRKHUBER; Klapuri, 2010) e *Chroma CENS* (*Chroma Energy Normalized* - Energia de croma normalizada) (MÜLLER; EWERT, 2011). Também no domínio da frequência foram utilizadas as características espetrais: *Entropy* (AL-NASHERI et al., 2017), *Spectral bandwidth* (WU; VINTON, 2017), *Spectral Flatness* (DUBNOV, 2004) e *Spectral contrast* (JIANG et al., 2002). Por fim, foram extraídos os coeficientes polinomiais de enésima ordem (*Poly*) (YANG et al., 2019) e *Tonnetz* (HARTE; SANDLER; GASSER, 2006), que está relacionado à tonalidade.

Chroma STFT é um atributo poderoso de propósito geral para o processamento de sinal de áudio. Ele define uma classe particularmente útil de distribuições de tempo-frequência que especifica a amplitude complexa versus tempo e frequência para qualquer sinal. A *Fourier transform* é uma característica bem

conhecida para analisar a distribuição de frequência de um sinal (ELLIS, 2007). Então o STFT discreto (D) sobre uma janela g com suporte compacto função pode ser escrita como:

$$STFT = F_{gf} [n, k] = \sum_{m=0}^{m-1} f[n-m] g[m] \epsilon_k[m], \quad (31.5)$$

onde,

$$\epsilon_k = e^{-2\pi m \frac{k}{N}}, \quad (31.6)$$

m é o comprimento da janela de g e n é o número de amostras em f . Este algoritmo pode ser interpretado como uma sucessiva avaliação das transformadas de Fourier sobre segmentos curtos de todo o sinal. Além disso, as frequências podem ser visualmente representadas exibindo a magnitude quadrada de Fourier coeficientes em cada seção. Este diagrama é chamado de spectrograma de f .

O CQT, segundo Brown (1991), é uma representação tempo-frequência que, diferente da STFT, apresenta um espectro de frequências com fator de seleitividade (também conhecido como fator Q) constante. Devido a isto, as componentes de frequência estão espaçadas geometricamente, ou seja, uma componente f_k é dada pela Equação 31.7.

$$CQT = f_{k-1} \left(\frac{1}{Q} + 1 \right) = f_{min} \left(\frac{1}{Q} + 1 \right)^k \quad (31.7)$$

Adicionando um outro grau de abstração considerando o curto espaço de tempo estatísticas sobre distribuições de energia dentro as bandas de croma, obtém-se CENS (Estatísticas normalizadas de energia cromada), que constitue uma família de características de áudio escaláveis e robustos. Estas características acabaram sendo muito úteis na correspondência e recuperação de áudio dos formulários. Na extração do CENS, uma quantização é aplicada com base em limites escolhidos logaritmicamente (MULLER; KURTH; CLAUSEN, 2005).

Inicialmente, uma quantidade que descreve o grau de desordem em um sistema termodinâmico, a entropia é posteriormente amplamente utilizada para avaliar a incerteza de um sistema. Da perspectiva da teoria da informação, entropia é a quantidade de informação contida em uma distribuição de probabilidade generalizada. Enquanto o paramétrico não linear que quantifica a complexidade de uma série temporal, pode ser usado para descrever sinais dinâmicos não lineares e instáveis (Equação 31.8) (AL-NASHERI et al., 2017).

$$Entropy = - \sum_{i=0}^{N-1} P_i * \log_2 (P_i) \quad (31.8)$$

Spectral bandwidth de banda e frequência dominante de uma potência de Fourier espectro para fornecer a base para a introdução de medidas espetrais instantâneas correspondentes. Essas quantidades são então reduzidas a atributos de rastreamento complexos facilmente computáveis. O objetivo de apresentar esses atributos nesse maneira é enfatizar seu papel como medidas variantes no tempo de propriedades espetrais médias. Isso os empresta intuitivamente significado atraente e sugere aplicações úteis (WU; VINTON, 2017).

Como conhecido da teoria de codificação, o ganho máximo que pode ser recuperado por redução de redundância usando métodos de codificação preditivos ou codificação de transformação é determinado pela desplanicidade da densidade espectral de potência do sinal e está relacionado com o chamado *Spectral Flatness* (DUBNOV, 2004).

A característica *Spectral contrast* pode refletir aproximadamente a distribuição relativa do componentes harmônicos e não harmônicos no espetro. Características anteriores, como MFCC, calculam a média do espetro distribuição em cada sub-banda, e assim perder o espetro espectral relativo em formação (JIANG et al., 2002).

O Tonnetz é uma representação plana dos sons através de um conjunto de linhas paralelas criadas a partir da circunferência do "círculo das quintas". Ele busca determinar a distância entre a tonalidade de cada som (Equação 31.9) (HARTE; SANDLER; GASSER, 2006).

$$t = \sqrt{\frac{1}{4} + d^2} > 1 \quad (31.9)$$

31.2.3.3 Características cepstrais

As características cepstrais tem relação com a forma com que a audição humana percebe os sons, principalmente a fala. As características cepstrais mais comuns são os coeficientes cepstrais de frequência mel (MFCCs, do inglês *Mel Frequency Cepstral Coefficients*), (VIRTANEN; PLUMBLEY; ELLIS, 2018b). Os MFCCs na realidade são um conjunto de características, geralmente de 10 a 20, que descrevem de forma concisa a forma geral de um envelope espectral, ou seja, o limite em que o espectro do sinal está contido. Matematicamente, os MFCCs são resultado da transformação inversa da discreta do cosseno da energia do *log* em bandas de frequência de mel, representado pela Equação 31.10 (SERIZEL et al., 2018).

$$MFCC(t, c) = \sqrt{\frac{2}{M_{mfcc}}} \sum_{m=1}^{M_{mfcc}} \log(|X|_m(t)) \cos\left(\frac{c(m - \frac{1}{2})}{M_{mfcc}}\right), \quad (31.10)$$

onde M_{mfcc} é o número de bandas de frequência mel, m o índice de banda de frequência, $|X|_m(t)$ é a energia na M nesima banda de frequência mel e c é o índice de do coeficiente cepstral.

Além disso, através do cálculo da primeira e segunda derivada dos MFCCs, foram obtidas as características também cepstrais, *delta1* (Equação 31.11) e *delta2* (Equação 31.12).

$$\text{Delta1} = f'(MFCC) \quad (31.11)$$

$$\text{Delta2} = f''(MFCC) \quad (31.12)$$

31.2.4 Seleção de Características

A fim de encontrar inconsistência ou redundância nas características obtidas na etapa de extração, foi aplicado uma seleção de características sobre o descritor gerado. Isso é feito pois a qualidade das características pode afetar no desempenho da classificação. Além disso, ao se identificar essas características, é possível reduzir a dimensionalidade do descritor, tornando o treinamento uma tarefa mais rápida e menos custosa computacionalmente.

O objetivo desta etapa é identificar quais características poderiam contribuir melhor para a discriminação dos sons urbanos. Para isso, foi utilizado o algoritmo de seleção de atributos *Information Gain* ou Ganho de Informação que funciona da seguinte forma: os valores de entrada variam de 0 (sem informação) a 1 (informação máxima), as características que contribuem com mais informações terão um maior valor de ganho, enquanto aquelas que não adicionam muita informação terão uma pontuação mais baixa (KAUR, 2019). Considerando um conjunto de dados S com s amostras pertencentes a m classes distintas, a Equação 32.2 apresenta o *information* (VOGADO et al., 2018) utilizado para classificar uma dada amostra.

$$\text{Info}(S) = - \sum_{i=1}^m p_i \log_2(p_i), \quad (31.13)$$

onde p_i é a probabilidade de uma amostra pertencer a uma classe C_i . A entropia de um determinado atributo A que tem valores v é mostrada na Equação 32.3.

$$\text{Entropy}(A) = - \sum_{i=1}^m \text{Info}(S) \frac{s_{1i} + s_{2i} + \dots + s_{mi}}{s}, \quad (31.14)$$

e a variável S_{ij} representa o número de amostras pertencentes à classe C_i do subconjunto S_i . O atributo gain A é representado pela Equação 32.4:

$$Gain(A) = Info(S) - Entropy(A). \quad (31.15)$$

31.2.5 Classificação

Para o reconhecimento dos padrões presentes nas características extraídas através dos métodos descritos anteriormente, foram utilizados dois classificadores amplamente aplicados pela literatura, o *Random Forest* (RF) (ESMAIL; AHMED; ELTAYEB, 2019) e o *Support Vector Machine* (SVM) (CORTES; VAPNIK, 1995b). Um outro fator importante para a escolha dos mesmos, está na sua arquitetura estrutural, pois o RF resolve o problema com auxílio de árvores, e o SVM com auxílio de vetores de suporte, maximizando, assim, a fronteira de decisão. A seguir, está uma breve descrição dos classificadores utilizados pelo presente trabalho.

- O algoritmo de classificação RF, como o próprio nome sugere, gera um conjunto de árvores de decisão, independentes entre si, com previsões aleatórias. Ou seja, uma floresta formada por árvores de decisão, contendo cada uma um sub-conjunto de indicadores escolhidos aleatoriamente. Aquela que obtiver o maior número de votos será escolhida como modelo de classificação (ESMAIL; AHMED; ELTAYEB, 2019).
- O SVM, proposto por Cortes e Vapnik (1995b), é um algoritmo utilizado em diversos problemas, fazendo parte das classes de algoritmos de aprendizado supervisionados, que analisam os dados e reconhecem padrões existentes nos mesmos, sendo amplamente utilizado para classificação e regressão. O SVM toma como entrada um conjunto de dados e realiza uma predição para cada entrada dada, analisando qual das classes anteriormente definidas a nova entrada adequa-se melhor. Sendo assim, a proposta de Cortes e Vapnik (1995b) torna-se um classificador linear binário não probabilístico. Um modelo SVM pode ser imaginado como composto por exemplos de pontos espaciais, mapeados de maneira que os exemplos de cada categoria sejam divididos por um espaço claro que seja tão amplo quanto possível. Os novos exemplos são então mapeados no mesmo espaço e preditos como pertencentes a uma categoria baseados em qual o lado do espaço eles são colocados. De forma geral, o intuito do SVM é buscar uma linha de separação, mais comumente chamada de *hiperplano* entre os dados categoricamente separados anteriormente. Essa linha procura maximizar a distância entre os pontos mais próximos em relação a cada uma das classes (CORTES; VAPNIK, 1995b).

31.2.6 Métricas de Validação

Para avaliar os resultados da classificação dos áudios, foram utilizadas medidas estatísticas baseadas na análise da matriz de confusão, que é calculada com base em quatro valores: verdadeiro positivo (TP), falso positivo (FP), falso negativo (FN) e verdadeiro negativo (TN). Esses valores servem para indicar o número de amostras classificadas corretamente e incorretamente. As medidas utilizadas foram: Acurácia (Acc), Índice *kappa* (κ) e a área sobre a curva ROC (AUC). A métrica de acurácia, representa a porcentagem de elementos que foram classificados corretamente (BARATLOO et al., 2015b). Ela é descrita como:

$$Acc = \frac{TN + TP}{TN + TP + FN + FP}. \quad (31.16)$$

O índice *kappa* indica como os classificadores utilizados são capazes de superar o classificador que simplesmente adivinha aleatoriamente de acordo com a frequência de cada classe x. De acordo com Cohen (1960), há uma categorização sobre os níveis de desempenho da classificação segundo o índice *kappa*, apresentado na Tabela 222. A Equação 32.6 apresenta a fórmula para calcular essa métrica.

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \quad (31.17)$$

onde p_o é o resultado alcançado e p_e é o resultado esperado.

Tabela 215 – Categorização do desempenho da classificação, de acordo com o índice *kappa*.

Kappa	Qualidade
$\kappa < 0,2$	Ruim
$0,2 \leq \kappa < 0,4$	Razoável
$0,4 \leq \kappa < 0,6$	Bom
$0,6 \leq \kappa < 0,8$	Muito bom
$\kappa \geq 0,8$	Excelente

A métrica AUC fornece informações sobre a capacidade da robustez do modelo em separar as classes usando o gráfico de desempenho fornecido pela curva ROC (*Receiver Operating Characteristic*). A curva ROC tem-se seu cálculo através de diferentes limiares, analisando duas métricas calculadas através da matriz de confusão: a Taxa de Verdadeiro Positivo (TVP) e Taxa de Falso Positivo (TFP), sendo assim uma métrica sensível a proporção de verdadeiros negativos corretamente classificados. Quanto mais próximo de 1 o valor da AUC for, melhor é o desempenho do método na distinção das classes (HANLEY; MCNEIL, 1982).

31.2.7 Trabalhos Relacionados

A literatura dispõe de estudos recentes sobre extração de características sonoras que possam ser utilizadas em algoritmos de classificação dos sinais de áudio. Neste Capítulo são apresentados trabalhos encontrados na literatura que, assim como este trabalho, também utilizam a técnica de extração manual de características para descrever os atributos relevantes dos áudios.

Silva et al. (2019) avaliou técnicas de aprendizagem de máquina para a classificação de sons urbanos em dispositivos embarcados. Eles aplicaram uma abordagem baseada na extração manual de características (*Mel Frequency Cepstral Coefficients* (MFCC), informação espectral, *root mean square* (RMS), e *zero crossing rate*) e classificação. Os autores adotaram o algoritmo *Information Gain* na fase de seleção de características, resultando em um vetor de características com dimensão total de 90. A proposta alcançou uma precisão de classificação de 46,90% com k-NN para o conjunto de dados ESC-50 e 53,50% com Naive Bayes para o conjunto de dados UrbanSound8K.

Giannakopoulos, Spyrou e Perantonis (2019) utilizou *Convolutional Neural Networks* (CNNs) para a extração de características e mel-espectrogramas como entrada para as CNNs na tarefa de reconhecimento de eventos sonoro em ambientes urbanos. O autor extraiu características manualmente diretamente da onda sonora e as combinou com as características extraídas pela CNN. A combinação das duas metodologias resultou em um aumento de 11% no desempenho de classificação com SVM. Eles alcançaram uma acurácia máxima de 79,3 e 54,2% para a UrbanSound8K e a ESC-50, respectivamente. O autor não implementou a fase de seleção de características, resultando em um vetor com uma dimensão total de 580, para cada áudio.

Su et al. (2019) propôs um sistema composto por três componentes: extração e combinação de características, treinamento de uma CNN e fusão em nível de decisão baseada na teoria de DS (Dempster—Shafer), para o reconhecimento inteligente de eventos sonoros. O autor combinou as características extraídas em dois conjuntos distintos, LMC, composto por: log-mel spectrogram, chroma, spectral contrast e tonnetz; e MC, composto por: MFCC, chroma, spectral contrast e tonnetz. Em seguida, ambos os conjuntos são combinados e um espectrograma é criado, sendo utilizado para o treinamento da CNN. O modelo TSCNN-DS, proposto pelo autor, atinge uma acurácia de classificação de 97,2% para o conjuntos de dados UrbanSound8K.

Mushtaq e Su (2020) utilizou três diferentes características sonoras: *Mel spectrogram* (Mel), *Mel Frequency Cepstral Coefficient* (MFCC) e *Log-Mel*. Tais características foram consideradas, pelo autor, como entrada para o modelo de CNN proposto. Para a ESC-10, a ESC-50 e a UrbanSound8K, a acurácia alcançada foi de 94,94%, 89,28%, e 95,37% respectivamente. O autor também adotou técnicas de deformação para o aumento dos dados.

Alves et al. (2020) utilizou a extração de características manuais para a tarefa de classificação de sons urbanos em sistemas embarcados. O autor extraiu as características *Mel Frequency Cepstral Coefficients* (MFCC), Spectral rolloff, Spectral centrid e *zero crossing rate* (ZCR). Gerando um vetor com dimensão total de 46

por áudio. As características foram classificadas utilizando os algoritmos de classificação SVM e K-NN alcançando uma acurácia de 73% para o conjunto UrbanSound8K.

Tabela 216 – Comparação dos trabalhos relacionados elencados.

Trabalho	Objetivo	Características	Seleção de características	Bases	Representação do som	Acurácia (Esc-10/ESC-50/UrbanSound8K)
Silva et al. (2019)	Utilizar características extraídas manualmente para classificação com aprendizagem de máquina.	MFCCs Informação cepstral RMS ZCR	Sim	ESC-50/UrbanSound8K	Onda sonora	-/46,9%/53,5%
Giannakopoulos, Spyrou e Perantonis (2019)	Concatenar características manuais e extraídas pela CNN para maximizar os resultados.	ZCR Energia Entropia da energia Spectral centroid Spectral spread Spectral entropy Spectral flux Spectral rolloff MFCCs Chroma vector Chroma deviation	Não	ESC-50/UrbanSound8K	Melspectrogramas	-/54,2%/79,3%
Su et al. (2019)	Testar diferentes conjuntos de características manuais no treinamento de CNNs.	MFCCs Chroma Spectral Contrast Log-mel spectrogram Tonnetz	Não	UrbanSound8K	Espectrogramas	-/-/97,2%
Mushtaq e Su (2020)	Usar espectrogramas extraídos manualmente como entrada de CNNs.	MFCCs Mel spectrogram Log-Mel	Não	Esc-10/ESC-50/UrbanSound8K	Espectrogramas	94,94%/89,28%/95,37%
Alves et al. (2020)	Concatenar características manuais e extraídas pela CNN para maximizar os resultados.	MFCCs Spectral rolloff spectral centroid ZCR	Não	UrbanSound8K	Espectrogramas	73%

A Tabela 216 apresenta um breve resumo dos trabalhos relacionados. Dentre os trabalhos apresentados apenas o Silva et al. (2019) realizou a etapa de seleção de característica, o que mostra que a maioria dos trabalhos não buscou reduzir a dimensionalidade de seus descritores. Também, nenhum dos trabalhos apresentados realizou uma análise para definir quais características melhor contribuem na classificação de sons urbanos.

31.3 Proposta

As cidades inteligentes se apoiam na expansão da conectividade com a IoT para o monitoramento ubíquo e inteligente através de redes de sensores e *smartphones*. Sendo o som uma importante fonte de informação a respeito da vida urbana (EDWARDS, 2018), a classificação automática dos sons urbanos pode auxiliar na detecção de situações de risco à segurança dos habitantes, melhoria da saúde e gerenciamento do tráfego em espaços urbanos. Diante do exposto, a proposta deste trabalho é a construção de um descritor, através da extração manual de características, capaz de fazer a distinção entre diferentes classes de sons urbanos. E aplicando uma seleção de características, transformar esse descritor em um descritor compacto e preciso com um menor custo computacional, tornando-o adequado para a utilização em aplicativos de reconhecimento de som móvel ou sistemas embarcados. Para isso, foi seguido o passo à passo definido na Figura 352.

31.3.1 Materiais e Métodos

Para o desenvolvimento do descritor proposto neste trabalho, foi utilizada uma metodologia composta por 6 etapas sequenciais, sendo assim, cada etapa utiliza o resultado da etapa anterior. O fluxograma exibido na Figura 352 detalha a metodologia utilizada. A primeira etapa consiste na aquisição dos áudios, adquiridos através de duas bases de dados públicas: UrbanSound8K (SALAMON; JACOBY; BELLO, 2014), e ESC-50 (PICZAK, 2015a). Em seguida, a etapa de pré-processamento busca garantir a uniformização de todos os áudios, o que é de grande importância para etapa de extração de características. Na quarta etapa ocorre a seleção das características extraídas, onde foram identificadas aquelas que não agregam informações relevantes na discriminação dos áudios. A etapa 5 consiste no treinamento dos classificadores para que possam distinguir as diferentes classes existentes. Finalmente, na etapa 6 os resultados obtidos pelos classificadores são avaliados com diferentes métricas de avaliação. As seções a seguir discutem cada uma dessas etapas.

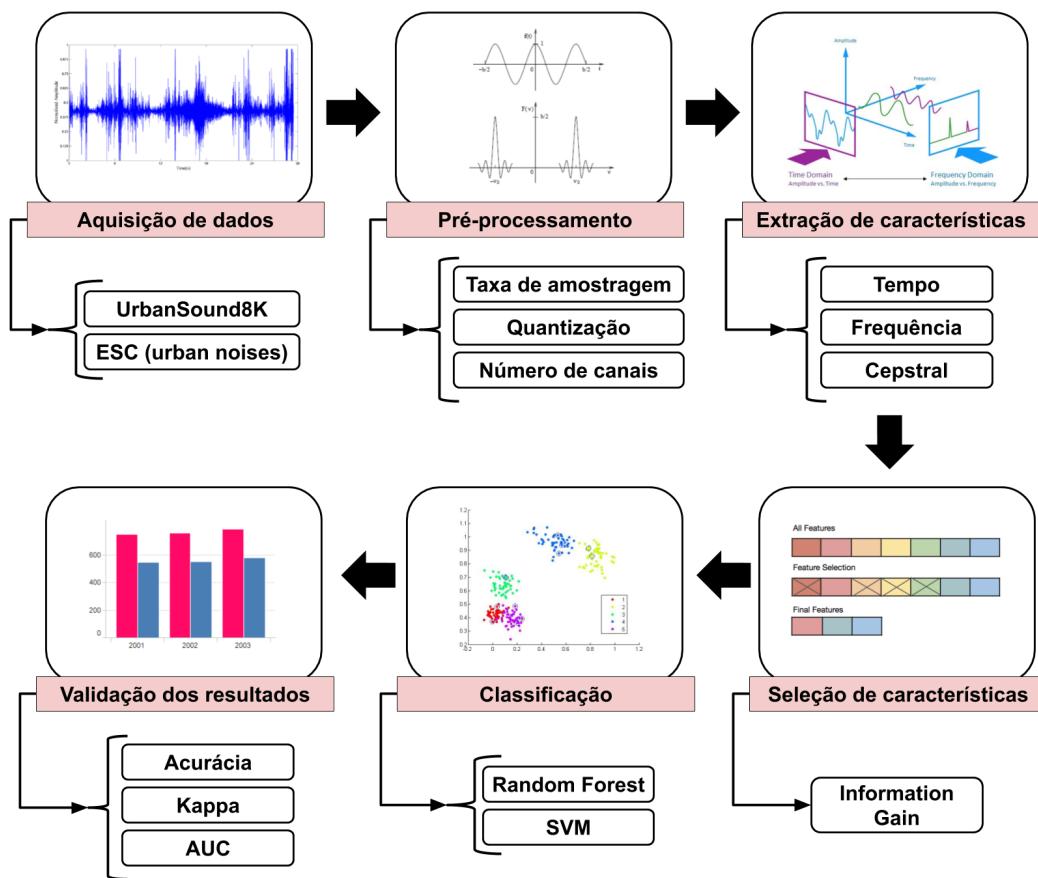


Figura 352 – Metodologia para classificação de sons urbanos.

31.3.1.1 Aquisição dos áudios

Dois dos mais populares conjuntos de dados públicos para reconhecimento de sons urbanos foram utilizadas para a aquisição dos dados: UrbanSound8K ([SALAMON; JACOBY; BELLO, 2014](#)) e ESC-50 ([PICZAK, 2015a](#)). O conjunto ESC-50 é composto por 50 classes de sons diferentes, no entanto, apenas 10 classes são diretamente relacionadas a ambientes urbanos. Assim, neste trabalho foi adotado o subconjunto ESC (urban noises), formado pelas 10 classes de sons urbanos do ESC-50.

O ESC (urban noises) é o subconjunto formado a partir do conjunto de dados ESC-50, ele possui 400 amostras de áudio distribuídas em 10 classes, que, diferentemente do UrbanSound8K, estão平衡eadas, logo, tem o mesmo número de amostras por classe. No entanto, o pequeno número de amostras por classe, cuja divisão pode ser observada na Tabela 224, pode impactar negativamente no desempenho dos classificadores. O ESC (urban noises) é um conjunto de dados homogêneo, onde todas as amostras têm uma duração fixa de 5 segundos e um único canal.

O UrbanSound8K é um conjunto de dados formado por 8732 áudios, exclusivamente urbanos, com duração igual ou inferior a 4 segundos agrupados em 10 classes diferentes. No entanto, suas classes são desbalanceadas, logo, o número de amostras de uma classe pode chegar a ser três vezes maior que o de outra, assim como demonstrado na Tabela 224. O UrbanSound8K é um conjunto de dados heterogêneo, visto que a taxa de amostragem, a codificação e o número de canais podem variar entre as amostras. Na Tabela 224, as classes comuns às duas bases estão em destaque.

Tabela 217 – Divisão de classes por conjunto de dados.

UrbanSound8K			ESC (urban noises)		
Classes	Número de amostras	Duração por classe (s)	Classes	Número de amostras	Duração por classe (s)
Ar condicionado	1000	3994.9287	Helicóptero	40	200
Buzina de carro	429	1053.9533	Buzina de carro	40	200
Crianças brincando	1000	3961.8745	Motoserra	40	200
Latido	1000	3148.7496	Fogos de artifício	40	200
Perfurador	1000	3548.244	Motor	40	200
Motor ocioso	1000	3935.9925	Trem	40	200
Tiro	374	616.7965	Sinos de igreja	40	200
Britadeira	1000	3610.9747	Avião	40	200
Sirene	929	3632.7016	Sirene	40	200
Música de rua	1000	4000.0	Serrote	40	200

31.3.1.2 Pré-processamento

Para realizar a manipulação dos áudios nas bases, foi utilizada a biblioteca *LibROSA* (MCFEE et al., 2015). Esta biblioteca normaliza por padrão os dados no intervalo [-1,1], e transforma os sinais em mono ao fazer a média das amostras entre os canais. Além disso, ela garante que todas as amostras possuam uma mesma taxa de amostragem, de 22050 Hz e uma quantização de 16 bits (MCFEE et al., 2015) (Tabela 218).

Ambos os conjuntos de dados passaram pelo pré-processamento mencionado, dessa forma, nenhuma das amostras utilizadas na etapa de extração de características, tampouco as etapas seguintes, possuem uma configuração diferente. A Tabela 218 mostra a configuração utilizada.

Apesar de no conjunto de dados UrbanSound8K as amostras possuírem diferentes tamanhos de duração, isso não impacta na extração de características, pois a dimensão de cada característica se mantém independente da duração do áudio. As características com variações em sua dimensão, foram transformadas em um único valor calculando sua média. Por esse motivo a duração das amostras não foi uma das configurações modificadas.

Tabela 218 – Configuração das amostras.

Medida	Valor
Taxa de amostragem	22050 Hz
Quantização	16 bits
Canais	1

31.3.1.3 Extração de características

Na fase de extração de características, também foi aplicada a biblioteca *LibROSA*, que possui uma extensa documentação com inúmeras funções para viabilizar o processamento de áudio. Ela fornece os componentes necessários para criar sistemas de recuperação de informações sonoras. Através das funções da *LibROSA*, foram extraídas características nos diferentes domínios: tempo, frequência e cepstral. Todas as características extraídas foram descritas na Seção 31.2.3.

Ao final da etapa de extração, obteve-se um descritor com uma dimensão total de 82 posições, ou seja, um conjunto de diferentes características capaz de descrever um som, de modo que seja possível à um classificador agrupar em classes, que, no caso deste trabalho, são diferentes classes de sons urbanos. O resumo das características extraídas neste trabalho pode ser visto na Tabela 219.

Tabela 219 – Características em seus diferentes domínios e dimensões.

Característica	Domínio	Dimensão
MFCC	Cepstral	60
Delta1	Cepstral	1
Delta2	Cepstral	1
ZCR	Tempo	1
Spectral rolloff	Tempo	1
Spectral centroid	Tempo	1
RMS	Tempo	1
Tempogram	Tempo	1
Chroma STFT	Frequência	1
Choma CQT	Frequência	1
Chroma CENS	Frequência	1
Entropy	Frequência	1
Spectral Flatness	Frequência	1
Spectral bandwidth	Frequência	1
Spectral contrast	Frequência	7
Poly	Frequência	1
Tonnez	Frequência	1

31.3.1.4 Seleção de características

A seleção das características que deveriam ser mantidas foi feita baseada na lista ranqueada gerada pelo algoritmo de seleção *Information Gain*. Com base em sua posição no ranque, as características foram consideradas como possuindo maior ou menor relevância para distinção dos sons, sendo consideradas com maior relevância as que possuíam posições mais altas no ranque e menor relevância as que possuíam posições mais baixas.

Para verificar a influência das características e dos subconjuntos de características, os testes foram realizados agrupando as características em grupos de 5, variando em um intervalo de 5 a 82, com acréscimos de 5. Após ser definido dentro desse intervalo qual o menor número de características que oferecia uma acurácia maior ou igual a acurácia alcançada com todas elas, um novo intervalo é testado, a partir do valor encontrado no intervalo anterior, testando agora em incrementos de 1, o desempenho com as 5 características anteriores e as posteriores a este valor.

Todos os teste realizados nesta etapa foram executados usando o software WEKA, uma suíte de algoritmos de mineração de dados e *Machine Learning*, desenvolvida pela University of Waikato, Nova Zelândia. Essa suíte contém ferramentas para pré-processamento de dados, seleção, classificação, regressão, agrupamento, regras de associação e visualização (HALL et al., 2009).

31.3.1.5 Classificação

Para verificar se as características resultantes da etapa de seleção possuem de fato eficiência para distinção das classes de áudio, foram utilizados os seguintes classificadores: *RF* e *SVM*. Esses algoritmos foram escolhidos levando em considerando sua ampla adoção na literatura relacionada, como mencionado na Seção 34.2. Os resultados foram avaliados utilizando 20% do conjunto de dados total para testes e 80% para treinamento, divididos aleatoriamente. Uma vez divididos, os mesmos conjuntos foram utilizados em todos os testes.

Assim como na etapa anterior, também foi utilizada a ferramenta WEKA nesta etapa, já que ela disponibiliza os algoritmos de classificação utilizados nesta metodologia. Em relação aos parâmetros de cada classificador, foram utilizados os valores padrão da ferramenta WEKA. Contudo, alguns parâmetros foram variados durante a execução do método com o intuito de encontrar a melhor configuração para os modelos. Esses parâmetros foram o número de árvores, no *RF*, e o *kernel*, no *SVM*. As variações desses parâmetros estão detalhadas na Seção 31.4.2.

31.3.1.6 Validação dos resultados

Como mencionado na Seção 34.2, para avaliar os modelos de classificação, foram utilizadas três métricas de validação comumente aplicadas na literatura: acurácia, índice *kappa* e AUC. As métricas escolhidas para realizar a validação dos resultados obtidos pelo presente trabalho são amplamente utilizadas por conseguirem demonstrar de forma clara a proporção de erros e acertos obtidos pelos classificadores, por esse motivo foram utilizadas neste trabalho. As três métricas estão disponíveis na avaliação dos classificadores pela ferramenta WEKA, sendo assim, os valores foram obtidos através dela.

31.4 Resultados

Levando em consideração a metodologia descrita na Seção 32.4, este capítulo apresenta os resultados obtidos pela extração, seleção e classificação das características presentes no descriptor proposto por esse trabalho. O descriptor proposto resultou em um vetor com dimensão total de 82 posições, composto pelas características apresentadas na Tabela 219.

A discussão dos resultados é dividida em três seções: na Seção 31.4.1 são apresentados os resultados obtidos com a seleção de características do descriptor, enfatizando as características mais relevantes para a classificação; na Seção 31.4.2 são apresentados os resultados alcançados por cada classificador; por fim, na Seção 31.4.3, os resultados da proposta são comparados com os trabalhos correlatos.

31.4.1 Seleção de Características

A etapa de seleção permitiu a redução da dimensionalidade do descriptor através da seleção de características importantes para a caracterização dos sons urbano. A Figura 353 apresenta a acurácia alcançada pelo algoritmo de classificação *Randon Forest* (RF) em função do número de características usadas nos testes, bem como os dois conjuntos de dados se comportaram durante os testes de seleção, sendo os algoritmos de classificação treinados com o conjunto de características selecionadas pelo algoritmo de seleção *Information Gain*.

Observa-se, através da Figura 353, que os conjuntos de dados apresentam padrões diferentes. Para o conjunto de dados UrbanSound8K, a acurácia melhora à medida que novas características são adicionadas, consequentemente, a maior acurácia é obtida com todas as 82 características, o que corresponde a 94,2153%. No entanto, com um subconjunto de 35 características, o RF atinge 93,2417% de acurácia. O conjunto de dados ESC (urban noises) mostra um padrão irregular, o que pode sugerir que certos subconjuntos de características podem conter informações mais inconsistentes com impacto no desempenho da classificação. A melhor acurácia foi de 72,5% com todas as características, por outro lado, com 20 características, a acurácia é de 71,25%. De acordo com a Figura 353, com um grupo de pelo menos 35 características foi possível alcançar um desempenho significativo para ambos os conjuntos de dados.

A Figura 354 apresenta a distribuição das primeiras 35 características classificadas pelo *Information Gain*. As características são apresentadas em grupos de 5, variando de 5 a 35. O *Information Gain* selecionou 14 de 20 e 27 de 35 das características em comum entre os conjuntos de dados. A Figura 354 revela a predominância de características de forma espectral entre as primeiras 15 posições de características para ambos os conjuntos de dados. Além disso, o *Tempogram* e as características relacionadas ao *Chroma* ficaram nas primeiras 5 posições selecionadas.

Os MFCCs foram divididos em 3 grupos, cada um contendo 20 coeficientes. O primeiro, segundo e terceiro grupo de MFCCs foram nomeados como MFCC (0-20), MFCC (20-40) e MFCC (40-60), respectivamente. O MFCC (0-20) passa a ser selecionado entre a sexta e a décima posição e atinge uma frequência alta entre 20^a e 25^a posição. O MFCC (20-40) só aparece após a 15^a posição para ESC (urban noises) e a 20^a posição para UrbanSound8K. Os últimos 20 coeficientes MFCC, o conjunto MFCC (40-60), aparecem na 20^a característica. Ambos MFCC (20-40) e MFCC (40-60) são menos frequentes do que MFCC (0-20). Além disso, a característica *Spectral roll-off*, de domínio do tempo, aparece nos subconjuntos que oferecem uma acurácia significativa para ambos os conjuntos de dados.

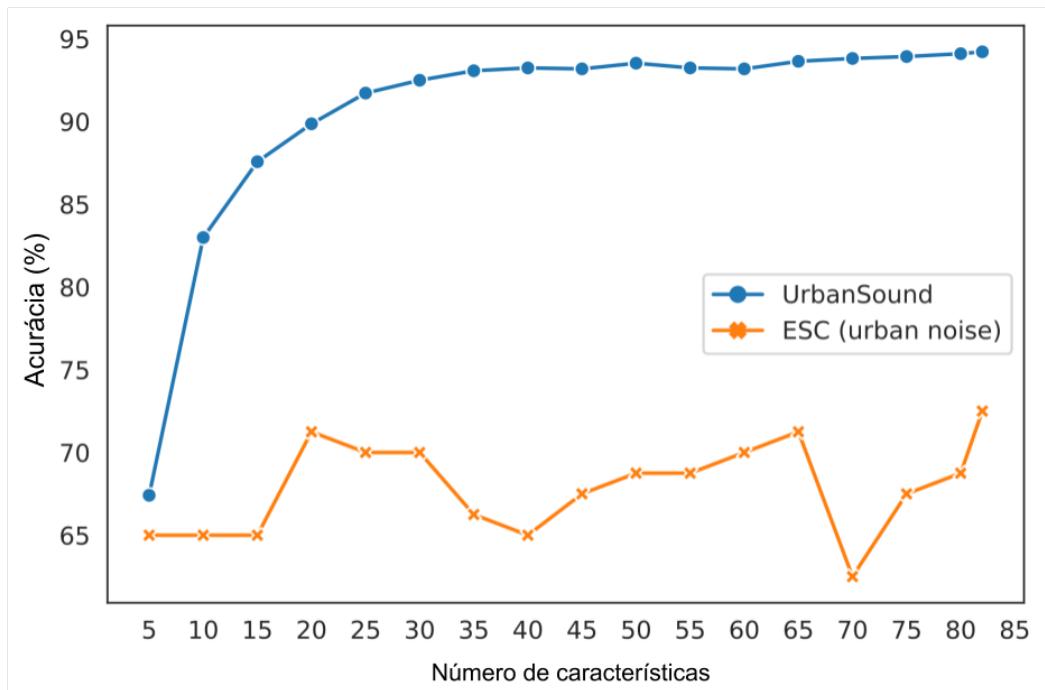


Figura 353 – Acurácia da seleção de características com o classificador RF.

31.4.2 Classificação

A Tabela 220 apresenta os resultados alcançados pelo descriptor resultante das etapas de extração e seleção de características, utilizando os classificadores *Random Forest* (RF) e SVM, descritos anteriormente na Seção 31.3.1.5.

Para definir o número de árvores a ser utilizado, foram realizados testes variando na faixa de 50 a 450, incrementado em 50. Depois de definir o número mais adequado nesta escala, foi avaliado o desempenho do classificador em incrementos de 10. O algoritmo de classificação SVM, usado para classificação e análise de regressão, teve seu desempenho avaliado usando quatro funções de *kernel*: *radial*, *linear*, *polynomial* e *sigmoid*. O *kernel sigmoid* obteve os melhores resultados para os dois conjuntos de dados.

A Figura 355 apresenta o desempenho do classificador *Random Forest* de acordo com a variação do número de árvores selecionadas. Com 350 árvores, o RF obteve o melhor desempenho, de 94,21%, para a UrbanSound. Considerando o conjunto de dados ESC (urban noises), o *Random Forest* obteve a maior acurácia em 50 árvores, com 71,25%. No entanto, os incrementos de 10 mostraram que 70 árvores apresentaram melhor desempenho, com 72,5% de acerto. O número inferior de amostras na base ESC (urban noises) faz com que seja necessário um número menor de árvores, isso porque, aumentando o número de árvores, aumenta também as chances de overfitting para o conjunto.

A configuração usada para todos os experimentos do conjunto de treinamento UrbanSound foi, 350 árvores e *seed* (semente) igual a 1, para o classificador *Random Forest*, e *kernel sigmoid* para o classificador SVM. Já para o conjunto ESC (urban noises), o *Random Forest* com 70 árvores e *seed* igual a 1, e o SVM também com o *kernel sigmoid*.

Observando a Tabela 220, é possível notar que, aplicando a seleção de características sobre o descriptor, pode-se chegar a uma redução de dimensionalidade. Isso pode ser melhor observado no caso do conjunto de dados ESC (urban noises), na classificação através do RF, que, apesar de não ter alcançado melhores resultados de acurácia, teve sua dimensionalidade reduzida de 82 posições para apenas 18. Também no

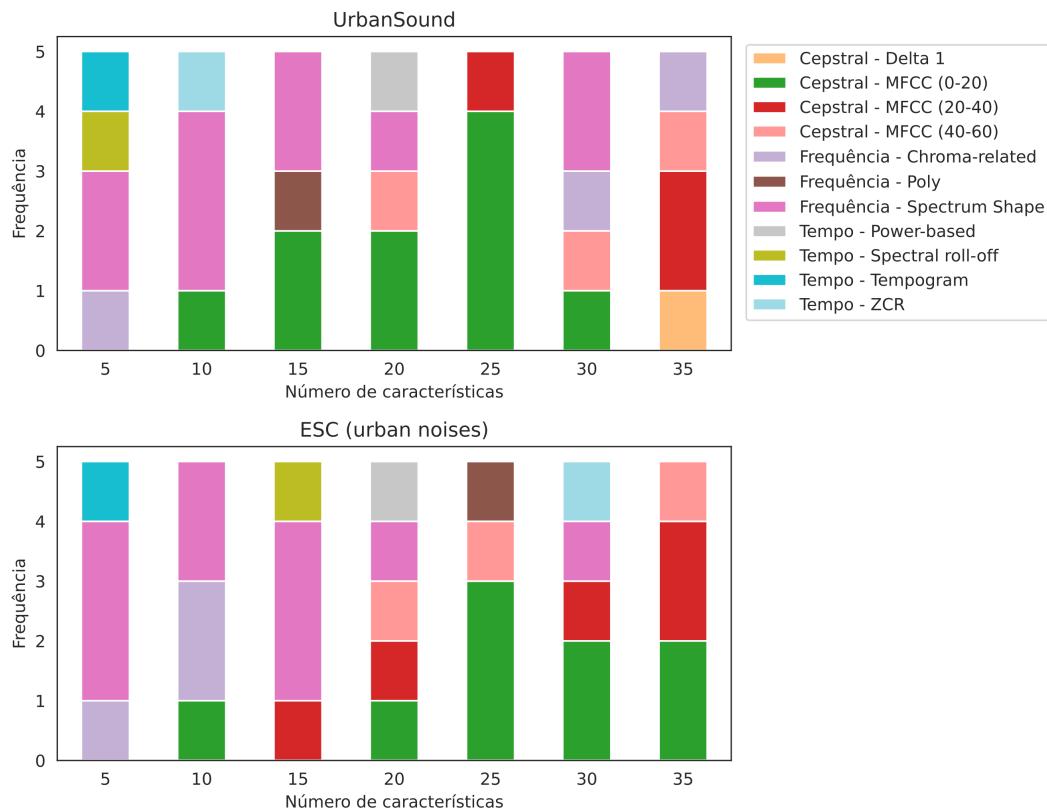


Figura 354 – Distribuição das 35 melhores características selecionadas pelo *Information Gain*.

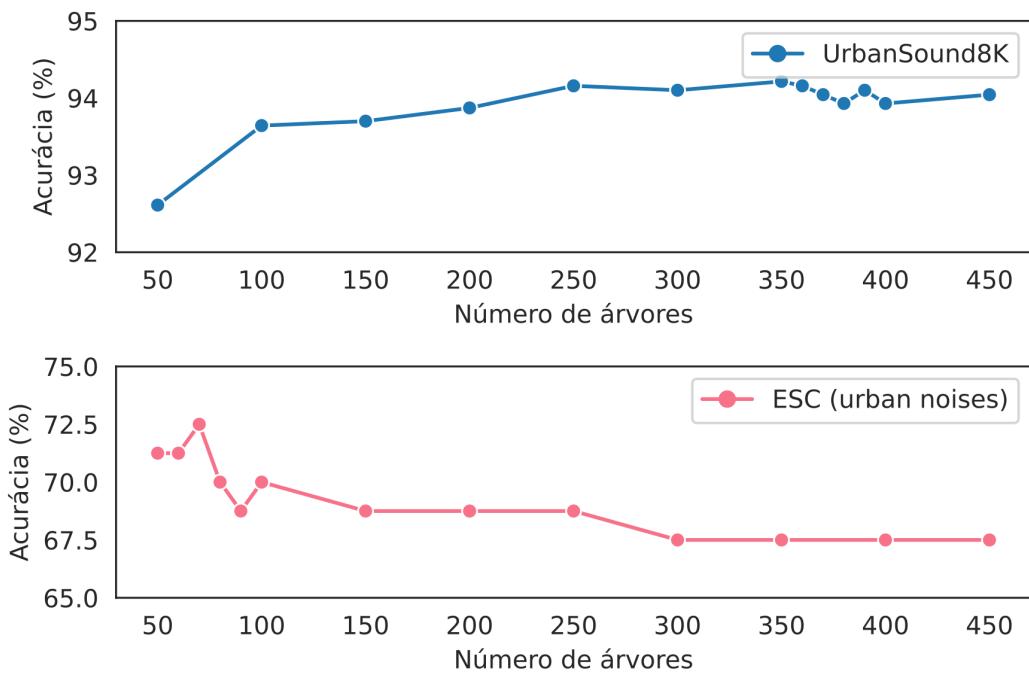


Figura 355 – Acurácia de acordo com o número de árvores do classificador RF.

Tabela 220 – Resultados com e sem seleção de características. Os números entre parênteses representam a dimensão do vetor.

Conjunto de dados	Classificador	Sem seleção de características			Com seleção de características		
		Acurácia	Kappa	AUC	Acurácia	Kappa	AUC
UrbanSound8K	RF	94,215% (82)	0,9351	0,996	94,100% (80)	0,9338	0,996
UrbanSound8K	SVM	84,822% (82)	0,8296	0,914	84,937% (80)	0,8309	0,915
ESC (urban noises)	RF	72,500% (82)	0,6927	0,952	72,500% (18)	0,6917	0,961
ESC (urban noises)	SVM	71,250% (82)	0,6768	0,839	78,750% (70)	0,7606	0,881

conjunto de dados ESC (urban noises), porém com classificador SVM, a seleção de características não só garantiu a redução de dimensionalidade, passando de 82 para 70 características no descritor, como também melhorou o desempenho de classificação com um aumento de 7,5% na acurácia, indo de 71,250% para 78,750%.

Já o conjunto de dados UrbanSound8K não mostrou nenhuma mudança significativa após a seleção de características, o que já era esperado baseado no padrão observado na Figura 353, onde, de acordo com que novas características iam sendo acrescentadas, sua acurácia aumentava. Também na Figura 353, é possível observar que, diferente do comportamento do conjunto ESC (urban noises), que se mostrou irregular tendo grandes mudanças de acurácia dependendo do subconjunto de características, o conjunto UrbanSound8K se manteve sempre constante. Isso pode ser justificado pela diferença na quantidade de amostras por base.

31.4.3 Comparação com a literatura

A Tabela 221 sumariza uma comparação entre os resultados da abordagem proposta neste trabalho e os resultados dos estudos descritos na Seção 32.3. Os critérios de comparação, além dos resultados, foram a dimensão do descritor e o fato de utilizar ou não a seleção de características. Vale ressaltar que, alguns dos trabalhos relacionados possuem valores de acurácia para base ESC, relacionados a todo o conjuntos de dados ou um subconjunto de dez classes (não relacionado ao som urbano). Já este trabalho apresenta valores de acurácia da ESC (urban noise), subconjunto formado pelas 10 classes de sons urbanos da Esc-50.

Tabela 221 – Comparação do método proposto com os trabalhos relacionados.

Trabalho	Seleção de características	Extração com CNN	Dimensão do vetor	UrbanSound8K (Acurácia)	Esc-10/Esc-50 (Acurácia)
Silva et al. (2019)	Sim	Não	90	53.5%	-/46.9%
Giannakopoulos, Spyrou e Perantonis (2019)	Não	Sim	580	79.3%	-/54.2%
Su et al. (2019)	Não	Sim	LMC(41 x 85) MC(41 x 85)	97.2%	-
Mushtaq e Su (2020)	Não	Sim	Melspectrogram-(28 x 128) Logmel(128x128) MFCC(20x128)	95.37%	94.94%/89.28%
Alves et al. (2020)	Não	Não	46	73%	-
Este trabalho	Sim	Não	82	94.2%	72,500%/-

A Tabela 221 mostra que o descritor proposto neste trabalho consegue representar, de maneira satisfatória, os dados para a classificação entre os diferentes tipos de sons urbanos, tendo em vista que a acurácia alcançada foi mais alta até mesmo que os modelos que utilizaram CNNs para a extração de características. Além disso, o descritor proposto apresentou uma dimensionalidade menor que os estudos relacionados, o que contribui para uma otimização na integração dessas características em ambientes de produção.

O trabalho de Mushtaq e Su (2020) apresentou uma acurácia superior ao método proposto neste trabalho. Contudo, os autores aplicaram representações diferentes para os sinais acústicos, com exceção do MFCC, que também foi utilizado neste trabalho. Ademais, por utilizar uma CNN, as representações utilizadas pelos autores podem ter impactado positivamente os resultados, proporcionando um melhor desempenho. Todavia, esta abordagem traz consigo um aumento no custo computacional em virtude da utilização da CNN, em contraste ao nosso método, que utiliza um descritor com uma dimensionalidade menor, mas que apresentou resultados promissores.

31.4.4 Discussão

Com base nos resultados apresentados na Seção 31.4, é possível perceber que diferentes conjuntos de características podem proporcionar diferentes desempenhos ao classificador, não sendo o ranqueamento de características fornecido pelo *Information Gain* o único determinante do grau de importância das características para distinção da classe sonora. Isso pode ser justificado através da observação do comportamento do conjunto de dados ESC (urban noises), uma vez que, mesmo adicionando características ranqueadas em boas posições pelo seletor, a acurácia sofre uma queda. A presença de informação com ruído nessas características pode explicar porque isso acontece.

O conjunto de dados *UrbanSound8K*, apesar de não ter obtido um melhor resultado com a redução de características, para um subconjunto de 35 características obteve uma acurácia muito próxima ao melhor resultado, com uma diferença um pouco maior que 1%. A redução da dimensionalidade produz um descriptor mais compacto, isso pode ser favorável para adoção do descriptor proposto em aplicações embarcadas, onde um menor custo computacional é desejado.

Percebe-se também que as características espectrais possuíram maior influência para a caracterização dos sons urbanos, tendo em vista a distribuição demonstrada na Figura 354, onde é possível perceber que as características espectrais se encontram, em ambos os conjuntos de dados, entre as primeiras 15 características ranqueadas. Dentre as características *MFCCs*, que foram divididas em 3 grupos com 20 coeficientes cada, o primeiro conjunto também se mostrou presente, em ambos os conjuntos, entre as características ranqueadas como de maior importância.

Durante os testes de parâmetro para o algoritmo de classificação *RF*, nos conjuntos *UrbanSound8K* e *ESC (urban noises)*, o modelo alcançou melhor desempenho com diferentes números de árvores, sendo 350 para o *UrbanSound8K* e 70 para o *ESC (urban noises)*. É possível que o baixo número de amostras disponível no conjunto *ESC (urban noises)* tenha influenciado nas diferenças existentes entre os resultados alcançados pelas bases, já que, por possuir um menor número de amostras.

Entre os modelos de classificação aplicados o algoritmo *RF* se destacou obtendo os melhores resultados. Nos testes realizados com o descriptor em sua dimensão total, o *RF* chegou a melhores valores de acurácia em ambos os conjuntos de dados, tendo uma diferença de 9,39% em comparação ao *SVM* no conjunto *UrbanSound8K*. Nos testes após ser aplicada a seleção de características o classificador *RF* também alcançou um melhor resultado que o *SVM* no conjunto *UrbanSound8K*. No entanto, no conjunto de dados *ESC (urban noises)* o algoritmo de classificação *SVM* obteve uma melhor acurácia, com 6,25% a mais em relação ao *RF*.

31.5 Conclusão

Este capítulo apresentou uma metodologia para a extração de características manuais para classificação de som ambiental. Além disso, a etapa de seleção de características resultou em descritores compactos com dimensão 80 e 70 que alcançaram 94,1 e 78,75% de acurácia, respectivamente, para os conjuntos de dados *UrbanSound8K* e *ESC (urban noises)*. Levando em consideração o resultado inicial alcançado pelo descriptor para o conjunto *Esc(urban noise)* (71,25%) e o resultado após a seleção de características (78,75%), percebe-se um aumento de 7,5% na acurácia. Pode-se concluir então que, a aplicação do método de seleção de características adequado pode melhorar o desempenho da classificação. Ao mesmo tempo, reduz notavelmente o tamanho do descriptor, podendo diminuir em alguns casos até 78% do tamanho original, o que torna esses descritores adequados para serem usados em aplicativos de reconhecimento de som móvel ou sistemas embarcados.

Com base nos resultados alcançados, é possível concluir que a extração manual de características em conjunto com a etapa de seleção é, de fato, uma forma eficiente de se obter um descriptor capaz de realizar a discriminação de diferentes classes de sons encontradas no ambiente urbano. Na base *UrbanSound8K*, o método proposto alcançou acurácia de 94,1%, *AUC* de 0,99, *Kappa* de 0,93 e um descriptor com dimensão 80, sendo o índice *Kappa* considerado excelente de acordo com a categorização de [Cohen \(1960\)](#). Já na base *ESC (urban noises)*, apesar das poucas amostras, o método obteve acurácia de 78,75%, *AUC* de 0,88, *Kappa* de 0,76 e um descriptor com dimensão 70, com um *Kappa* considerado muito bom, de acordo com a categorização de [Cohen \(1960\)](#).

Apesar do melhor resultado na UrbanSound8K ter sido sem a seleção de características, a diferença entre os resultados com e sem seleção foi quase imperceptível, com apenas 0,11% de diferença. Outro ponto importante é que, no teste executado com 35 características, o método obteve uma acurácia de 93,24% na UrbanSound8K, bem próximo ao resultado máximo alcançado pelo método. Sendo assim, nota-se que a utilização da seleção torna-se mais relevante para o método, levando em consideração que a redução da dimensionalidade do descriptor pode reduzir o custo computacional da classificação.

Em comparação aos estudos relacionados, a abordagem proposta neste trabalho mostrou-se promissora diante dos dois conjuntos de dados utilizados na avaliação. Além disso, o descriptor proposto resultou em uma dimensionalidade menor de características, o que pode reduzir o custo computacional em comparação a outros métodos comumente utilizados, como no caso de algumas redes neurais. Em suma, este trabalho propôs um descriptor com baixa dimensionalidade e com um significante desempenho de classificação, concentrando informações mais relevantes do problema em decorrência da seleção de características.

Em relação às limitações deste trabalho, vale destacar o não tratamento do desbalanceamento da base UrbanSound8K, que possui um número limitado de amostras nas classes Buzina de carro e Tiro, o que pode impactar na classificação. Além disso, um aumento de dados poderia ser útil para melhorar os resultados na base ESC (urban noises), já que ela possui poucas amostras por classe, sendo esta uma dificuldade para obter um bom desempenho nos modelos de classificação. Com o intuito de mitigar esses problemas e melhorar os resultados no geral, pretende-se realizar os seguintes trabalhos futuros:

- Realizar o balanceamento da base UrbanSound8K através de técnicas de *undersampling* ou *oversampling*;
- Aplicar um aumento de dados na base ESC (urban noises) para ampliar o conjunto de dados em cada classe, podendo ser feito através da concatenação de classe ou aplicando técnicas de transformações nos áudios;
- Fundir as características manuais com características de CNNs para verificar se a união destes descriptores proporcionam melhores resultados;
- Avaliar outros classificadores presentes na literatura, como o XGBoost, o MLP e o KNN;
- Avaliar outras bases de sons urbanos disponíveis publicamente;
- Avaliar outros algoritmos de seleção de características ou de redução da dimensionalidade, como o PCA.

32

CAPÍTULO

Processamento e Classificação de Cenários Acústicos para Sons Urbanos

Autores: Jederson Sousa Luz; Deborah Maria Vieira Magalhães.

Resumo: Nas cidades, o som é uma fonte valiosa de informação para oferecer serviços aos cidadãos, como redução da poluição sonora e sistema de vigilância. No entanto, a tarefa de classificação de som ambiental é desafiadora porque o cenário urbano é altamente heterogêneo, com várias fontes sonoras de diferentes naturezas muitas vezes sobrepostas. A maioria dos sons ambientais são não estacionários, não estruturados e são pequenos com relação sinal/ruído; em consequência, características manuais muitas vezes caracterizam um som específico e não podem ser extrapolados para outro. Diante do cenário, este trabalho propõe o desenvolvimento de uma representação compacta e eficaz capaz de caracterizar diferentes sons urbanos a partir das características profundas extraídas através da uma rede neural convolucional. Para este fim, foi desenvolvido e treinado um novo modelo rede baseado na arquitetura LeNet com mél-spectrogramas de dois conjuntos de dados públicos. O descritor proposto neste trabalho alcançou 95,6% e 81,2% de acurácia de classificação para UrbanSound8K e ESC (urban noises), respectivamente, com uma divisão de treino e teste de 80/20. Os resultados mostraram-se promissores quando comparados às abordagens dos estudos relacionados, que utilizam arquiteturas mais profundas e com uma quantidade maior de parâmetros do que o modelo proposto. Os resultados promissores enfatizam o desempenho das características profundas para a representação dos sinais acústicos.

Palavras-chaves: Sons Urbanos, Vigilância de Áudio, Detecção de Som, Extração de Características, *Convolutional Neural Networks*.

32.1 Introdução

Nas cidades, o som é uma fonte valiosa de informação para oferecer serviços aos cidadãos, como redução da poluição sonora (NAVARRO et al., 2020) e sistema de vigilância (LAFFITTE et al., 2019). Esses aplicativos podem melhorar os aspectos essenciais da vida urbana, incluindo saúde, imóveis e segurança (RATHORE et al., 2016). No entanto, a tarefa de classificação de som ambiental (ESC, do inglês *Environmental Sound Classification*) é desafiadora porque o cenário urbano é altamente heterogêneo, com várias fontes sonoras de

diferentes naturezas (pássaros, sirenes, música de rua, motor) muitas vezes sobrepostas (BELLO; MYDLARZ; SALAMON, 2018).

Tradicionalmente, os trabalhos elaboravam características acústicas relevantes a partir da forma de onda de áudio para distinguir entre diferentes categorias de som (MEDHAT; CHESMORE; ROBINSON, 2020). A maioria dos sons ambientais são não estacionários e apresentam uma pequena relação sinal/ruído (MUSHTAQ; SU, 2020); em consequência, características manuais muitas vezes caracterizam um som específico e não podem ser extrapolados para outro. Assim, eles precisam ser combinados com outras características, levando a descritores de alta dimensão (SERIZEL et al., 2018). Nos últimos anos, modelos de Deep Learning (DL) têm sido aplicados com sucesso na tarefa de detecção de sons urbanos, e, entre eles, a Convolutional Neural Network (CNN) é uma das mais amplamente adotadas (DEMIR; ABDULLAH; SENGUR, 2020; SU et al., 2019; PARK; YOO, 2020; CRAMER et al., 2019).

As arquiteturas profundas podem usar diretamente o sinal bruto (ABDOLI; CARDINAL; KOERICH, 2019) ou uma representação tempo-frequência, chamada spectrograma. A maioria dos modelos de CNN são treinados com imagens de spectrograma de áudio, uma vez que são uma representação mais compacta e resistente a ruído do que formas de onda brutas (ABDOLI; CARDINAL; KOERICH, 2019). Além disso, os modelos de CNN podem ser aplicados de uma maneira fim a fim para classificação de som ambiental (MEDHAT; CHESMORE; ROBINSON, 2020; PARK; YOO, 2020), ou as características profundas podem ser extraídas das camadas totalmente conectadas da CNN e, em seguida, aplicadas como entrada de um classificador convencional. Esta última abordagem supera as características manuais para a classificação de som urbano (DEMIR; ABDULLAH; SENGUR, 2020; GIANNAKOPOULOS; SPYROU; PERANTONIS, 2019).

No entanto, alguns modelos mais profundos requerem um grande número de parâmetros treináveis. Eles oferecem grande capacidade de discriminação, mas são mais suscetíveis a overfitting (ARAUJO et al., 2018), podendo extrair padrões não representativos dos dados, diminuindo sua capacidade de generalização. Além disso, redes muito profundas possuem um custo computacional elevado e podem levar uma quantidade considerável de tempo para o treinamento devido ao número significativo de parâmetros treináveis que deverão ser ajustados durante o treinamento da rede.

Ao se adquirir dados através de bases públicas, problemas com custo de instalação e investimento com tempo de monitoramento dos sons são solucionados, com tudo, surgem novos problemas, pois não é possível assegurar a qualidade do sinal, a quantidade de dados pode ser insuficiente e podem ser provenientes de diferentes sensores, o que os torna heterogêneos.

32.1.1 Objetivos

Diante do exposto, o objetivo geral deste trabalho é a definição de uma arquitetura de Deep Learning para a extração de características que ofereça uma boa discriminação entre as classes, contribuindo para a etapa de detecção de situações de risco à segurança dos habitantes em espaços urbanos.

Os objetivos específicos deste trabalho são:

1. Investigar a viabilidade da utilização de uma arquitetura de rede neural para realizar a extração de características representativas para a distinção dos sinais de áudio;
2. Aplicar algoritmos de classificação clássicos como, *Random Forest* e *Support Vector Machine*, a fim de avaliar o descritor proposto; e
3. Investigar a seleção de características aplicada ao descritor extraído das camadas totalmente conectadas da CNN a fim de reduzir dimensionalidade;
4. Construir uma metodologia para classificação dos sinais acústicos que contribua para a detecção de situações de risco à segurança no contexto urbano.

32.1.2 Organização do Trabalho

Este capítulo está organizado nas seguintes seções: Seção 34.2 apresenta a fundamentação teórica para o entendimento da metodologia abordada neste trabalho. A Seção 32.3, são detalhados os principais trabalhos

relacionados. Na Seção 32.4, são detalhadas as etapas de desenvolvimento da pesquisa, iniciando pela aquisição das bases de áudio, pré-processamento destes áudios, aprendizado de características, seleção de características, classificação e validação da metodologia proposta. A Seção 32.5 apresenta e discute os resultados obtidos com a execução da metodologia proposta. E, por fim, a Seção 32.6 discute os principais achados e perspectivas futuras.

32.2 Referencial Teórico

32.2.1 Som

O som é produzido por alguma vibração mecânica em contato com o ar (por exemplo, o tampo de um violão) ou uma rápida modulação no fluxo de ar (por exemplo, o vocal dobra periodicamente interrompendo a expulsão de ar dos pulmões ao falar ou ao cantar). Esses fenômenos criam oscilações da pressão do ar que se propagam através deste meio (LEMAITRE; GRIMAUT; SUIED, 2018), conforme demonstra a Figura 356. Essas ondas sonoras podem ser convertidas novamente em movimento físico pelo tímpano ou diafragma de um microfone e geralmente são representadas como uma forma de onda.

Quando uma onda é criada, a distância entre uma compressão e a próxima é chamada de comprimento de onda. Quanto mais rápido as ondas sonoras passam por um determinado ponto, menor é o comprimento de onda e maior a frequência. A frequência de um som é o que determina qual nota musical ele representa, sendo a mais conhecida a frequência de 440Hz, a nota lá. A frequência ou altura, também determinam se um som é grave ou agudo, onde o som grave é aquele mais “grosso”, enquanto o agudo é o mais “fino”.

As vibrações também podem comprimir as moléculas de ar com muita força ou suavidade. Esse aperto é chamado de amplitude e é representado na metade superior da Figura 356. A metade inferior desta mesma Figura é uma representação da pressão do ar durante uma onda sonora. A linha horizontal representa a pressão normal do ar.

A amplitude é uma medida da força ou intensidade da onda. Por exemplo, ao observar uma onda sonora, a amplitude medirá a intensidade ou volume do som. A unidade de medida utilizada para representar essa propriedade é o bel (decibel).

Outra característica importante do som é o timbre, pois ele é o que nos permite distinguir diferentes sons, mesmo que tenham a mesma frequência e mesma intensidade, como vozes de coral e instrumentos musicais, como instrumentos de corda, instrumentos de sopro e instrumentos de percussão. Isso ocorre porque cada material produz ondas sonoras com formatos completamente distintos, o que faz com que cada instrumento ou voz possa ser reconhecida.

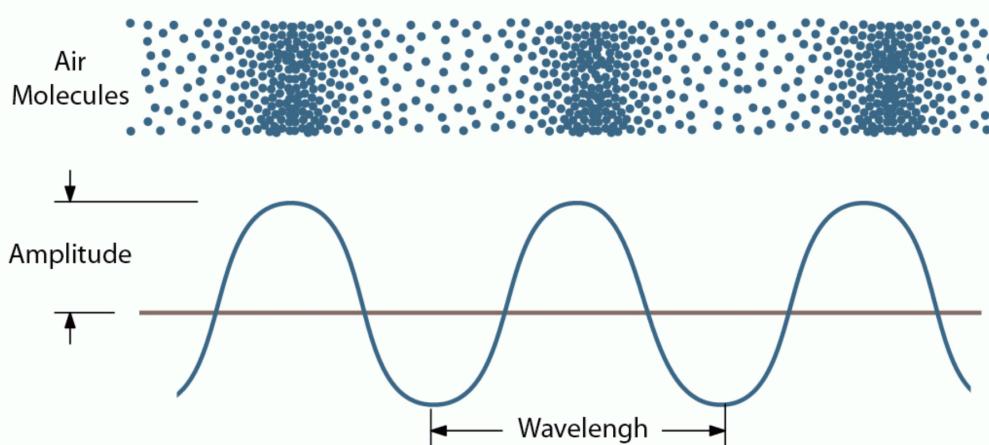


Figura 356 – Movimento das ondas sonoras no ar. Fonte: (KARAKOÇ; VAROL, 2017)

32.2.2 Pré-Processamento

O pré-processamento é aplicado ao sinal de áudio antes da extração de características, se necessário. A principal função dessa etapa é aprimorar certas características do sinal de entrada para maximizar o desempenho da análise de áudio nas fases posteriores do sistema de análise. Isso é obtido reduzindo os efeitos do ruído ou enfatizando os sons-alvo no sinal (VIRTANEN; PLUMBLEY; ELLIS, 2018a).

Se os dados de áudio forem coletados de várias fontes, é mais provável que sejam capturados em configurações de gravação não uniformes, com variações na quantidade de canais de áudio, taxa de amostragem e quantização de bits. Essas variações podem ser tratadas convertendo o sinal de áudio em um formato uniforme, mixando-o em um número fixo de canais e fazendo uma nova amostragem em uma mesma taxa de amostragem fixa para todas as amostras, bem como também a conversão da quantização de bits para que todos se encontrem nas mesmas condições para o estudo.

Para serem usados em CNNs, também é necessário aplicar outros tipos de pré-processamento nos áudios, visto que em sua forma bruta, conforme apresentado na Figura 357, é apenas um vetor representado o domínio do tempo. Uma das técnicas para adequação dos áudios para este propósito é a modificação de seu domínio de representação, podendo ser alterado para o domínio da frequência ou para o domínio do tempo-frequência.

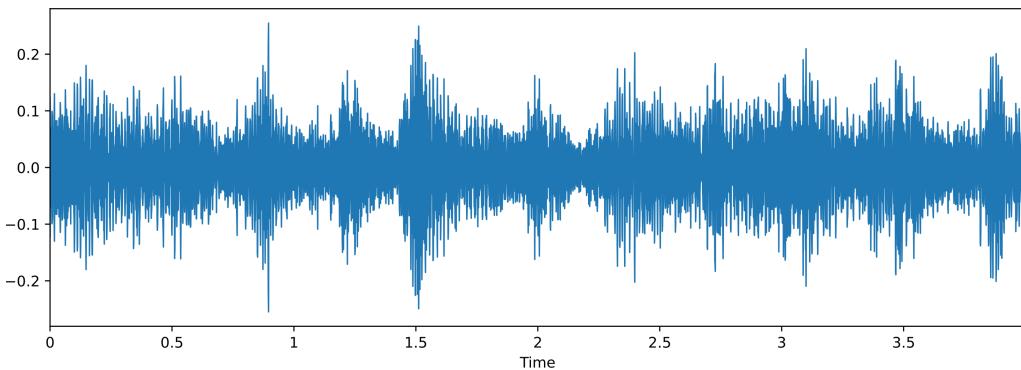


Figura 357 – Representação visual do áudio em sua forma bruta no domínio do tempo. Fonte: Autoria própria

Para criar uma representação no domínio do tempo-frequência é necessário a aplicação das transformadas discretas de Fourier ou *Discrete-time Fourier Transform* do inglês (*DFT*), onde esta é uma variante numérica da Transformada de Fourier. Especificamente, dado um vetor de n amplitudes de entrada, como $\{f_0, f_1, f_2, \dots, f_{n-2}, f_{n-1}\}$, a *DFT* produz um conjunto de n magnitudes de frequência. A *DFT* é definida pela Equação 32.1:

$$X[f] = \sum_{n=0}^{N-1} x[n] e^{-i2\pi f n} \quad (32.1)$$

onde, f é usado para denotar o ordinal do domínio da frequência e n é usado para representar o ordinal do domínio do tempo. O grande “ N ” é o comprimento da sequência a ser transformada.

Através do uso da *DFT*, é possível chegar a uma representação chamada mel-espectrograma, que é uma representação do som onde a energia em cada banda de frequência é dada em função do tempo, conforme apresentado na Figura 358.

32.2.3 Deep Learning

O *Deep Learning* é uma classe de *Machine Learning* que apresenta um desempenho melhor em dados não estruturados. As técnicas de *Deep Learning* estão superando as técnicas atuais de *Machine Learning*, elas permitem que modelos computacionais aprendam características progressivamente a partir de dados em vários níveis. A popularidade do *Deep Learning* aumentou conforme a quantidade de dados disponíveis

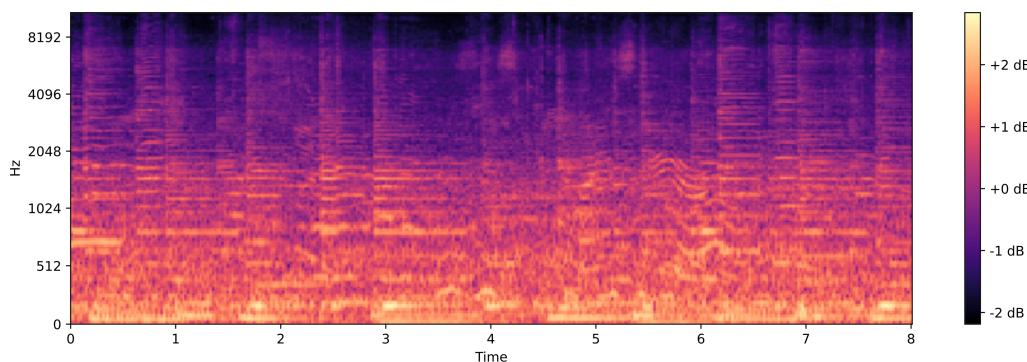


Figura 358 – Representação visual de um mel-espectrograma no domínio do tempo-frequência. Fonte: Autoria própria

aumentava, bem como o avanço do hardware que fornece computadores poderosos ([MATHEW; AMUDHA; SIVAKUMARI, 2020](#)).

Como visto em [Mathew, Amudha e Sivakumari \(2020\)](#), o algoritmo de *Deep Learning* passa os dados por várias camadas; cada camada é capaz de extrair as características progressivamente e passá-las para a próxima camada. As camadas iniciais extraem características de baixo nível e as camadas seguintes combinam os atributos para formar uma representação completa.

O *Deep Learning* busca identificar uma estrutura complexa em grandes conjuntos de dados usando o algoritmo de *backpropagation* para indicar como uma rede deve alterar seus parâmetros internos para diminuir a taxa de erro durante o aprendizado. As *CNNs* são algoritmos de *Deep Learning* que trouxeram avanços no processamento de imagens, vídeo, fala e áudio, enquanto as redes recorrentes se destacaram em dados sequenciais, como texto e fala ([LECUN; BENGIO; HINTON, 2015](#)).

As *CNNs* são uma variação da rede neural artificial perceptron de múltiplas camadas, baseada no processamento de dados visuais, capaz de aplicar filtros nesses dados, mantendo a relação de vizinhança entre os pixels da imagem ao longo do processamento da rede ([LECUN et al., 1998](#)). O nome “Convolutional Neural Network” indica que a rede emprega uma operação matemática chamada convolução.

Após uma camada convolucional, geralmente existe uma camada de *pooling*. O objetivo dessa camada é reduzir progressivamente a dimensionalidade das imagens de entrada, diminuindo o custo computacional da rede e evitando o *overfitting*. A saída das camadas convolucionais e de *pooling* representam as características extraídas da imagem de entrada. O objetivo das camadas totalmente conectadas é utilizar essas características para classificar a imagem em uma classe previamente especificada ([ARAÚJO et al., 2017b](#)).

32.2.3.1 LeNet

Um dos primeiros projetos de *CNNs* foi a *LeNet*, proposta por [LeCun et al. \(1998\)](#), que contribuiu para impulsionar a área de *Deep Learning*. Inicialmente, essa arquitetura foi utilizada para reconhecimento de caracteres, tais como código postal e dígitos numéricos. Novas arquiteturas foram propostas nos últimos anos como forma de melhoria da *LeNet*, embora as versões de *CNNs* melhoradas compartilhem os conceitos fundamentais.

A *LeNet* consiste em duas camadas convolucionais, com 6 e 16 filtros de tamanhos 5x5, com passo 1 e função *ReLU* (*Rectified Linear Units*). Duas camadas de *max pooling* imediatamente após as duas camadas convolucionais, com tamanho 2x2 e passo 2. Após as camadas convolucionais e *pooling*, há duas camadas totalmente conectadas com 120 e 84 neurônios. A última camada é a função *softmax*, que calcula a saída da rede. A Figura 359 ilustra a representação da arquitetura básica da *LeNet*.

Novas arquiteturas de *CNNs* que surgiram na última década são basicamente formadas pelas mesmas camadas presentes na *LeNet*, mas variam em relação ao propósito da classificação, à quantidade de camadas convolucionais, entre outros fatores. Em meio a diversidade dessas arquiteturas, este projeto explorou a arquitetura *LeNet*, propondo algumas modificações para a adequação ao método proposto.

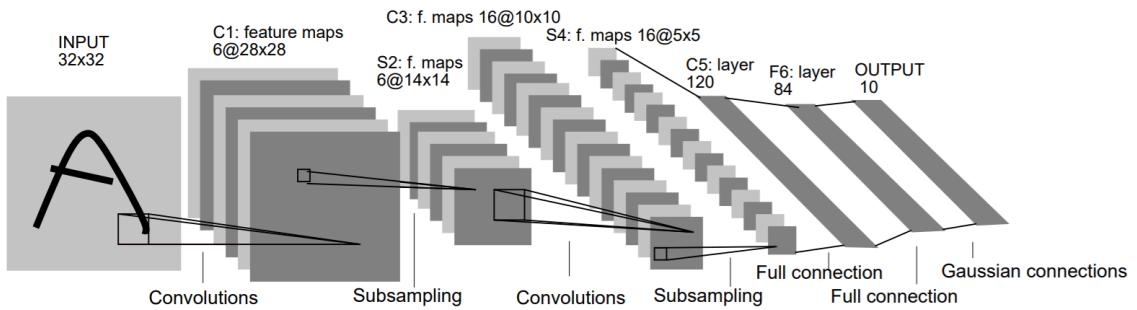


Figura 359 – Arquitetura de LeNet-5, uma rede neural convolucional, aqui para reconhecimento de dígitos. Cada plano é um mapa de características, ou seja, um conjunto de unidades cujos pesos são restritos a serem idênticos. Fonte: ([LECUN et al., 1998](#))

32.2.4 Seleção de Características

A fim de encontrar inconsistência ou redundância nas características obtidas na etapa de extração, foi aplicado uma seleção de características sobre o descriptor gerado. Isso é feito pois a qualidade das características pode afetar no desempenho da classificação. Além disso, ao se identificar essas características, é possível reduzir a dimensionalidade do descriptor, tornando o treinamento uma tarefa mais rápida e menos custosa computacionalmente.

O objetivo desta etapa é identificar quais características poderiam contribuir melhor para a discriminação dos sons urbanos. Para isso, foi utilizado o algoritmo de seleção de atributos *Information Gain* ou Ganho de Informação que funciona da seguinte forma: os valores de entrada variam de 0 (sem informação) a 1 (informação máxima), as características que contribuem com mais informações terão um maior valor de ganho, enquanto aquelas que não adicionam muita informação terão uma pontuação mais baixa ([KAUR, 2019](#)). Considerando um conjunto de dados S com s amostras pertencentes a m classes distintas, a Equação 32.2 apresenta o *information* ([VOGADO et al., 2018](#)) utilizado para classificar uma dada amostra.

$$Info(S) = - \sum_{i=1}^m p_i \log_2(p_i), \quad (32.2)$$

onde p_i é a probabilidade de uma amostra pertencer a uma classe C_i . A entropia de um determinado atributo A que tem valores v é mostrada na Equação 32.3:

$$Entropy(A) = - \sum_{i=1}^m Info(S) \frac{s_{1i} + s_{2i} + \dots + s_{mi}}{s}, \quad (32.3)$$

e a variável s_{ij} representa o número de amostras pertencentes à classe C_i do subconjunto S_i . O atributo gain A é representado pela Equação 32.4:

$$Gain(A) = Info(S) - Entropy(A). \quad (32.4)$$

32.2.5 Classificação

No processo de classificação, os valores obtidos na etapa de extração de características são os insumos para que uma técnica de *Machine Learning* possa então discernir entre possíveis padrões contidos em um grupo de atributos ([FILHO et al., 2016a](#)). Segundo [Looney \(1997\)](#), padrão é tudo aquilo para o qual existe uma entidade nomeável representante, entendido a partir de alguma forma de conhecimento.

Neste trabalho, foram aplicados os seguintes algoritmos para classificar os atributos resultantes da etapa de extração de características: *Random Forest (RF)* ([BREIMAN, 2001](#)) e *Support Vector Machine (SVM)*

(CORTES; VAPNIK, 1995b). Tais classificadores foram selecionados considerando sua ampla adoção na literatura relacionada (CARVALHO et al., 2020; CLARO et al., 2019; ARAÚJO; SANTANA; NETO, 2016).

O RF é uma combinação de preditores de árvores de decisão, de modo que cada árvore depende dos valores de um vetor aleatório amostrado de forma independente e com a mesma distribuição para todas as árvores na “floresta”. Cada árvore dá um voto indicando sua decisão sobre a classe do objeto. Assim, a classe com o maior número de votos é escolhida para o objeto (MA et al., 2017).

O SVM é uma técnica que consiste no mapeamento de vetores de entrada em um espaço amostral com dimensão superior, por meio de construção do hiper-plano máximo de separação. Para isso, dois hiper-planos auxiliares são construídos de uma forma que o hiper-plano de separação resultante maximize a distância entre eles. É assumido que, quanto maior for a margem de distância entre os planos, menor é o erro de generalização do classificador (DUDA; HART et al., 1973). O SVM apresenta-se como uma alternativa para conjuntos de dados com poucas amostras disponíveis para treinamento.

32.2.6 Métricas de Validação

Após treinados, os classificadores geram previsões para os dados de teste. Essas previsões precisam ser avaliadas para que seja possível uma análise dos resultados, a fim de mensurar o desempenho do método de classificação. Neste trabalho, foram selecionadas três métricas de validação amplamente adotadas na literatura para avaliar modelos de classificação: acurácia, índice *kappa* e AUC. Essas métricas fazem uso da matriz de confusão, que indica a classificação correta ou incorreta das classes em uso, agrupando os resultados em quatro classes, sendo elas: Falso Negativo (FN), Falso Positivo (FP), Verdadeiro Positivo (VP) e Verdadeiro Negativo (VN).

A métrica acurácia (*Acc*) (BARATLOO et al., 2015a) pode ser descrita como:

$$Acc = \frac{VP + VN}{VP + VN + FP + FN}, \quad (32.5)$$

O *kappa* (COHEN, 1960) indica como os classificadores selecionados superaram o classificador que simplesmente adivinha aleatoriamente de acordo com a frequência de cada classe. De acordo com (VOGADO et al., 2018), há uma categorização para o índice *kappa*, apresentado na Tabela 222. A fórmula do *kappa* é definida como:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \quad (32.6)$$

onde p_o é o resultado observado, e p_e é o esperado.

Em adição, também foi adotado como métrica a área sob a curva ROC (AUC) (HANLEY; MCNEIL, 1982). A melhor performance é atingida quando o valor de AUC se aproxima de 1 (equivalente to 100%).

Tabela 222 – Categorização do desempenho da classificação, de acordo com o índice *kappa*.

<i>Kappa</i>	Qualidade
$K < 0.2$	Ruim
$0.2 \leq K < 0.4$	Razoável
$0.4 \leq K < 0.6$	Bom
$0.6 \leq K < 0.8$	Muito bom
$K \geq 0.8$	Excelente

32.3 Trabalhos Relacionados

Modelos de aprendizado profundo têm sido aplicados com sucesso para reconhecimento de som e superam os classificadores convencionais como o *Support Vector Machine* (SVM) ou o *Gaussian Mixture Model* (GMM) (LI et al., 2017b). A literatura dispõe de estudos recentes sobre reconhecimento de sons acústicos usando CNNs.

Trabalhos recentes aplicaram CNNs com dados brutos de formas de onda para definir um descritor para sons urbanos (DAI et al., 2017). Abdoli, Cardinal e Koerich (2019) propuseram uma CNN 1D ponta a

ponta para classificação de sons ambientais, resultando em um modelo compacto com poucos parâmetros, reduzindo o custo de computação, que é adequado para aplicações de reconhecimento de som móvel. Os autores inicializaram a primeira camada convolucional não treinável como um banco de filtros *Gammatone* para obter uma compensação entre atributos manuais e aprendizagem de representação. O modelo alcançou 72% de acurácia no conjunto de dados UrbanSound8K com uma divisão de treino e teste de 80/20.

Medhat, Chesmore e Robinson (2020) introduziram a Rede Neural Condisional Mascarada (*MCLNN*) para explorar as propriedades específicas do mel-espectrograma para detecção de som. Por exemplo, duas classes de som diferentes podem ter a mesma amplitude de energia, embora a posição da amplitude em um compartimento de frequência específico possa diferenciá-los. Assim, o *MCLNN* foi projetado para preservar a localização espacial, considerando o posicionamento energético durante o treinamento. Eles dividiram o sinal de áudio em vários quadros e empregaram um aumento de dados com base em diferentes taxas de sobreposição de quadros. Os autores realizaram uma avaliação de modelo extensa com base em conjuntos de dados para classificação de gênero musical e reconhecimento de som ambiental, incluindo os conjuntos de dados UrbanSound8K e ESC-50. O modelo atingiu 74,22% e 66,60% de acurácia de classificação para UrbanSound8K e ESC-50, respectivamente, com uma divisão de treino e teste de 90/10.

O trabalho de Cramer et al. (2019) investigou um solução de *transfer learning* baseada no modelo de *deep audio embedding L³-Net (Look, Listen e Learn)*. Não havia exigido dados anotados, embora um grande conjunto de dados fosse indispensável. Menos de 40 milhões de amostras de treinamento implicam em um *embedding* sub-ótimo. Consequentemente, os autores adotaram um procedimento de aumento onde novas amostras de áudio são geradas alterando aleatoriamente o volume para até 10%. O modelo atingiu 79,37% de acurácia no conjunto de dados UrbanSound8k e 79,82% de acurácia no conjunto de dados ESC-50.

Park e Yoo (2020) apresentaram um modelo de arquitetura composto de uma camada convolucional 1D (*LGTFB*) para extrair uma representação espectro-temporal da forma de onda de áudio, camada de normalização de intensidade igual (*EN*) para controlar a compensação entre equalização de energia e preservação da estrutura da representação de entrada, e uma *CNN 2D*. Eles empregaram aumento de dados de áudio usando alongamento de tempo, deslocamento de tom e deformações de compressão de faixa dinâmica. A proposta obteve uma acurácia de classificação de 85,8% e 88,1% nos conjuntos de dados UrbanSound8K e ESC-50, usando *cross-validation* de 10 e 5 *folds*, respectivamente.

Acurácia de classificação de eventos sonoros possui uma relação robusta com a extração de atributos (DEMIR; ABDULLAH; SENGUR, 2020). O trabalho de Mushtaq e Su (2020) propôs duas arquiteturas *CNN* com *max-pooling* ou sem *max-pooling* para classificar sons ambientais. Eles consideram os mel-espectrogramas, MFCC, log-mel espectrogramas separadamente como a entrada dos modelos *CNN*. Além disso, os autores adotam diferentes formas de deformação de áudio para aumento de dados, como mudança de tom, corte de silêncio, alongamento de tempo e adição de ruído branco. O modelo sem uma função de *max-pooling* com 3,1 milhões de parâmetros usando o log-mel espectrograma atingiu as maiores acuráncias: 94,94%, 89,28% e 95,37% para ESC-10, ESC-50 e UrbanSound8K, respectivamente.

Além disso, alguns modelos mais profundos requerem um grande número de parâmetros treináveis. Eles oferecem grande capacidade de discriminação, mas são mais suscetíveis a *overfitting* (ARAUJO et al., 2018). O estudo de Demir, Abdullah e Sengur (2020) introduziu um novo modelo *CNN* para extrair características profundas das duas camadas totalmente conectadas com 500 e 450 de dimensão, respectivamente, usando mel-espectrograma como uma representação de entrada para descrever os sons ambientais. O modelo continha cerca de 12,2 milhões de parâmetros treináveis. O desempenho de discriminação de características profundas foi avaliada por meio do classificador de conjuntos *K Nearest Neighbor* (*KNN*). O descritor proposto alcançou 86,70% de acurácia de classificação no conjunto de dados UrbanSound8K com um *cross-validation* de 10 *folds*.

As *CNNs* possuem uma alta capacidade de representação de conhecimento devido aos seus milhões de parâmetros. Analisando os trabalhos descritos, percebe-se que os modelos investigados possuem uma quantidade consideravelmente grande de parâmetros. Contudo, a expansão do número de parâmetros treináveis acarreta no aumento do custo computacional dos modelos. Em contrapartida, o método proposto neste trabalho propõe a utilização de uma arquitetura compacta de *CNN*, de modo a utilizar menos parâmetros do que as abordagens relacionadas e, ainda assim, proporcionar um bom desempenho na representação dos sinais acústicos.

Tabela 223 – Resumo dos trabalhos relacionados.

Model	Objetivo	Representação do Som	Bases	Amostras	Resultados
MCLNN (MEDHAT; CHESMORE; ROBINSON, 2020)	Classificação com MCLNN	Mel-espectrograma	UrbanSound8K ESC-50	8732 2000	Acurácia: 74,22% Acurácia: 66,60%
MelFB+LGTFB-EN-CNN (PARK; YOO, 2020)	Classificação com aprendizado profundo	Espectro-temporal da forma de onda de áudio	UrbanSound8K ESC-50	8732 2000	Acurácia: 85,8% Acurácia: 88,1%
ID CNN Gamma (ABDOLI; CARDINAL; KOERICH, 2019)	CNN 1D para classificação de sons ambientais	Áudio bruto (Forma de onda)	UrbanSound8K	8732	Acurácia: 72%
L^3 -Net (CRAMER et al., 2019)	Classificação com aprendizado profundo	-	UrbanSound8K ESC-50	8732 2000	Acurácia: 79,37% Acurácia: 79,82%
New deep CNN model (DEMIR; ABDULLAH; SENGUR, 2020)	Extração com CNN e Classificação com KNN	Mel-espectrograma	UrbanSound8K	8732	Acurácia: 86,70%
DCNN without maxpooling (MUSHTAQ; SU, 2020)	Classificação com duas CNNs	Mel-espectrograma MFCC e Log-mel espectrograma	UrbanSound8K ESC-50 ESC-10	8732 2000 400	Acurácia: 95,37% Acurácia: 89,28% Acurácia: 94,94%

32.4 Proposta

Este trabalho propõe descritores baseados em *CNN* para extrair características representativas de eventos sonoros urbanos a fim de classificá-las através de modelos amplamente utilizados na literatura. Ainda, inspirados por uma lacuna identificada na literatura, este trabalho propõe investigar se as características extraídas pela *CNN* são susceptíveis a redução de dimensionalidade através da seleção de características.

32.4.1 Materiais e Métodos

Para alcançar os descritores acústicos, a metodologia adotada comprehende seis etapas com suas técnicas e ferramentas específicas, conforme ilustrado na Figura 360. A primeira etapa consiste na aquisição de dados dos *datasets* públicos UrbanSound8K ([SALAMON; JACOBY; BELLO, 2014](#)) e ESC (urban noises) ([PICZAK, 2015b](#)), seguido pela segunda etapa chamada pré-processamento, onde os parâmetros para gerar a representação tempo-frequência dos áudios foram apresentados. Além disso, esta etapa apresenta as transformações realizadas para padronizar as propriedades das diferentes amostras de áudio. Na terceira etapa, aprendizado de características, uma rede neural profunda foi treinada para extrair padrões de áudio de relações espectrais e temporais em sinais de áudio a partir de suas camadas totalmente conectadas. Na quarta etapa, seleção de características, foi aplicado um algoritmo de seleção de características para definir um ranque de significância com as características extraídas, objetivando a redução da dimensionalidade do descriptor. A quinta etapa consistiu no treinamento de classificadores para distinguir entre classes contextuais (por exemplo, sirene, trem, avião, perfuração, entre outros.), seguido pela etapa final, onde os resultados dos classificadores foram validados usando diferentes métricas de avaliação.

32.4.1.1 Aquisição de Áudios

Foram selecionados dois dos *datasets* de som urbano *open-source* mais populares usados para reconhecimento de som: UrbanSound8K e ESC-50. O ESC-50 pré-configura 50 classes de sons; no entanto, apenas 10 classes são diretamente relacionadas ao urbano. Assim, adotamos o subset *exterior/urban noise* neste trabalho. A seguir, está uma breve descrição sobre os *datasets* utilizados:

- UrbanSound8K ([SALAMON; JACOBY; BELLO, 2014](#)): Este *dataset* possui 8732 áudios compostos exclusivamente de sons urbanos com uma duração de 4 segundos ou menos, rotulados e agrupados em dez classes diferentes. O UrbanSound8K é um *dataset* heterogêneo, pois a taxa de amostragem, a codificação e o número de canais podem variar entre as amostras. Além disso, é um conjunto de dados não balanceado; por exemplo, a classe *Gun shot* tem quase um terço das amostras presentes na classe *Street music*, conforme apresentado na Tabela 224.
- ESC (urban noise) ([PICZAK, 2015b](#)): Este subconjunto possui 400 amostras de sons distribuídas em 10 classes (veja a Tabela 224). É um *dataset* homogêneo, onde todas as amostras têm duração fixa de 5 segundos e apenas um canal. ESC (urban noises) é uma base balanceada, conforme mostrado na Tabela 224. No entanto, observa-se um número reduzido de amostras por categoria, o que pode impactar negativamente no desempenho dos classificadores.



Figura 360 – Passo a passo para extração das características e avaliação dos descritores na caracterização de eventos sonoros.

32.4.1.2 Pré-processamento

Foi adotado o mel-espectrograma como uma representação de tempo-frequência. Para tanto, as amostras de áudio foram divididas em janelas menores (2048). Após o janelamento do sinal, as transformadas discretas de Fourier (*Discrete Wavelet Transform (DFT)*) sobre as janelas sobrepostas foram calculadas para atingir a representação de tempo-frequência, chamada de espectrograma. Para isso, utilizou-se a função de janelas *Blackman-Harris*. Além disso, adotou-se uma sobreposição de 50% das janelas (comprimento do salto) para garantir a dependência estatística entre elas.

Foram re-amostrados os dois conjuntos de dados para 44,100 Hz, cobrindo a faixa de frequência audível (0-22050 Hz). O dataset UrbanSound8K tem sua duração de amostras entre 0 e 4 segundos, com a concentração mais alta em 4 segundos. A técnica de preenchimento de zeros foi empregada para inserir zeros na matriz do mel-espectrograma a fim de preservar a uniformidade da dimensão das entradas da CNN ([ZHOU](#);

Tabela 224 – Distribuição das categorias de sons por *dataset*. As linhas em negrito indicam as classes em comum para ambos os *datasets*.

UrbanSound8K			ESC (urban noises)		
Categorias	Número de amostras	Duração total (s)	Categorias	Número de amostras	Duração total (s)
Air conditioner	1000	3994.9287	Helicopter	40	200
Car horn	429	1053.9533	Car horn	40	200
Children playing	1000	3961.8745	Chainsaw	40	200
Dog bark	1000	3148.7496	Fireworks	40	200
Drilling	1000	3548.244	Engine	40	200
Engine idling	1000	3935.9925	Train	40	200
Gun shot	374	616.7965	Church bells	40	200
Jackhammer	1000	3610.9747	Airplane	40	200
Siren	929	3632.7016	Siren	40	200
Street music	1000	4000.0	Hand saw	40	200

(SONG; SHU, 2017). Desta forma, fixamos os mel-espectrogramas com 173 bandas usando um tamanho de janela de 23 ms (2048 amostras), com um tamanho de salto correspondente à metade do tamanho da janela, resultando em dimensões do mel-espectrograma de 173 x 173 para o *dataset* UrbanSound8K. Foram utilizados os mesmos valores de parâmetro para gerar os mel-espectrogramas da base ESC (urban noise), além de suas amostras terem uma duração fixa de 5 segundos. Assim, foram adotados mel-espectrogramas com dimensões de 216 x 216.

A biblioteca LibROSA (MCFEE et al., 2015) foi utilizada para gerar os mel-espectrogramas usados como representação de entrada para a CNN. Esta biblioteca normaliza os dados na faixa [-1,1] por padrão e transforma os sinais em mono fazendo a média das amostras entre os canais. A tabela 225 mostra os parâmetros usados para gerar os espectrogramas para ambos os *datasets*.

Tabela 225 – Parâmetros de representação dos mel-espectrogramas.

Parâmetros	Valores
Taxa de amostragem	44.100 Hz
Tamanho da janela	2048
Comprimento do salto	1024
Função de janelamento	Blackman-Harris

32.4.1.3 Aprendizado de Características

Neste trabalho, foi proposta uma arquitetura CNN baseada em LeNet (LECUN et al., 1998), que consiste em quatro camadas convolucionais, com 24, 48, 48 e 48 filtros de tamanhos 5x5, com passo 1, zero padding, batch normalization e função ReLU (Rectified Linear Units). Foram utilizadas quatro camadas de max pooling imediatamente após as quatro camadas convolucionais, com tamanho 2x2 e passo 2. Após as camadas convolucionais e pooling, há duas camadas totalmente conectadas com 192 e 64 neurônios, respectivamente, e uma dropout de 0,5 entre elas. A última camada é a função softmax, que calcula a saída da rede. Os parâmetros de treinamento foram: taxa de aprendizagem = 0,001, tamanho do lote = 128 e número de épocas = 50. A Tabela 226 apresenta o sumário da arquitetura proposta.

A arquitetura proposta tem aproximadamente 1,08 milhões de parâmetros treináveis para UrbanSound8K e 1,48 milhões para o ESC (urban noises), representando uma arquitetura compacta em comparação com outras abordagens na literatura. Embora o UrbanSound8K e o ESC possuam tamanhos de dimensão de espectrogramas diferentes, os mapas de características gerados como a entrada das camadas total-

Tabela 226 – Parâmetros da arquitetura CNN proposta. O tamanho da camada de entrada tem a dimensão de 173 x 173 para o conjunto UrbanSound8K e 216 x 216 para o conjunto ESC (urban noises).

Hyperparâmetros	Valores correspondentes
Tamanho da camada de entrada	(173 x 173) e (216 x 216)
Número de camadas convolucionais	4
Número de filtros por camada	(24, 48, 48 e 48)
Tamanho dos kernels convolucionais	(5, 5)
Tamanho dos kernels de <i>pooling</i>	(2, 2)
Camadas totalmente conectadas	2 (192 e 64 neurônios)
<i>Softmax</i>	10 neurônios

mente conectadas têm dimensionalidade próxima. Isso ocorre porque o tamanho do spectrograma é reduzido pela metade após cada uma das quatro operações de *pooling*. Assim, os mapas de características do UrbanSound8K têm dimensionalidade 11 e os do ESC têm dimensionalidade 14.

32.4.1.4 Seleção de Características

Na etapa de seleção de características, foi utilizado o algoritmo *Information Gain (IG)* para a classificação do nível de relevância de cada característica extraída pela CNN. O objetivo é identificar características redundante e inconsistentes que podem afetar negativamente o desempenho da classificação. Uma vez que identificamos essas características, é possível reduzir a dimensionalidade do descritor, tornando o treinamento uma tarefa mais rápida e menos dispendiosa computacionalmente.

Visando identificar a influência das características extraídas da CNN, foram realizados testes agrupando as características em grupos de 5, de acordo com o nível de relevância de cada uma segundo o *IG*, incrementando em mais 5 características até a quantidade total de cada descritor, sendo 64 e 192 dimensões cada um dos descritores extraídos. Dessa forma os testes puderam fornecer a informação de qual a menor quantidade de características era necessário para atingir uma acurácia igual ou próxima a alcançada com o descritor completo.

Todos os teste realizados nesta etapa foram executados usando o software WEKA, uma suíte de algoritmos de mineração de dados e *Machine Learning*, desenvolvida pela University of Waikato, Nova Zelândia. Essa suíte contém ferramentas para pré-processamento de dados, seleção e classificação, sendo utilizado em diferentes etapas da metodologia (HALL et al., 2009).

32.4.1.5 Classificação

Na etapa de classificação, foram utilizados os algoritmos *RF* e *SVM*, como discorrido na Seção 34.2. Em cada classificação realizada, os dados foram divididos aleatoriamente em 80% para treino e 20% para testes, sendo que, uma vez divididos, os mesmos dados foram utilizados em todas as execuções. A proporção de dados foi selecionada com base em outros estudos da literatura (LIMA et al., 2020; ARAUJO et al., 2019). Em relação aos parâmetros de cada classificador, foram utilizados os valores padrão da ferramenta WEKA, conforme apresentados na Tabela 227.

32.4.1.6 Validação

Como mencionado na Seção 34.2, para avaliar os classificadores, foram utilizadas três métricas de validação comumente aplicadas na literatura: acurácia, índice *kappa* e *AUC*. As métricas escolhidas para realizar a validação dos resultados obtidos pelo presente trabalho são amplamente utilizadas por conseguirem demonstrar de forma clara a proporção de erros e acertos obtidos pelos classificadores. As três métricas estão disponíveis na avaliação dos classificadores pela ferramenta WEKA, sendo assim, os valores foram obtidos através dela.

Tabela 227 – Parâmetros principais usados nos classificadores.

Classificador	Parâmetros Usados
RF	$numIteration$ (<i>number of trees</i>) = 100, $numFeatures$ = 0, $numDecimalPlaces$ = 2, $numExecutionSlots$ = 1, $maxDepth$ = 0 (<i>unlimited</i>), $seed$ = 1
SVM	$coef0$ = 0.0, $cost$ = 1.0, $degree$ = 3, eps = 0.001, $gamma$ = 0.0, $kernelType$ = radial basis function: $\exp(-gamma * u-v ^2)$, $loss$ = 0.1, $normalize$ = false

32.5 Resultados e Discussões

Esta seção apresenta e discute os resultados alcançados a partir da execução da metodologia proposta neste trabalho para a classificação de sons urbanos, como apontado na Seção 32.4. Além disso, também é feita uma comparação do método proposto com os resultados dos estudos relacionados.

32.5.0.1 Classificação sem Seleção de Características

A Tabela 228 apresenta o desempenho dos classificadores com os descritores da CNN de tamanho 64 e 192, sendo que os números entre parênteses representam o tamanho do descritor e os números em negrito são os melhores resultados. Esta tabela mostra que aumentar o número de características extraídas pela CNN de 64 para 192 aumentou o desempenho da classificação, exceto para ESC (urban noises) com SVM. Uma possível justificativa é que o classificador SVM se adapta melhor a pequenos conjuntos de dados, se comparado ao RF.

A Tabela 228 pode indicar que optar por descritores mais compactos pode implicar na perda de informação que se reflete negativamente na classificação. Na UrbanSound8K, os descritores maiores trazem uma maior variabilidade de informação, tornando a descrição mais rica, melhorando a acurácia. Também é possível notar que os resultados alcançados com a base UrbanSound8K foram consideravelmente maiores que os da base ESC (urban noises). Isto pode ser explicado pela diferença entre a quantidade de amostras de cada base, o que pode ter prejudicado a CNN na extração de padrões dos espectrogramas, visto que CNNs se beneficiam melhor com grandes quantidades de dados para o treinamento. Ainda de acordo com a Tabela 228, é possível notar que o classificador RF possui melhores resultados que o SVM para os dois descritores extraídos da base UrbanSound8K. No entanto, para a base ESC (urban noises) o RF é melhor apenas no caso do descritor de 192 dimensões, perdendo para o SVM no descritor de 64 dimensões.

32.5.0.2 Classificação com Seleção de Características

Nesta seção realizamos um experimento de classificação associado a seleção de características a fim de investigar se as características extraídas pela CNN são susceptíveis a redução de dimensionalidade. Através da revisão da literatura, identificamos que a seleção de características é aplicada a descritores manuais, no entanto, nenhum trabalho aplicou a seleção em características extraídas pela CNN. A Tabela 229 indica que a seleção de características promoveu um ganho na acurácia de classificação para ambos os classificadores e base de dados. E, em termos de Kappa e AUC, o desempenho se manteve ou melhorou na maioria significativa dos cenários.

Em todos os cenários apresentados na Tabela 229, a redução da dimensionalidade é observada. Na base de dados UrbanSound8K, foi alcançada uma redução de 70% no descritor de dimensionalidade 64 e 45% de redução para o descritor de dimensionalidade 192. Já na base ESC (urban noises), foi alcançada

Tabela 228 – Resultados da classificação dos descritores para UrbanSound8K e ESC (urban noises), considerando as métricas acurácia, kappa e AUC.

Datasets	Classificadores	Métricas	Características Profundas (64)	Características Profundas (192)
Urban Sound8K	RF	Acurácia	0.952	0.956
		Kappa	0.946	0.9512
		AUC	0.993	0.997
	SVM	Acurácia	0.951	0.954
		Kappa	0.945	0.949
		AUC	0.973	0.974
ESC (urban noises)	RF	Acurácia	0.787	0.812
		Kappa	0.759	0.788
		AUC	0.956	0.975
	SVM	Acurácia	0.812	0.787
		Kappa	0.788	0.761
		AUC	0.896	0.883

uma redução de 25% para o descritor de dimensão 64 e 45% para o descritor de dimensionalidade 192. Nós observamos uma maior redução para a base de dados UrbanSound8K indicando que os descritores para esse conjunto podem apresentar um maior número de características redundantes e com presença de ruído. Nós identificamos que mesmo as características extraídas pela CNN estão sujeitas a seleção de características, isso é especialmente útil quando abordamos aplicações embarcadas onde a obtenção de um descritor compacto é pertinente por limitação computacional ou de bateria.

Tabela 229 – Resultados da classificação dos descritores propostos após a etapa de seleção de características para os conjuntos de dados UrbanSound8K e ESC (urban noises). Os números entre parênteses nos resultados de precisão indicam o tamanho do descritor após a seleção e os números entre parênteses no título dos descritores indicam seu tamanho original. Os números em negrito significam manutenção ou melhoria do desempenho de classificação associada a uma redução do descritor.

Datasets	Classificadores	Métricas	Características Profundas (64)	Características Profundas (192)
Urban Sound	RF	Accuracy	0.955 (52)	0.958 (105)
		Kappa	0.950	0.953
		AUC	0.992	0.997
	SVM	Accuracy	0.953 (19)	0.955 (135)
		Kappa	0.947	0.950
		AUC	0.974	0.975
ESC (urban noises)	RF	Accuracy	0.800 (60)	0.837 (125)
		Kappa	0.774	0.817
		AUC	0.956	0.965
	SVM	Accuracy	0.837 (45)	0.787 (105)
		Kappa	0.817	0.760
		AUC	0.910	0.881

32.5.0.3 Comparação com a Literatura

Na Tabela 230, nós realizamos uma comparação dos resultados obtidos pelo método aqui proposto com os resultados dos estudos relacionados, apresentados na Seção 32.3. A Tabela mostra a acurácia e o número de parâmetros do modelo para diferentes abordagens nos conjuntos de dados UrbanSound8K e ESC (urban noises). É importante ressaltar que alguns trabalhos relacionados relatam os valores de acurácia da base ESC para todas as suas classes ou um subconjunto de dez classes (não relacionado especificamente aos

sons urbanos). As linhas em negrito indicam os resultados dos descritores propostos neste trabalho. O M significa milhão e k significa milhares.

Conforme observado na Tabela 230, os descritores propostos neste trabalho obtiveram a maior acurácia para a base de dados UrbanSound8K e a terceira maior acurácia para a base ESC (urban noises). Nesta última, existe um desafio na comparação pois nosso trabalho adotou um subconjunto da base contendo dez classes referentes a sons urbanos. Os trabalhos relacionados usaram todo o conjunto de dados com 50 classes (ESC-50) ou um subconjunto com dez classes (ESC-10), conforme relatado na Tabela 230. Além disso, nenhum deles relatou resultados para as métricas *kappa* e AUC. Vale ressaltar que o ESC-10 possui classes diferentes daquelas adotadas por neste trabalho. (MUSHTAQ; SU, 2020) obteve a maior acurácia de 89,28% para ESC-50, e 94,94% para ESC-10. O descritor proposto neste trabalho alcançou 81,2% de acurácia para a ESC (urban noises), utilizando 192 características.

No que toca o número de parâmetros, a Tabela 230 mostra que os modelos propostos possuem um número de parâmetros menor que todos os demais trabalhos, exceto pelo modelo proposto 1D CNN *Gamma* em (ABDOLI; CARDINAL; KOERICH, 2019) para a base UrbanSound8K e pelo modelo MCLNN proposto em (MEDHAT; CHESMORE; ROBINSON, 2020). Redes mais robustas (com um número maior de parâmetros) são mais suscetíveis a *overfitting* em conjuntos de dados menores (ARAUJO et al., 2018). Este trabalho conseguiu uma representação mais compacta comparando as dimensões 192 e 64 dos descritores com os trabalhos relacionados que também extraíram características das camadas totalmente conectadas da CNN, como Demir, Abdullah e Sengur (2020), com dimensões 500 e 450. Os resultados mostraram que os descritores deste trabalho superaram a maioria dos modelos relacionados e apresentaram desempenho significativamente próximo à maior acurácia alcançada na literatura para o conjunto de dados UrbanSound8K, sem aumento de dados.

Tabela 230 – Acurácia e o número de parâmetros para diferentes modelos nos conjuntos de dados UrbanSound8K e ESC.

Model	Objetivo	UrbanSound8K # de parâmetros	UrbanSound8K Acurácia	ESC(50/10) # de parâmetros	ESC(50/10) Acurácia
MCLNN (MEDHAT; CHESMORE; ROBINSON, 2020)	Classificação com MCLNN	1.2 M	74.22%	1 M/3 M	66.60% 85.5%
MelFB+LGTFB-EN-CNN (PARK; YOO, 2020)	Classificação com <i>Deep Learning</i>	3.1 M	85.8%	3.1 M/-	88.1%
1D CNN Gamma (ABDOLI; CARDINAL; KOERICH, 2019)	CNN 1D para classificação de sons ambientais	550 k	89%	-/-	-/-
<i>L</i> ³ -Net (CRAMER et al., 2019)	Classificação com <i>Deep Learning</i>	4.7 M	79.34%	4.7 M	79.82%/-
New deep CNN model (DEMIR; ABDULLAH; SENGUR, 2020)	Extração com CNN e Classificação com KNN	12.2 M	86.7%	-/-	-/-
DCNN without maxpooling (MUSHTAQ; SU, 2020)	Classificação com duas CNNs	3.1 M	95.37%	3.1 M/3.1 M	89.28%/94.94%
Características profundas (64) [proposto]	Extração com CNN e Classificação com RF e SVM	1.08 M	95.5%	1.48 M	-/83.7%
Características profundas (192) [proposto]	Extração com CNN e Classificação com RF e SVM	1.08 M	95.8%	1.48 M	-/83.7%

32.6 Conclusão

Este trabalho apresentou uma abordagem para a extração de características sonoras com base em um modelo de CNN compacto e de poucos parâmetros, que apresentou um desempenho superior as arquiteturas de CNN para classificação de som ambiental propostas nos trabalhos relacionados. O melhor resultado alcançado pelo método proposto no conjunto UrbanSound8K foi uma acurácia de 95,8%, AUC de 0.99 *Kappa* de 0.95 e uma dimensão de 105, sendo o índice *Kappa* considerado excelente de acordo com a categorização de (COHEN, 1960). Já no conjunto Esc (urban noises), o melhor resultado obtido foi uma acurácia de 83,7%, AUC de 0.91, *Kappa* de 0.81 e um vetor com dimensão 45, com um *Kappa* também considerado excelente segundo a categorização de (COHEN, 1960).

Em relação a análise dos resultados obtidos e os resultados dos trabalhos relacionados, é possível notar que, em relação ao conjunto de dados UrbanSound8K, os descritores propostos alcançaram a maior acurácia. Enquanto o conjunto ESC (urban noises) apresentou uma acurácia inferior ao encontrado na literatura. No entanto, o modelo possui um número significativamente menor de parâmetros em relação ao trabalho com

maior acurácia. Além disso, o subconjunto adotado neste trabalho é restrito aos sons urbanos e diferente dos demais adotados na literatura.

A etapa de seleção de características conseguiu de fato identificar características redundantes ou informações com ruído, fazendo assim com que essa características fossem eliminadas, o que leva a uma redução da dimensionalidade do vetor de características, como no caso do UrbanSound8K com o descriptor de dimensão 64, utilizando o classificador SVM, que teve uma redução de 70%. A eliminação de características redundantes ou informações de ruído também leva a um melhor desempenho na performance da classificação, como no caso da ESC (urban noises) com o classificador SVM, que passou de 81,2% para 83,7%. Em vista do que foi apresentado, É possível concluir então que a seleção de características permite uma análise eficiente das características relevantes, o que é especialmente útil para aplicações que exigem descritores com menor dimensionalidade. Ainda, vimos que há espaço para selecionar características extraídas das camadas totalmente conectadas da CNN, abordagem pouco explorada na literatura.

A principal limitação deste trabalho foi não aplicar técnicas de *data augmentation*, o que poderia ter melhorado o desempenho da classificação usando as características extraídas da CNN. Além disso, adotou-se apenas um modelo CNN não muito profundo para extrair características dos mel-espectrogramas. Em vista disso, destaca-se os seguintes trabalhos futuros:

- Realizar o agrupamento de sons da mesma categoria ou de categoria semelhante para aumentar o número de amostras por classe;
- Adotar diferentes técnicas de aumento de dados, como aplicação de máscaras de tempo e frequência, inserção de ruído de fundo e mudança de tom;
- Combinar diferentes espectrogramas, como por exemplo o espectrograma STFT e o espectrograma harmônico, para a entrada da CNN, objetivando uma descrição mais detalhada;
- Explorar outras arquiteturas de CNN clássicas e robustas, como a VGG, a ResNet e a NasNet, com estratégias de *fine-tuning* e *cross-validation*; e
- Investigar o uso do algoritmo PCA para a redução da dimensionalidade dos mapas de características.

Parte IV

Ferramentas Computacionais

GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos

Autores: Francisco das Chagas dos Anjos Carvalho Júnior; Romuere Rodrigues Veloso e Silva.

Resumo: A bioinformática consiste em aplicações de técnicas da informática na área da biologia. Com o crescente número de genomas sequenciados e publicados, teve-se a necessidade de criar bancos de dados específicos para seu armazenamento, além de ferramentas para a comparação e visualização de genomas. Embora existam inúmeras ferramentas de comparação e visualização de genomas, muitas destas apresentam informações incompletas na comparação de genes entre diferentes espécies, além de disponibilizarem poucos recursos de interação com usuários. Este trabalho tem como intuito desenvolver a ferramenta GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos, com objetivo de fornecer uma ferramenta com diversas funcionalidades, para o usuário utilizar de maneira interativa. Foi utilizado a linguagem de programação Python, juntamente com a biblioteca PyQt5, o *Basic Local Alignment Search Tool* (BLAST) e os algoritmos *Needleman-Wunsch* e *Smith-Waterman*. Para avaliar o desempenho de usabilidade do GeMapCom, foi utilizado o sistema de pontuação *System Usability Scale* (SUS), a ferramenta teve um desempenho médio de 88,3 pontos sendo “melhor que o imaginável” comprovada a partir da métrica de pontuação do SUS.

Keywords: Alinhamento, Bioinformática, Comparação, Genomas, SUS.

33.1 Introdução

A bioinformática consiste basicamente na aplicação de técnicas da informática nas áreas de estudo da biologia. Com isso, é possível a utilização e desenvolvimento de ferramentas computacionais para estudo, análise, interpretação e processamento de dados biológicos. A bioinformática teve seu surgimento no início dos anos 60, quando computadores emergiram como ferramentas importantes na biologia molecular ([ALVES, 2013](#)).

Segundo [Hagen \(2000\)](#), três fatores contribuíram para o surgimento da bioinformática. Primeiro, a expansão de sequências de aminoácidos forneceu uma fonte de dados que eram impossíveis de resolver sem o poder de processamento de números dos computadores. Segundo fator, a ideia de que as macromoléculas

carregam informações tornou-se uma parte central da estrutura conceitual da biologia molecular. Em terceiro, surgiram os computadores digitais de alta velocidade.

De acordo com Junior et al. (2002), o súbito número de genomas sequenciados e publicados acarretou na necessidade de análise a cerca das sequências geradas. Isso porque, a partir dessas análises, é possível delimitar mais precisamente as caracterizações funcionais dos organismos estudados e seus aspectos evolutivos.

Como essas sequências geradas produzem um grande volume de dados, criou-se bancos de dados específicos para seu armazenamento. A exemplo disso, pode-se *The National Center for Biotechnology Information* (NCBI) (NCBI, 2018) dos (EUA) sendo considerado o banco de dados central de informações genômicas. Integrado ao NCBI tem-se o *GenBank Sequence Database* (BENSON et al., 2013) tido como principal banco de dados por armazenar publicamente todas as sequências disponíveis de DNA, proteínas e RNA.

A partir desses conjuntos de dados é possível realizar a comparação genômica. Através dela é feita a comparação de dois organismos semelhantes a fim de identificar sequências biológicas e descobrir a sua função. Assim, o processo de comparação consiste em localizar as sequências do segundo indivíduo com base no primeiro (SEPÚLVEDA, 2019).

Nesse contexto, existem inúmeras ferramentas de comparação e visualização de genomas, muitas delas apresentam informações incompletas na comparação de genes entre diferentes espécies, além de disponibilizarem poucos recursos de interação com usuários. Partindo desse problema, surgiu a motivação para o desenvolvimento de uma ferramenta para comparação genômica e visualização de alinhamentos que facilite a interação com usuários da área de bioinformática.

33.1.1 Objetivo Geral

O presente trabalho tem como objetivo desenvolver uma ferramenta para comparação e visualização de dados genômicos denominada GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos, utilizando a linguagem de programação Python, juntamente com a biblioteca PyQt5, o *Basic Local Alignment Search Tool*¹ (BLAST) e os algoritmos *Needleman-Wunsch* (NEEDLEMAN; WUNSCH, 1970) e *Smith-Waterman* (SMITH; WATERMAN et al., 1981). Com objetivo de fornecer uma ferramenta com diversas funcionalidades, para o usuário utilizar de maneira interativa apresentando informações de forma clara e coesa.

33.1.2 Objetivos Específicos

1. Identificar as principais ferramentas utilizadas para visualização genômica;
2. Desenvolver uma ferramenta de visualização genômica;
3. Estabelecer um comparativo entre a ferramenta desenvolvida e as principais ferramentas já existentes acerca da visualização genômica;
4. Utilizar um método para avaliação de usabilidade da ferramenta proposta.

33.1.3 Organização do Trabalho

Este capítulo foi dividida em seis seções, organizados da seguinte maneira: Seção 33.1, apresenta a introdução, no Seção 33.2, são exibidos principais trabalhos relacionados com esta pesquisa tecnológica. No Seção 33.3 está apresentado o referencial teórico, onde são detalhados conceitos relevantes para este trabalho. No Seção 33.4, detalho-se como foi o desenvolvimento do GeMapCom. O Seção 33.5 apresentou-se a avaliação de usabilidade da ferramenta desenvolvida. O por último, no Seção 33.6 foram apresentadas as principais conclusões obtidas e trabalhos futuros.

¹ <https://blast.ncbi.nlm.nih.gov/Blast.cgi>

33.2 Trabalhos Relacionados

O trabalho de Mueller et al. (2008), consistiu em desenvolver um visualizador comparativo via Web² para o mapeamento de dados, mapas genéticos, físicos e citológicos. Utilizou a linguagem Perl orientada a objetos, com uma interface extensível para gravar adaptadores de dados para outros esquemas de bancos de dados relacionais e formatos de arquivo simples. É implementado em Perl orientado a objetos e banco de dados ou com outros sistemas de *backend*.

No trabalho de Youens-Clark et al. (2009) é uma ferramenta Web³ para exibir e comparar mapas genômicos. Utilizaram a linguagem Perl para a construção da aplicação intitulada CMap com esquema relacional de banco de dados. É uma aplicação feita na linguagem Perl que é executado em um servidor Web Apache nas variantes do Windows e do UNIX.

O trabalho de Batesole et al. (2014), consiste em uma ferramenta Web⁴ de código aberto que oferece um recurso de mapeamento comparativo dinâmico de acordo com entrada de sequências biológicas, entre dois ou mais conjuntos de mapas. Os dados são analisados e carregados no banco de dados relacional do MySQL para serem exibidos e comparados com os mapas de entrada das sequências biológicas ou outros dados públicos disponíveis no YouGenMap. Foi desenvolvido através de uma combinação e colaboração de várias bibliotecas JavaScript e bibliotecas PHP.

No trabalho de Overmars et al. (2015), foi desenvolvido uma ferramenta Web⁵ que cria mapas circulares personalizados para auxiliar na análise de genomas microbianos e elementos de sequência. Foi implementado como uma aplicação de página única, o *frontend* faz chamadas do lado do servidor através de Jquery e AJAX e recebe uma resposta do servidor. Utilizou a linguagem de programação Python, Javascript, SVG e banco de dados MySQL.

No trabalho de (HOLTZ; DAVID; RANWEZ, 2017) foi desenvolvido uma aplicação Web⁶ de comparação de mapas genéticos. No mesmo foi utilizado R shiny e Plotly para desenhar e comparar múltiplos mapas genéticos.

Na Tabela 231 é abordado um resumo dos trabalhos correlatos a ferramenta proposta.

Tabela 231 – Trabalhos Relacionados.

Trabalho	Objetivo	Metodologia	Ferramentas/ Técnicas
Mueller et al. (2008)	Visualizador comparativo Web.	Linguagem Perl com bancos de dados relacionais.	Perl e banco de dados orientado por clusters.
Youens-Clark et al. (2009)	Uma ferramenta Web para exibir e comparar mapas.	Linguagem Perl com bancos de dados relacionais.	Perl, MySQL, HTML e JavaScript.
Batesole et al. (2014)	Uma ferramenta Web para mapeamento comparativo dinâmico de mapas.	Os dados são analisados e carregados no banco de dados MySQL.	JavaScript, HTML, PHP e MySQL
Overmars et al. (2015)	Uma ferramenta Web que cria mapas circulares para análise de genomas.	Uma aplicação que realiza chamadas do lado do servidor.	Python, Javascript, MySQL e SVG.
Holtz, David e Ranwez (2017)	Uma aplicação Web para comparação de mapas.	R shiny e Plotly para desenhar e comparar mapas genéticos.	R shiny e Plotly

² <https://sgn.cornell.edu/>

³ <http://www.gmod.org/cmap>

⁴ <http://bioinfolab.miamioh.edu/yougenmap/index.php>

⁵ <http://civi.cmbi.ru.nl/>

⁶ <http://bioweb.supagro.inra.fr/geneticMapComparator/>

Diferente dos trabalhos citados nesta seção, o GeMapCom tem como objetivo ser uma ferramenta para desktop, onde o usuário poderá utilizar os algoritmos de Needleman-Wunsch, Smith-Waterman e BLAST para realizar alinhamentos, além de salvar os resultados gerados neste processo, em forma de imagem ou texto.

33.3 Referencial Teórico

Nesta seção são expostos assuntos fundamentais para a compreensão do presente trabalho, tais como métodos, técnicas e tecnologias. A Subseção 33.3.1 apresenta conceitos sobre a Genômica Comparativa. Na Subseção 33.3.2 é abordado os principais Banco de Dados Genômicos. A Subseção 33.3.3 trata-se de demonstrar os tipos de alinhamentos que existem. Nas Subseções 33.3.4, 33.3.5, e 33.3.6 são aboradados conceitos dos algoritmos de Needleman-Wunsch, Smith-Waterman e Basic Local Alignment Search Tool (BLAST), respectivamente. E na Subseção 33.3.7 será demonstrado o método System Usability Scale (SUS).

33.3.1 Genômica Comparativa

De acordo com [Bachhawat \(2006\)](#), a genômica comparativa é uma técnica que visa o estudo de como os dados genômicos, como a localização de cadeias e genes, as categorias e funções de diferentes espécies estão relacionadas entre si.

A partir da genômica comparativa, os pesquisadores podem identificar regiões de similaridade e diferença entre espécies. Com posse dessas informações, é possível melhorar o entendimento sobre a estrutura e a função dos genes comparados ([INSTITUTE, 2015](#)). Uma das aplicações da genômica comparativa são a descoberta de genes e de RNAs ainda não codificados.

Uma sequência completa do genoma de um organismo pode ser considerada o mapa genético final, no sentido de que as características hereditárias são codificadas dentro do DNA e que a ordem de todos os nucleotídeos ao longo de cada cromossomo é conhecida. No entanto, o conhecimento da sequência do DNA não nos diz diretamente como essa informação genética leva aos traços e comportamentos observáveis (fenótipos) que queremos entender ([HARDISON, 2003](#)).

É possível, para os cientistas, o entendimento sobre a aparência, o comportamento e a biologia dos seres vivos em geral, porém essas características podem variar ao longo do tempo. Portanto, o papel da genômica comparativa é fornecer uma ferramenta para o estudo da evolução, observando as relações evolutivas entre as espécies e as diferenças correspondentes em seu DNA ([INSTITUTE, 2015](#)).

A genômica comparativa tem amplas aplicações no campo da medicina molecular e evolução molecular. A aplicação mais significativa da genômica comparativa na medicina molecular é a identificação de alvos de drogas de muitas doenças infecciosas. Por exemplo, análises comparativas de genomas de fungos levaram à identificação de muitos alvos putativos para novos antifúngicos ([SIVASHANKARI; SHANMUGHAVEL, 2007](#)).

33.3.2 Bancos de Dados Genômicos

De acordo com [Doolittle \(1990\)](#), as biossequências podem ser interpretadas como cadeias de texto. A partir disto, os biólogos moleculares podem coletá-las e guardá-las em arquivos de texto. No início do processo de sequenciamento era feito desse modo.

Com o avanço da tecnologia, aumentou-se a produção de biossequências. Devido a isto, os dados armazenados tiveram um grande crescimento, dificultando a sua manutenção e o acesso a partir dos programas responsáveis pela manipulação das informações. Então, os biólogos moleculares começaram a usar Sistemas Gerenciadores de Bancos de Dados (SGBD), mais apropriados para gerenciar grandes volumes de dados.

Assim que as biossequências passaram a ser armazenadas, esses dados só eram obtidos através de artigos científicos. Com o aumento das biossequências e com o avanço da tecnologia, esses dados passaram a ser submetidos aos bancos de dados através da Internet ([DOOLITTLE, 1990](#)). Essa mudança possibilitou uma facilidade na submissão das biossequências aos bancos de dados e, consequentemente, os pesquisadores puderam acessar e realizar novas análises mais rapidamente.

Os Bancos de Dados de Biologia Molecular (BDBM) operam com sistemas estruturados como: orientados a objetos e banco de dados relacional, e também o *flat files* (NAVATHE; KOGELNIK, 1999), que são sistemas não estruturado.

Existe uma grande variedade de BDBM, porém esses bancos de dados armazenam, além de sequências de nucleotídeos (DNA) e de aminoácidos (proteínas), inúmeras informações bem específicas. A exemplo disto, tem-se organismos especiais (ex.: *Eukariotic* (PÉRIER et al., 2000) e *Drosophila* (CONSORTIUM, 2003)), enzimas, famílias de biossequências (filogenia), biossequências específicas (ex.: tRNA e rRNA), entre outras. Existem outros bancos de dados que guardam estruturas tridimensionais das biossequências, por exemplo, o *The Protein Data Bank* (PDB) (BERMAN et al., 2000).

Entre os BDBM que armazenam sequências de nucleotídeos e de aminoácidos, pode-se destacar, *Gen-Bank Sequence Database* (BENSON et al., 2013), *The Integrated Microbial Genomes* (IMG) (MARKOWITZ et al., 2006), *EMBL Nucleotide Sequence Database* (KANZ et al., 2005), *Protein Identification Resource* (PIR) - *International Protein Sequence Database* (BARKER et al., 2000). Estes bancos de dados apresentam anotações relevantes armazenadas, além das próprias biossequências.

33.3.3 Alinhamento de Sequência

O alinhamento de sequências consiste no processo de comparar duas ou mais sequências (de nucleotídeos ou aminoácidos) observando-se seu nível de similaridade. Esta técnica de comparação de sequências é implementada segundo um conceito de desenvolvimento de programas conhecido como algoritmo guloso, um dos pilares da bioinformática. Algoritmo guloso é ideal para situações de otimização, pois a cada decisão ele irá escolher a alternativa mais promissora (PROSOCIMI, 2007).

Com o alinhamento de sequências, pode-se realizar a identificação de genes e proteínas desconhecidas, comparação da ordem de nucleotídeos, genes e aminoácidos em genomas de organismos proximamente relacionados, mapeamento de sequências expressas dentro de um genoma para a identificação de genes, além de fazer montagem de genomas. Na genômica comparativa existem basicamente dois tipos de alinhamentos: global e local.

O alinhamento global obtém a correspondência máxima entre as sequências, pois assume que as duas sequências são semelhantes. Este alinhamento tenta combinar as duas sequências ao longo de toda sua extensão, mesmo que estas sejam diferentes em algumas partes (NAVEED; SIDDIQUI; AHMED, 2005). Na Figura 361 demonstrou-se um alinhamento global.

```

NLGPSTKD[FGK]ISESREFDNQ
|   |   |   |
QLNQLERS[FGK]INMRLEDALV
  
```

Figura 361 – Exemplo de um Alinhamento Global. Fonte: (NAVEED; SIDDIQUI; AHMED, 2005).

O alinhamento local procura a parte das duas sequências que combinam bem (figura (NAVEED; SIDDIQUI; AHMED, 2005)). O resultado desse alinhamento é a região que apresenta uma grande semelhança, de acordo com algum critério (NAVEED; SIDDIQUI; AHMED, 2005).

```

NLGPSTKD[FGK]ILGPSTKDDQ
|   |   |   |
QNQLERSSN[FGK]INQLERSSNN
  
```

Figura 362 – Exemplo de um Alinhamento Local. Fonte: (NAVEED; SIDDIQUI; AHMED, 2005).

33.3.4 Algoritmo Needleman-Wunsch

O algoritmo *Needleman-Wunsch* (1970) foi proposto por (NEEDLEMAN; WUNSCH, 1970) para alinhamento global. Este algoritmo busca a solução ótima, utilizando o conceito de matriz de pontuação (*scores*) para medir a similaridade dos caracteres. O número de suas linhas e colunas é fornecido de acordo com o tamanho das duas sequências que deram entrada no algoritmo (WANG, 2002).

Este algoritmo utiliza três parâmetros para realizar o alinhamento, são eles: *match* (pares de bases iguais), *mismatch* (pares de bases diferentes) e *gap* (penalidade por lacuna).

A Figura 363 mostra um exemplo de como é feito um alinhamento global com o algoritmo de *Needleman-Wunsch*. Foram utilizados duas sequências: a primeira GCAT; e a segunda GAT. Os parâmetros são definidos da seguinte maneira: *match*: 1; *mismatch*: -1; *gap*: -1.

	-	G	C	A	T
-	0	-1	-2	-3	-4
G	-1	1	0	-1	-2
A	-2	0	0	1	0
T	-3	-1	-1	0	2

Figura 363 – Matriz de Pontuação para o Alinhamento Global com o Algoritmo de *Needleman-Wunsch*.

M é a matriz, i é a linha da matriz e j a coluna da matriz.

Para inicializar a primeira linha da matriz foi utilizado a Equação 33.1.

$$M[0][i] = M[0][i - 1] + gap; \quad (33.1)$$

Para inicializar a primeira coluna da matriz foi utilizado a Equação 33.2.

$$M[j][0] = M[j - 1][0] + gap; \quad (33.2)$$

Para preencher o restante da matriz é preciso realizar o cálculo do valor da diagonal superior esquerda, do topo e da esquerda de cada célula da matriz.

O cálculo para obter o valor da diagonal superior esquerda, foi utilizando a Equação 33.3.

$$diagonal = M[i - 1][j - 1] + score; \quad (33.3)$$

O score será igual ao *match* ou *mismatch*

O cálculo para obter o valor do topo, foi utilizando a Equação 33.4.

$$topo = M[i - 1][j] + gap; \quad (33.4)$$

O cálculo para obter o valor da esquerda, foi utilizando a Equação 33.5.

$$esquerda = M[i][j - 1] + gap; \quad (33.5)$$

Após realizar os cálculos da diagonal superior esquerda, do topo e esquerda é preciso obter o valor máximo da célula da matriz, foi utilizado a Equação 33.6 para obter este valor:

$$M[i][j] = max(diagonal, topo, esquerda); \quad (33.6)$$

As setas informam qual foi a célula que deu origem a cada entrada da matriz, a partir delas é possível construir o alinhamento.

O alinhamento é feito seguindo duas regras:

1. Se a seta veio da diagonal, as duas sequências recebem caracteres;
2. Se a seta veio da esquerda, a sequência 1 recebe caractere e a sequência 2 recebe a lacuna (*gap*).

Neste caso, o resultado final ficaria assim, sequência 1: GCAT e sequência 2: G-AT.

33.3.5 Algoritmo Smith-Waterman

O algoritmo *Smith-Waterman* foi desenvolvido para encontrar o alinhamento local ideal entre duas sequências. Este algoritmo foi proposto por (SMITH; WATERMAN et al., 1981) e aprimorado por Gotoh (1982). O alinhamento de duas sequências é baseado no cálculo de uma matriz de pontuação (*scores*). O número de suas linhas e colunas é fornecido pelo número de resíduos nas sequências de consulta e banco de dados, respectivamente.

Este algoritmo utiliza os mesmos parâmetros do algoritmo *Needleman-Wunsch*.

A Figura 364 mostra um exemplo de como é feito um alinhamento local com o algoritmo de *Smith-Waterman*. Foram utilizadas duas sequências: a primeira GGAT; e a segunda GAT. Os parâmetros são definidos da seguinte maneira: *match*: 1; *mismatch*: -1; *gap*: -1.

	-	G	G	A	T
-	0	0	0	0	0
G	0	1	1	0	0
A	0	0	0	2	1
T	0	0	0	1	3

Figura 364 – Matriz de Pontuação para o Alinhamento Local com o Algoritmo de *Smith-Waterman*.

M é a matriz, i é a linha da matriz e j a coluna da matriz.

Para inicializar a primeira linha e coluna da matriz é utilizado o valor 0, foi utilizando a Equação 33.7.

$$M[0][0] = 0; M[0][1] = 0; \quad (33.7)$$

Para preencher o restante da matriz é preciso realizar o cálculo do valor da diagonal superior esquerda, do topo e da esquerda de cada célula da matriz.

O cálculo para obter o valor da diagonal superior esquerda, foi utilizando a Equação 33.8.

$$\text{diagonal} = M[i-1][j-1] + \text{score}; \quad (33.8)$$

O score será igual ao *match* ou *mismatch*

O cálculo para obter o valor do topo, foi utilizando a Equação 33.9.

$$\text{topo} = M[i-1][j] + \text{gap}; \quad (33.9)$$

O cálculo para obter o valor da esquerda, foi utilizando a Equação 33.10.

$$\text{esquerda} = M[i][j-1] + \text{gap}; \quad (33.10)$$

Após realizar os cálculos da diagonal superior esquerda, do topo e esquerda é preciso obter o valor máximo da célula da matriz, foi utilizado a Equação 33.11 para obter este valor:

$$M[i][j] = \max(\text{diagonal}, \text{topo}, \text{esquerda}); \quad (33.11)$$

As setas informam qual foi a célula que deu origem a cada entrada da matriz, a partir delas é possível construir o alinhamento.

O alinhamento é feito seguindo duas regras:

1. Se a seta veio da diagonal, as duas sequências recebem caracteres;
2. Se a seta veio da esquerda, a sequência 1 recebe caractere e a sequência 2 recebe a lacuna (*gap*).

Neste caso, o resultado final ficaria assim, sequência 1: GAT e sequência 2: GAT.

33.3.6 BLAST

O *Basic Local Alignment Search Tool*⁷ (BLAST) é uma ferramenta de busca de similaridade entre sequências biológicas (DNA ou aminoácidos). O algoritmo BLAST e o programa computacional que o implementa foram desenvolvidos por [Altschul et al. \(1990\)](#), na década de 1980.

O algoritmo prioriza alinhamentos de locais específicos da sequência, com respaldo estatístico, em lugar de realizar alinhamentos “globais”, como os realizados por programas de alinhamento de múltiplas sequências, como o Clustal⁸ ([CHENNA et al., 2003](#)). Com isto, o programa é capaz de identificar relação entre sequências que compartilham similaridade, mesmo que esta ocorra somente em algumas regiões isoladas ([ALTSCHUL et al., 1990](#)).

O BLAST é um algoritmo heurístico, ou seja, dado um estado inicial, o algoritmo busca pelo melhor resultado possível durante o alinhamento, porem não garante que o resultado ótimo seja atingido, como nos algoritmos de *Needleman-Wunsch* e *Smith-Waterman*.

Na Figura 365 apresentou-se um exemplo de alinhamento utilizando o algoritmo BLAST. Pode-se observar alguns dados que são disponibilizados no resultado do alinhamento: *Query* é a sequência de entrada para realizar o alinhamento. O *Subject* é uma sequência do banco de dados. *Score* é a nota atribuída pelo algoritmo baseada no número de pareamentos perfeitos (*match*) e imperfeitos (*mismatch*) entre a sequência de entrada e a sequência do banco de dados. O *Expect* é o *e-value*, que indica o número de alinhamentos que seriam esperados apresentando valores de *score* iguais ou melhores que o encontrado por acaso, dadas as sequências para alinhamento. O *Identities* é o quanto uma sequência é parecida com a outra.

33.3.7 System Usability Scale (SUS)

A usabilidade de interfaces segundo [Neto e Riccio \(2003\)](#) é uma qualidade que pode ser afetada por aspectos de outras categorias como funcionalidade, confiabilidade e eficiência.

Segundo [Schmeil \(2013\)](#), um sistema de informação está inserido à tecnologia da informação, no entanto não se resume apenas a isto, está também inserido ao contexto que é utilizado, como os perfis dos usuários que o utilizam e como os dados são coletados para análise.

John Brooke em 1986 ([BROOKE, 1996](#)) desenvolveu o instrumento *System Usability Scale* (SUS), que possui um questionário com 10 perguntas que visa medir a usabilidade de diversos produtos e serviços. Este questionário é bastante utilizado para avaliar diversos produtos e serviços, como hardware, WebSites entre outros. A Tabela 232 mostra o questionário, com repostas numéricas quem vão de 1 (Discordo Completely) até 5 (Concordo Fortemente).

Logo após o uso do sistema, é aplicado o questionário ao entrevistado, depois de respondido é preciso aplicar três regras de avaliação definidas por [Brooke \(1996\)](#), mostradas a seguir:

1. As respostas das perguntas ímpares (1,3,5,7 e 9), subtraia 1 da resposta selecionada;
2. As respostas das perguntas pares (2,4,6,8 e 10), subtraia de 5 do valor da resposta;
3. Por último, some todas as respostas e multiplique o resultado por 2,5.

Depois da execução deste processo, o resultado obtido pode variar de 0 a 100. Com resultado quantitativo gerado, é preciso converter para qualitativo. Seguindo esse contexto, [Brooke \(1996\)](#) propôs a Tabela 233 para que fosse possível realizar classificação do produto avaliado.

⁷ <https://blast.ncbi.nlm.nih.gov/Blast.cgi>

⁸ <https://www.ebi.ac.uk/Tools/msa/clustalo/>

BLASTN 2.2.31+

Query= NC_003143.1 Yersinia pestis C092 chromosome, complete genome
Length=4653728

Subject= NC_003902.1 Xanthomonas campestris pv. campestris str. ATCC 33913
chromosome, complete genome
Length=5076188

Score = 2815 bits (1524), Expect = 0.0
Identities = 2498/2961 (84%), Gaps = 96/2961 (3%)
Strand=Plus/Plus

Query	Sbjct	Sequence	Start	End
3649197	4557914	GTTGTAAGGTTAACGCTCACGGATCATTAGTACTGGTTAGCTCAATACATCGCTGCACTT 	3649256	4557973
3649257	4557974	ACACACCCCAGCCTATCAACGTCAATAGCTTTAACGTTCTTCAGGGGCTTAAGCCCCA 	3649316	4558031
3649317	4558032	GGAAAGACTCATCTGAGGCCAACGTTCCGCTTAGATGCTTCAGCGGTTATCTTCCG 	3649376	4558091
3649377	4558092	AATTTAGCTACCGGGCAATGCCATTGGCATGACAACCCGAACACCCAGTGATTCTGCACT 	3649436	4558151
3649437	4558152	CCGGTCTCTCGTACTAGGAGCAGCCCCCTCTCAATCTTCAAACGCCACGGAGATAAGGG 	3649496	4558211
3649497	4558212	ACCGAACTGTCTACGACGTTCAAACCCAGCTCGCTTACCACTTAAATGGCGAACAGC 	3649556	4558271
3649557	4558272	CATACCCCTGGGACCTACTTCAGCCCCAGGATGTGATGAGCCGACATCGAGGTGCCAAC 	3649616	4558331
3649617	4558332	ACCGCCGTCGATGAACTCTGGCGGTATCAGCTGTTATCCCCGGAGTACCTTTAT 	3649676	4558391

Figura 365 – Exemplo de Alinhamento com Algoritmo BLAST.

33.4 GeMapCom

Esta seção apresenta informações sobre a ferramenta GeMapCom, abordando a metodologia seguida, seu funcionamento, os testes realizados e os resultados alcançados.

33.4.1 Diagrama de caso de uso

O diagrama de casos de uso resume os detalhes dos usuários do sistema e as interações deles com o sistema, definido na *Unified Modeling Language* (UML), que tem como objetivo a elaboração da estrutura de projetos de software.

A Figura 366 demonstra o ator, definido pela UML como o usuário do sistema, interagindo com a ferramenta GeMapCom, além de mostrar os casos de uso, que são tarefas ou funcionalidades realizadas pelo ator.

33.4.2 Requisitos Funcionais e Não Funcionais

Os requisitos são as variadas funcionalidades que um sistema pode fornecer ao seu usuário, já os requisitos não funcionais são as regras de negócio e restrições que o sistema possui, as Tabelas 234 e 235 mostram os requisitos funcionais e não funcionais do sistema, respectivamente.

Tabela 232 – Questionário de Avaliação do Instrumento SUS.

Ordem	Pergunta
1	Eu acho que gostaria de usar esse sistema com frequência
2	Eu acho o sistema desnecessariamente complexo
3	Eu achei o sistema fácil de usar
4	Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema
5	Eu acho que as várias funções do sistema estão muito bem integradas
6	Eu acho que o sistema apresenta muita inconsistência
7	Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente
8	Eu achei o sistema atrapalhado de usar
9	Eu me senti confiante ao usar o sistema
10	Eu precisei aprender várias coisas novas antes de conseguir usar o sistema

Tabela 233 – Sistema de Pontuação Utilizado pelo SUS.

Pontuação do Usuário	Pontuação SUS
0 até 25	Pior que o imaginável.
26 até 40	Fraco.
41 até 50	Mediano.
51 até 70	Bom.
71 até 85	Muito Bom.
86 até 100	Melhor que o imaginável.

Tabela 234 – Requisitos Funcionais da Ferramenta GeMapCom.

Identificador	Descrição
RFO1	O usuário deve realizar alinhamento de duas sequências.
RFO2	O usuário deve realizar alinhamento de duas sequências utilizando os algoritmos de Needleman-Wunsch, Smith-Waterman ou BLAST
RFO3	O usuário deve realizar alinhamento de duas sequências utilizando os parâmetros de <i>match</i> , <i>mismatch</i> e <i>gap</i>
RFO4	O usuário deve salvar alinhamento de duas sequências em imagem ou texto
RFO5	O usuário deve fazer conversão de Genbank para Fasta ou de Fastq para Fasta

33.4.3 Ferramentas e Tecnologias

As ferramentas e tecnologias utilizadas para o desenvolvimento desde trabalho foram selecionadas de acordo com a eficiência que estas mostraram para o desenvolvimento de outros trabalhos.

A linguagem de programação selecionada foi o *python*, por ser uma linguagem de programação de alto nível e orientada a objetos. Foi utilizado o Qt, que é um mescla de bibliotecas em C++ e ferramentas de desenvolvimento, que consiste em abstrações independentes de plataforma, para criação de interfaces gráficas de usuários. O Qt pode suportar multiplataformas, como sistemas *mobiles*, criação de ambientes

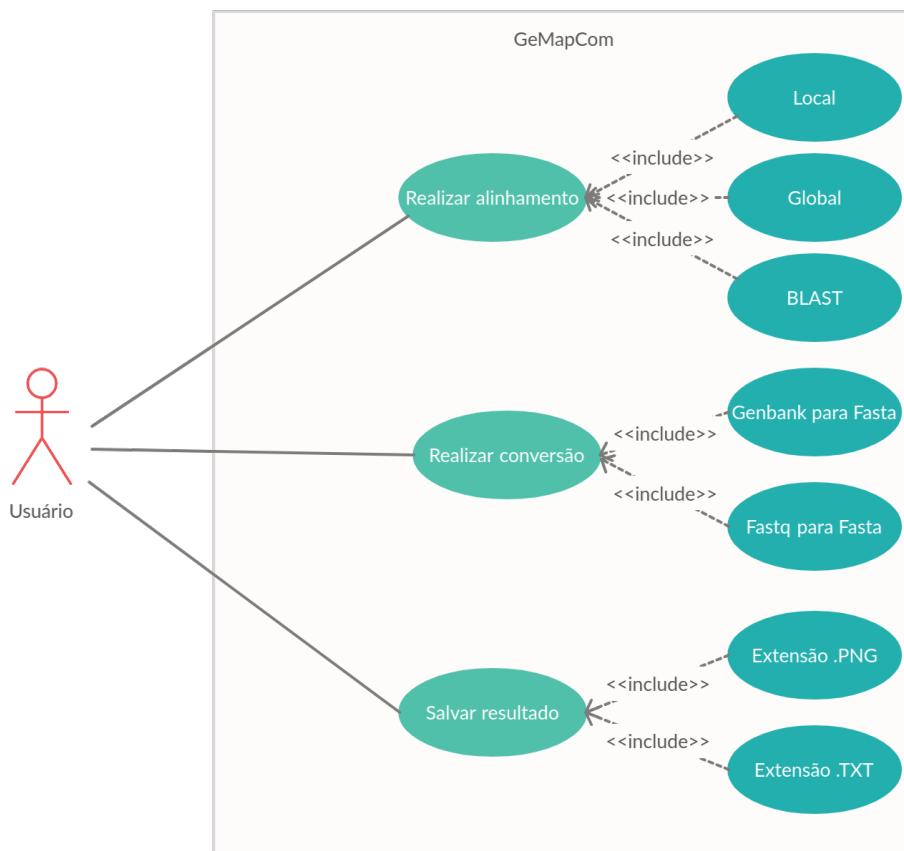


Figura 366 – Diagrama de Caso de Uso com o Ator Interagindo com a Ferramenta GeMapCom.

desktop, entre outras.

33.4.4 Telas do Sistema

A metodologia utilizada para o desenvolvimento deste trabalho consiste em tornar a experiência de uso do usuário a melhor possível, utilizando a ferramenta de maneira interativa apresentando informações de forma clara e coesa.

A Figura 367 demonstra a arquitetura utilizada no desenvolvimento desta ferramenta, elucidando a comunicação entre usuários e a ferramenta.

A Figura 368 mostra a tela principal da ferramenta, com uma interface disponibilizando as seguintes funções, que podem ser utilizadas pelos usuários: Comparar Sequências; Converter Sequência; Outras Ferramentas; e Sair.

Na Figura 369 podemos observar a tela de comparação. No campo de seleção de tipo de comparação, os usuários podem escolher tanto global como local, sendo usados os algoritmos *Needleman-Wunsch* e *Smith-Waterman*, respectivamente. Nos campos de *match*, *mismatch* e *gap* são passados os números para alinhamento das sequências. O *match* é quando os caracteres das sequências são iguais, o *mismatch* ocorre quando os caracteres são diferentes e o *gap* é quando acontece uma penalidade por lacuna.

A Figura 370 mostra a tela de resultado, que está dividida com algumas informações sobre a comparação genômica. Tem-se a identidade, que é como as sequências são pertencentes uma a outra, o score, que é calculado a partir das matrizes de pontuações criadas pelos algoritmos de *Needleman-Wunsch* e *Smith-Waterman*, medindo a similaridade entre os caracteres das sequências, e o *sym* que é onde aconteceu o *match* entre os caracteres. O usuário pode salvar esse resultado em um arquivo com extensão texto ou imagem.

Tabela 235 – Requisitos não Funcionais da Ferramenta GeMapCom.

Identificador	Descrição	Categoria
RNFO1	O sistema deve apresentar boa visualização de dados genômicos.	Usabilidade de Interface
RNFO2	O sistema deve apresentar fácil entendimento ao usuário.	Usabilidade de Interface
RNFO3	O sistema deve apresentar rapidez nos alinhamentos de sequências.	Desempenho

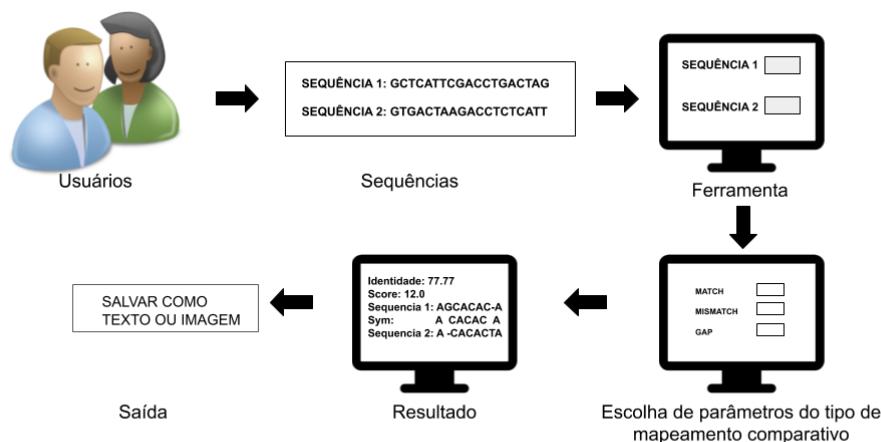


Figura 367 – Arquitetura da Ferramenta GeMapCom.

A Figura 371 mostra a funcionalidade de converter sequências, sendo suportados dois formatos para a conversão de Genbank para Fasta e de Fastq para Fasta.

A Figura 372 (a) mostra o resultado salvo em imagem, com os seguintes dados: identidade que significa o quanto uma sequência é parecida com a outra (resultado dado em porcentagem), o score, que é a indicação se o alinhamento foi bom ou não, tem as legendas verde, que foi onde aconteceu o *match*, e a vermelha que indica onde não aconteceu o *match*. A Figura 372 (b) mostra o resultado salvo em texto, com os mesmos dados da 372 (a), menos a legenda.

Na Figura 373, pode-se observar a tela BLAST, e, que os usuários vão poder salvar o resultado em vários formatos. Nos campos de *match*, *mismatch* e *gap* os números são passados para alinhamento das sequências. O BLAST pode ser utilizado pelo site ou por um *software* instalado na máquina do usuário.

A Figura 374 mostra o resultado gerado do alinhamento com o BLAST.

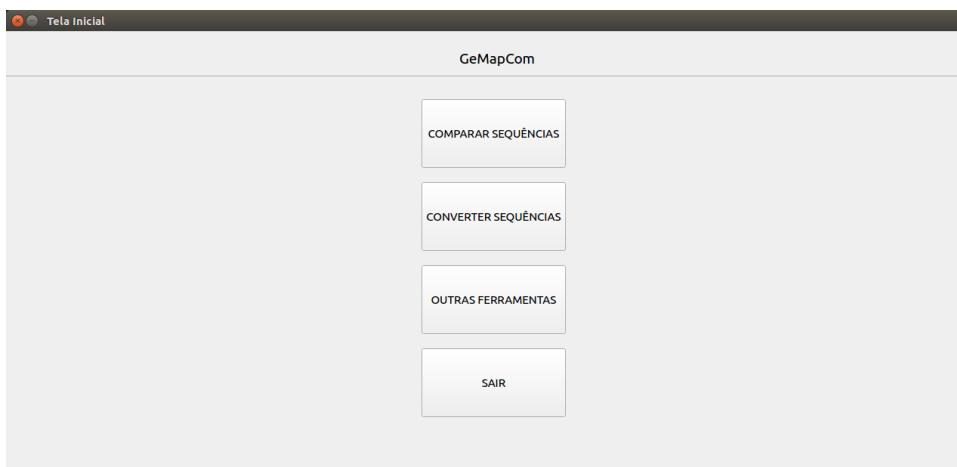


Figura 368 – Tela Principal da Ferramenta GeMapCom.

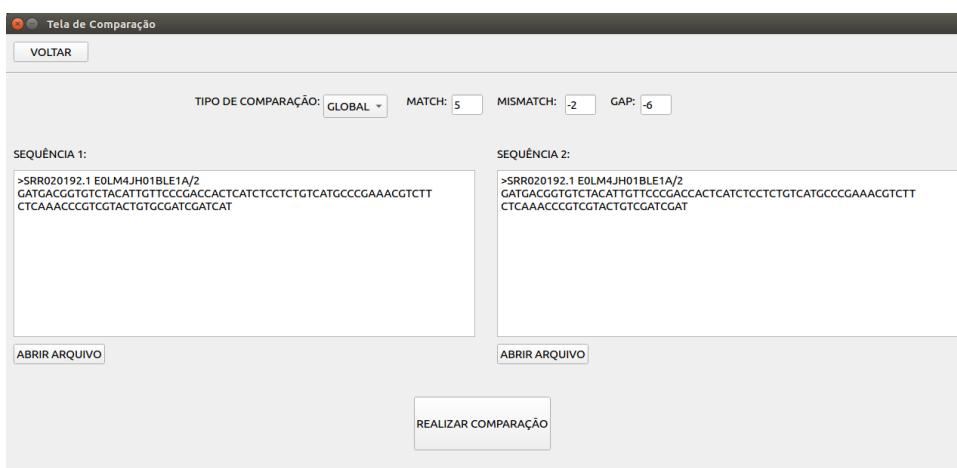


Figura 369 – Tela de Comparação da Ferramenta GeMapCom.

33.5 Avaliação

33.5.1 Avaliação da usabilidade

Para fazer a avaliação de usabilidade desta ferramenta, foi utilizado o instrumento SUS, que é um método de avaliação quantitativa e qualitativa. O teste foi realizado com nove pessoas, com três perfis diferentes:

1. Quatro pessoas trabalham com ferramentas na área da bioinformática;
2. Três pessoas já utilizaram alguma vez ferramentas na área da bioinformática;
3. Duas pessoas nunca utilizaram ferramentas na área da bioinformática, mas possuem um pouco de conhecimento de bioinformática.

O teste de usabilidade ocorreu por uma videoconferência, onde foi feita uma demonstração da ferramenta, mostrando as funcionalidades desenvolvidas, como comparar sequências, converter sequências e outras ferramentas. Logo após a demonstração, foi aplicado o questionário do instrumento SUS para os nove participantes.

Com as respostas do questionário, foi realizado o levantamento de pontuação seguindo as três regras disponibilizadas na Seção 33.3, na Seção 33.3.7. Foi obtida uma média de 88,3 pontos, sendo uma pontuação “melhor que o imaginável”, de acordo com a Tabela 233.

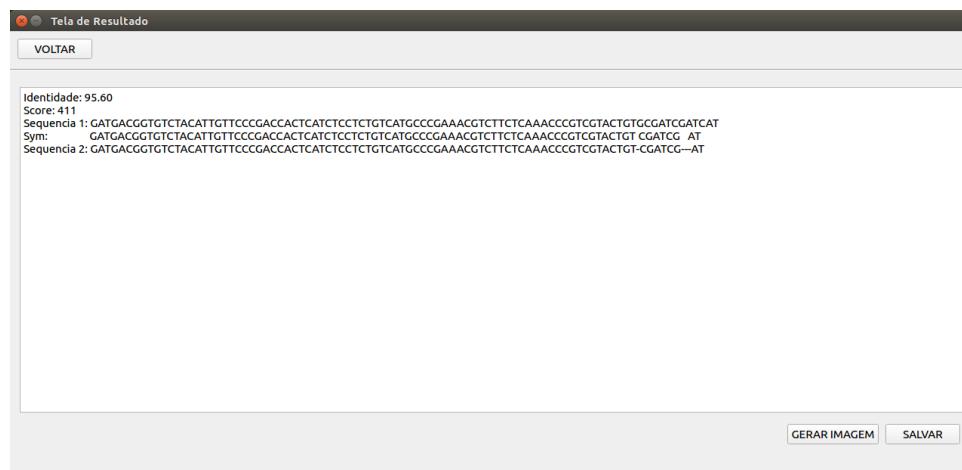


Figura 370 – Tela de Resultado da Ferramenta GeMapCom.

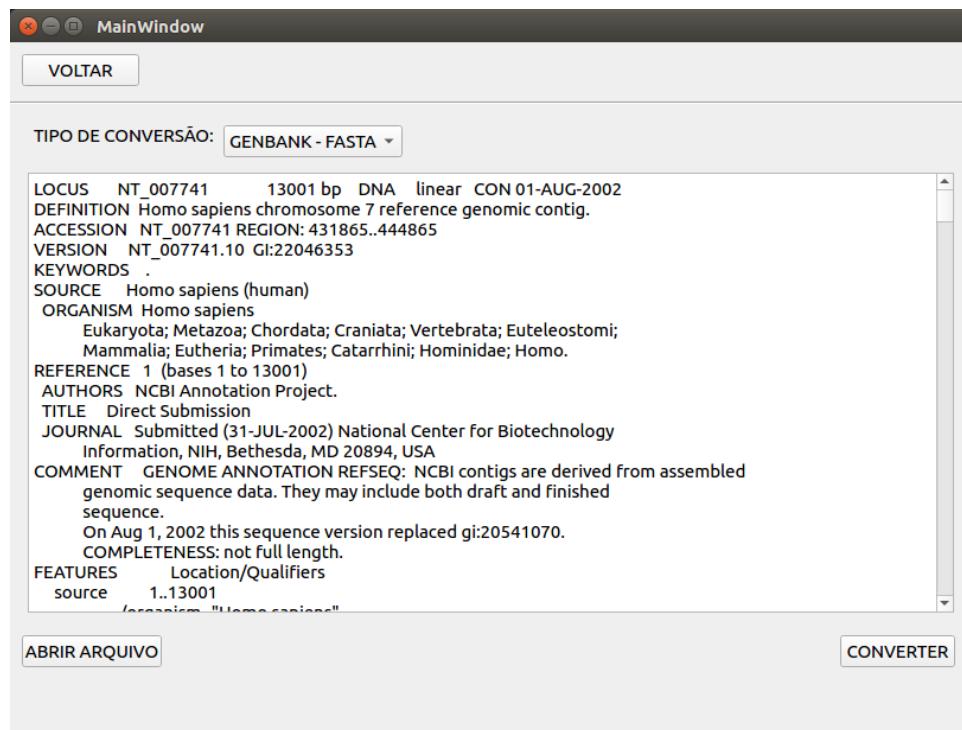


Figura 371 – Tela de Conversão de Sequência da Ferramenta GeMapCom.

Na Figura 375, pode-se observar que todas as respostas passaram dos 70 pontos e 44,4% das respostas alcançaram 90 ou mais pontos.

A partir dos dados coletados com as respostas do questionário, foi feita também uma análise de cada pergunta:

- Pergunta 1: “Eu acho que gostaria de usar esse sistema com frequência”, obteve seis respostas com escore 5, duas respostas com escore 4 e uma resposta com escore 3;
- Pergunta 2: “Eu acho o sistema desnecessariamente complexo”, obteve oito respostas escore 5 e uma resposta com escore 2;

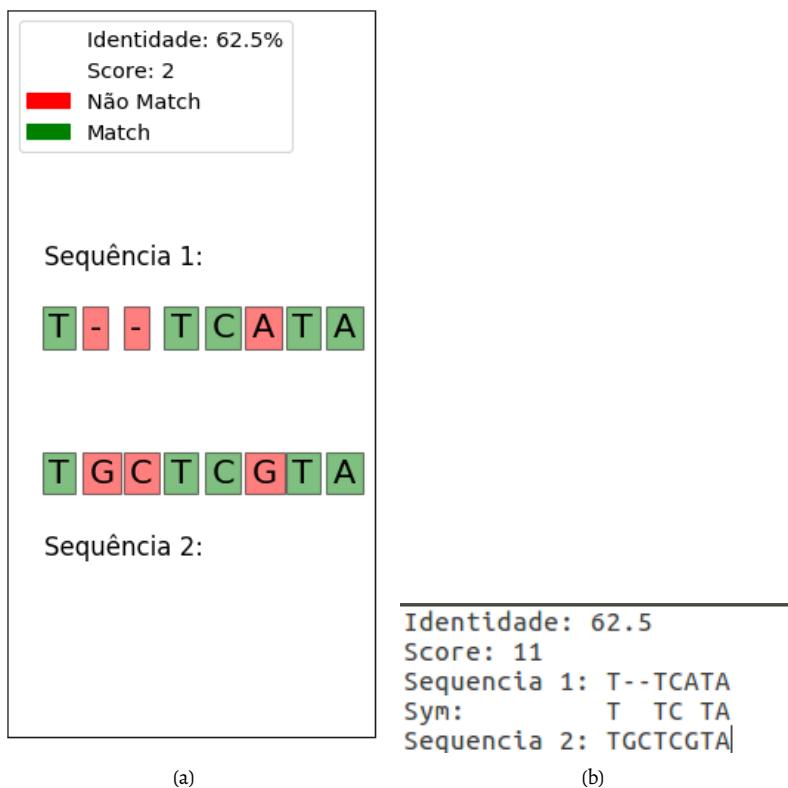


Figura 372 – (a) Resultado do Alinhamento Salvo em Imagem, (b) Resultado do Alinhamento Salvo em Texto.

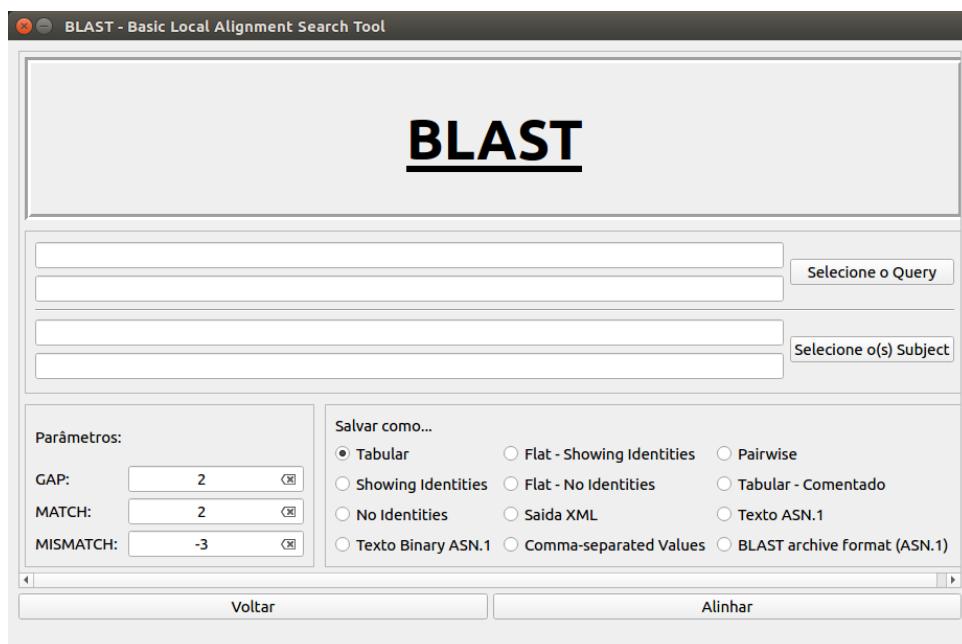


Figura 373 – Tela BLAST da Ferramenta GeMapCom.

BLAST - Basic Local Alignment Search Tool										
NC_003143.1	NC_003902.1	70.91	7203	1729	294	4197357 4204372 1050459 1043436 0.0	2830			
NC_003143.1	NC_003902.1	82.81	3212	421	97	3648978 3652127 4557678 4560820 0.0	2804			
NC_003143.1	NC_003902.1	82.81	3212	421	97	4175262 4178411 4557678 4560820 0.0	2804			
NC_003143.1	NC_003902.1	82.82	3213	419	99	14253 17402 4560820 4557678 0.0	2804			
NC_003143.1	NC_003902.1	82.81	3212	421	97	3648978 3652127 4945546 4948688 0.0	2804			
NC_003143.1	NC_003902.1	82.81	3212	421	97	4175262 4178411 4945546 4948688 0.0	2804			
NC_003143.1	NC_003902.1	82.82	3213	419	99	14253 17402 4948688 4945546 0.0	2804			
NC_003143.1	NC_003902.1	82.75	3211	425	94	4384623 4387772 4557678 4560820 0.0	2796			
NC_003143.1	NC_003902.1	82.75	3211	425	94	4384623 4387772 4945546 4948688 0.0	2796			
NC_003143.1	NC_003902.1	82.69	3212	425	98	1219466 1222615 4560820 4557678 0.0	2788			
NC_003143.1	NC_003902.1	82.69	3212	425	98	1219466 1222615 4948688 4945546 0.0	2788			
NC_003143.1	NC_003902.1	82.66	3213	424	98	4218126 4221275 4557678 4560820 0.0	2785			
NC_003143.1	NC_003902.1	82.66	3213	424	98	4218126 4221275 4945546 4948688 0.0	2785			
NC_003143.1	NC_003902.1	85.07	1567	199	30	4178918 4180465 4561296 4562846 0.0	1532			
NC_003143.1	NC_003902.1	85.07	1567	199	30	4388219 4389766 4561296 4562846 0.0	1532			
NC_003143.1	NC_003902.1	85.07	1567	199	30	12259 13806 4562846 4561296 0.0	1532			
NC_003143.1	NC_003902.1	85.07	1567	199	30	4178918 4180465 4949164 4950714 0.0	1532			
NC_003143.1	NC_003902.1	85.07	1567	199	30	4388219 4389766 4949164 4950714 0.0	1532			
NC_003143.1	NC_003902.1	85.07	1567	199	30	12259 13806 4950714 4949164 0.0	1532			
NC_003143.1	NC_003902.1	85.00	1567	200	30	1217472 1219019 4562846 4561296 0.0	1528			
NC_003143.1	NC_003902.1	84.99	1566	202	28	3652634 3654181 4561296 4562846 0.0	1528			
NC_003143.1	NC_003902.1	85.00	1567	200	30	4221782 4223329 4561296 4562846 0.0	1528			
NC_003143.1	NC_003902.1	85.00	1567	200	30	1217472 1219019 4950714 4949164 0.0	1528			
NC_003143.1	NC_003902.1	84.99	1566	202	28	3652634 3654181 4949164 4950714 0.0	1528			
NC_003143.1	NC_003902.1	85.00	1567	200	30	4221782 4223329 4949164 4950714 0.0	1528			
NC_003143.1	NC_003902.1	67.48	1685	467	66	3517213 3518854 1134008 1132362 6e-119	437			
NC_003143.1	NC_003902.1	69.14	1105	293	37	1023533 1024610 918172 917089 6e-97	364			
NC_003143.1	NC_003902.1	66.62	1306	347	75	226784 228041 1060568 1061832 1e-69	273			
NC_003143.1	NC_003902.1	80.56	319	47	12	3955751 3956065 858275 857968 1e-62	249			

Figura 374 – Resultado do Alinhamento com o BLAST.

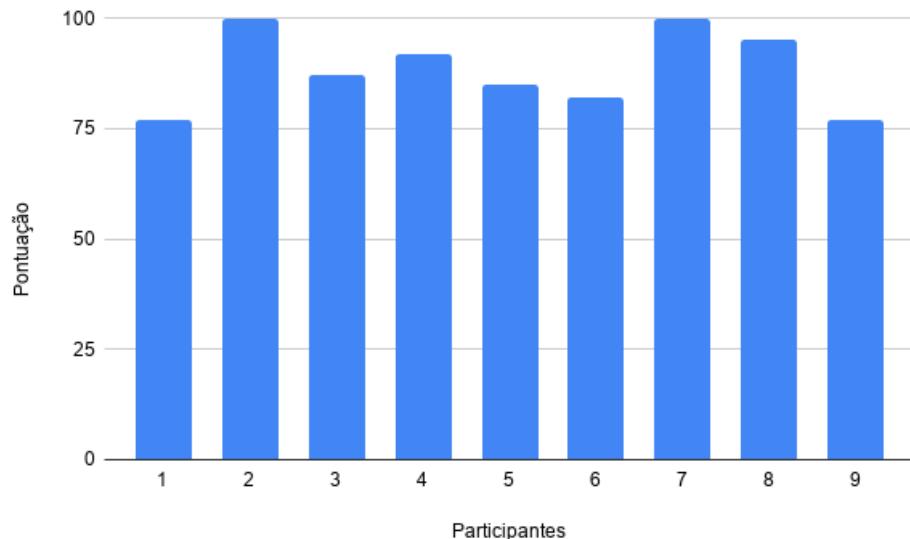


Figura 375 – Resultado da Pontuação com todos os Participantes.

- Pergunta 3: “Eu achei o sistema fácil de usar”, obteve cinco respostas com escore 5 e quatro respostas com escore 4;
- Pergunta 4: “Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema”, obteve quatro respostas com escore 1, três respostas com escore 2 e duas respostas com escore 4;
- Pergunta 5: “Eu acho que as várias funções do sistema estão muito bem integradas”, obteve cinco respostas 4 e quatro respostas 5;
- Pergunta 6: “Eu acho que o sistema apresenta muita inconsistência”, obteve sete respostas com escore 1 e duas respostas com escore 2;
- Pergunta 7: “Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente ?”, obteve sete respostas com escore 5 e duas respostas com escore 4;
- Pergunta 8: “Eu achei o sistema atrapalhado de usar”, obteve sete respostas com escore 1 e duas respostas com escore 2;
- Pergunta 9: “Eu me senti confiante ao usar o sistema”, obteve cinco respostas com escore 4 e quatro respostas com escore 5;
- Pergunta 10: “Eu precisei aprender várias coisas novas antes de conseguir usar o sistema”, obteve cinco respostas com escore 1, três respostas com escore 3 e uma resposta com escore 2.

Com a análise realizada, pode-se observar que as perguntas 1, 4 e 10 tiveram mais discordâncias nas respostas. Isso ocorreu pelo fato de que a ferramenta em questão, para ser utilizada precisa de conhecimento em como utilizar ferramentas na área da bioinformática.

De acordo com as métricas utilizadas para avaliar o desempenho de usabilidade da ferramenta e com os resultados obtidos, a ferramenta apresentou um desempenho “melhor que o imaginável”.

33.6 Conclusão

Este trabalho apresentou uma ferramenta com uma interface prática e fácil comprovada com método de avaliação de usabilidade SUS para o auxílio visual e escolha dos parâmetros desejados, com usuário o tendo controle dos dados que estão sendo passados para comparação.

Todos os requisitos definidos para execução deste trabalho foram alcançados. Para a comprovação do desempenho de usabilidade da ferramenta, foi utilizado o método de sistema de pontuação do SUS, obtendo um resultado com uma média de 88,3 pontos com a realização deste método.

Os problemas ocorridos durante o desenvolvimento foram, principalmente, realizar o levantamento de requisitos. A partir deste levantamento de requisitos foi possível estabelecer um padrão a ser seguido para o desenvolvimento das interfaces e funcionalidades da ferramenta.

Como trabalhos futuros, pretende-se: realizar o paralelismo nos algoritmos *Needleman-Wunsch* e *Smith-Waterman* para suportar sequências biológicas com uma quantidade de pares de bases maior, pois a mesma só suporta sequências relativamente pequenas; Adicionar uma funcionalidade para fazer a filtragem de alinhamentos; Além dos *feedbacks* positivos e negativos obtidos a partir da demonstração que vai ajudar aprimorar cada vez mais a ferramenta.

Apêndice

Questionário aplicado para avaliar o desempenho de usabilidade da ferramenta GeMapCom:



**TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA
“JOSÉ ALBANO DE MACEDO”**

Identificação do Tipo de Documento

- () Tese
() Dissertação
() Monografia
() Artigo

Eu, **Francisco das Chagas dos Anjos Carvalho Júnior**, autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de 02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem resarcimento de direitos autorais, o texto integral da publicação “**GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos**” de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título de divulgação da produção científica gerada pela Universidade.

Picos-PI 05 de outubro de 2020.

Francisco das Chagas dos Anjos Carvalho Júnior

Assinatura

Francisco das Chagas dos Anjos Carvalho Júnior

Assinatura

GeMapCom: Uma Ferramenta para Mapeamento Comparativo e Visualização de Dados Genômicos

Prezado(a), esse questionário é parte integrante de uma pesquisa, que tem como objetivo analisar a usabilidade da ferramenta GeMapCom.

Esta ferramenta foi desenvolvida pelo discente Francisco das Chagas dos Anjos Carvalho Júnior, com orientação do Prof. Dr. Romuere Rodrigues Veloso e Silva para trabalho de conclusão de curso em Bacharel Sistemas de Informação, na Universidade Federal do Piauí, campus Senador Helvídio Nunes Picos.

O objetivo é apresentar uma ferramenta para comparação e visualização de dados genômicos denominada GeMapCom, com objetivo de tornar a experiência do usuário a melhor possível, utilizando a ferramenta de maneira interativa apresentando informações de forma clara e coesa.

O questionário é baseado no instrumento System Usability Scale (SUS) desenvolvido por John Brooke em 1986, que possui um questionário com 10 perguntas, que visa medir a usabilidade de diversos produtos e serviços. Este questionário é bastante utilizado para avaliar diversos produtos e serviços, como hardware, WebSites entre outros. As respostas são feitas de forma numéricas quem vão de 1 (Discordo Completamente) até 5 (Concordo Fortemente).

***Obrigatório**

TERMO DE CONSENTIMENTO LIVRE ESCLARECIDO (TCLE)

Declaro ter conhecimento de que estou participando de uma avaliação de usabilidade conduzido por um estudante e um professor/pesquisador, que tem como principal objetivo conhecer avaliar a usabilidade da ferramenta GeMapCom.

Estou ciente que as informações pessoais são estritamente confidenciais, e que se houver qualquer má conduta na pesquisa quanto aos procedimentos adotados, poderei questionar e até interromper minha participação.

1. Eu acho que gostaria de usar esse sistema com frequência ? *

Marcar apenas uma oval.

1 2 3 4 5



2. Eu acho o sistema desnecessariamente complexo ? *

Marcar apenas uma oval.

1 2 3 4 5



3. Eu achei o sistema fácil de usar ? *

Marcar apenas uma oval.

1 2 3 4 5



4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema ? *

Marcar apenas uma oval.

1 2 3 4 5



5. Eu acho que as várias funções do sistema estão muito bem integradas ? *

Marcar apenas uma oval.

1 2 3 4 5



6. Eu acho que o sistema apresenta muita inconsistência ? *

Marcar apenas uma oval.

1 2 3 4 5



7. Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente ? *

Marcar apenas uma oval.

1 2 3 4 5



8. Eu achei o sistema atrapalhado de usar ? *

Marcar apenas uma oval.

1 2 3 4 5



9. Eu me senti confiante ao usar o sistema ? *

Marcar apenas uma oval.

1 2 3 4 5



10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema ? *

Marcar apenas uma oval.

1 2 3 4 5

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

Parte V

Mapeamento Sistemático para Sistemas Computacionais

Mapeamento Sistemático de Características Musicais para Reconhecimento Automático de Emoções

Autores: Naara Macedo Camilo; Deborah Maria Vieira Magalhães; Lana Beatriz Medeiros de Mesquita

Resumo: A música proporciona efeitos benéficos à mente humana. É usada no cotidiano para diminuir a tensão, como forma distração e, até mesmo, como forma de expressão de sentimentos. Por essas razões muitas pesquisas surgem com o objetivo de explorar. Nas ciências da computação, pode ser observada pesquisas em relação ao reconhecimento de emoções musicais e desenvolvimento de sistemas de recomendação de músicas. Para o reconhecimento de emoções na música é necessário extração de conteúdo musical, e métodos que facilita sua extração. Pensando nisso, este trabalho tem por objetivo apresentar, através dos resultados do Mapeamento Sistemático, um aparato geral sobre ao tema referido, de modo a auxiliar futuros estudos. Foram encontrados 35 artigos associados ao tema referido, que no total apresentou 85 características musicais, assim como, possibilitou o levantamento de dados referente aos métodos e ferramentas de extração e associação de características a emoção.

Palavras-chaves: Reconhecimento de Emoções. Música e Emoção. Extração de Características.

34.1 Introdução

A música tem sido associada à emoção durante séculos. Isso é percebido, na eras medievais em que os guerreiros entoavam canções não apenas como forma de comunicação no campo de batalha, mas, também, como elemento psicológico para animar as tropas e deixarem o inimigo temeroso (CARVALHO, 2007). Além desse contexto histórico, existe o caso da música “Gloomy Sunday”, que é denominada: “a música do suicídio”, por conta dos quadros de suicídios ocorridos ao ouvi-la (STACK; KRYINSKA; LESTER, 2008).

Atualmente, diversas áreas tem estudado a relação da música com a emoção. O campo da psicoacústica realiza estudos das sensações humanas em relação à música, a partir da percepção sonora do indivíduo, através da análise de características acústicas, como: timbre, volume e rugosidade (COUTINHO; CANELOSI, 2011). A musicologia, estuda a fundamentação teórica e histórica dos aspectos musicais (JUSLIN;

[SLOBODA, 2001](#)). Na área das ciências da computação, que associada as áreas anteriormente citadas, auxilia no reconhecimento automático das emoções na música ([TROHIDIS et al., 2008](#)).

A extração de características musicais é o que permite a análise do seu conteúdo. Dessa maneira, é utilizada a Recuperação da Informação Musical (RIM), que possibilita a extração de informação da música para uso multidisciplinar ([LAMERE, 2008](#)). A RIM está aliada ao estudo sobre Reconhecimento de Emoções Musicais (REM), que tem trazido resultados relevante para pesquisa, como apresentação de combinações de características musicais que fornecem melhores resultados para o reconhecimento da emoção musical no campo computacional ([BARTHET; FAZEKAS; SANDLER, 2012](#)).

Através do estudo do REM e RIM, é possível a resolução de alguns problemas existentes na pesquisa, como: classificação de gêneros musicais, recomendação musical e o auxílio na musicoterapia para recomendação da música. A classificação de gêneros pode ser utilizada para estabelecer correlação à emoção. A recomendação musical é usada por aplicações para indicar músicas de acordo com a emoção do ouvinte. A musicoterapia auxilia em tratamentos neurológicos.

Entretanto, a área da percepção musical é relativamente nova à luz das outras áreas da psicoacústica, percepção visual e percepção da fala. Por conseguinte, tem enfrentado dificuldades na pesquisa, como, por exemplo, a subjetividade das respostas emocionais, o que afeta no REM. Outro problema pertinente é a ausência de bases de dados. Além de a ambiguidade dos adjetivos usados para rotular as músicas.

Baseado na observação das lacunas existentes na área de música e emoção associadas a computação, demonstra necessidade de pesquisas para o seu desenvolvimento.

34.1.1 Objetivos

O objetivo principal deste mapeamento sistemático se dedica a responder a seguinte questão de pesquisa: Como um sistema computacional consegue identificar a emoção transmitida por uma música?

Os objetivos específicos deste trabalho são:

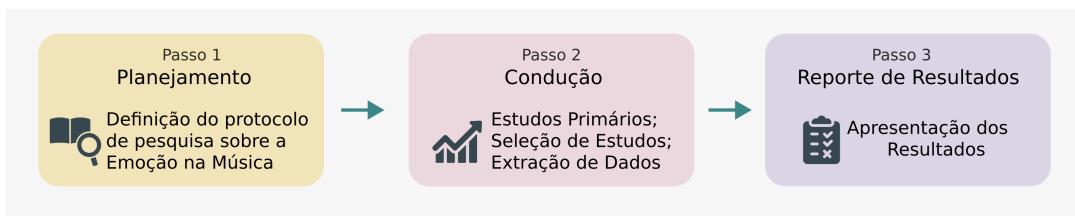
1. Estabelecer correlação entre as características musicais e uma emoção, ou um conjunto de emoções
2. Apresentar um conjunto de características musicais utilizados na literatura para reconhecimento automático de emoções e, dentre elas, quais apresentam melhor resultado na associação a emoção;
3. Reunir métodos, ferramentas e informações sobre os processos utilizados na literatura para reconhecimento de emoções musicais.

34.1.2 Metodologia

Este estudo partiu da investigação da relação de intervalos musicais, especificamente o intervalo da terça menor, com a tristeza. A dificuldade de encontrar trabalhos especializados que tratassesem essa relação, nos levou a ampliar nossa busca para compreender como sistemas computacionais poderiam detectar automaticamente a emoção transmitida em uma música. Para tanto, fizemos uso de uma metodologia bem definida que guiou nosso levantamento do estado da arte.

O presente estudo foi iniciado a partir da formulação de questões de pesquisas, a fim de serem respondidas através do mapeamento sistemático. Foi realizada a busca nas bases com aplicação previa do filtro do idioma e a disponibilidade do artigo. A partir das perguntas, foi desenvolvido critérios de pesquisa, que foram aplicados através de filtros para afunilar os resultados. Com a obtenção dos artigos resultantes, iniciou a extração de dados relevantes ao tema proposto. A Figura 376, apresenta um fluxograma com as etapas de planejamento, condução e reporte dos resultados.

Figura 376 – Metodologia



34.1.3 Contribuições

Os resultados obtidos através da aplicação de uma busca sistemática possibilitou as seguintes contribuições:

1. Uma visão geral sobre a evolução dos estudos sobre o processamento de áudio para identificação de emoção na música;
2. Uma lista de características extraídas das músicas devidamente descritas;
3. A associação das características musicais e o modelo de emoção;
4. Levantamento de informações sobre as ferramentas e métodos usados para extração dos dados;
5. Destaque sobre os problemas a serem resolvidos;

34.1.4 Organização do Trabalho

Esta capítulo está dividido da seguinte forma:

Seção 2: Contém conceitos que fazem parte da construção deste trabalho, que são: a relação da música e emoção, os tipos de modelos de emoção existentes, definições sobre características musicais e acústicas e os tipos de ferramentas utilizadas na extração de características.

Seção 3: São apresentadas as fases do método de pesquisa utilizado, que é o Mapeamento Sistemático. As fases são divididas em planejamento, condução e reporte do mapeamento.

Seção 4: Detalha os resultados obtidos com a extração das informações dos artigos. Apresentamos informações gerais, em relação a bases de dados e a localização do laboratório de pesquisa dos artigos. No decorrer da seção, respondemos as questões de pesquisas através das seções de reconhecimento de emoção, os métodos para extração de características e a relação das características e a emoção. E, por fim, apresentamos uma tabela com as características mapeadas.

Seção 5: apresentamos as principais conclusões, frisando as questões encontradas na área de reconhecimento de emoção musical.

34.2 Referencial Teórico

Na presente seção é disponibilizada uma visão geral de conceitos pertinentes à condução deste trabalho. Para organização, os temas foram dispostos nas seguintes subseções: Subseção 34.2.2, trata a relação entre emoção e emoção na música; enquanto que, na Subseção 34.2.3, são expostos os conceitos sobre as características musicais. E por fim, a Subseção 34.2.4 apresenta as ferramentas para extração de características.

34.2.1 Música e emoção

Durante doze séculos, os filósofos Platão, Boécio e Erígena (de 4 aC a 9 aC) viam a música como as leis da natureza criada por Deus e suas relações com "movimento das esferas celestes"(PERLOVSKY, 2010). Apenas no século 13, direcionaram os estudos da música a emoção dos ouvintes. A partir daí foi possível estudar teoricamente a ligação entre música e emoção.

Diversos estudos tentaram desvendar a origem da música e sua relação com a emoção. Junto às teorias evolutivas da relação entre música e emoção, os estudos de Justus e Hutsler (2005) e McDermott e Hauser (2005) apresentam a teoria que o ser humano tem predisposição biológica para a música. O trabalho de Levitin (2008), levanta a ideia de que a música se originou com os gritos de animais. O autor afirma que esta teoria funciona nos dias de hoje no reconhecimentos de emoção, pelo fato de poder ser reconhecido emoção através fala.

Além de a música ser uma intimamente ligada à emoção, existe a influência cultural. Não se sabe ao certo, até que ponto essa concepção se deve ao conhecimento cultural ou percepção que excede os conceitos formados pela cultura. Alguns teóricos defendem a ideia que o significado musical é dado somente pelo costume cultural Blacking (1974), Feld (1994). Porém, estudos comprovam que o ser humano pode se adaptar a estilos e idiomas diferentes (CASTELLANO; BHARUCHA; KRUMHANSL, 1984), (KESSLER; HANSEN; SHEPARD, 1984).

34.2.2 Modelos de classificação da emoção

A emoção era considerada como um fenômeno inconveniente, causando a sua desvalorização entre os filósofos (STRONGMAN, 1978). Somente, a partir dos anos 60 e 70, os pesquisadores evidenciaram a emoção como estudo relevante (LAZARUS, 1991). Além de ser obtidos conceitos na psicologia e filosofia, estudos foram ampliados à antropologia e às neurociências (SOLOMON, 1993).

Para análise do humor ou emoção, é importante entender as diferenças conceituais entre os termos. Entre pesquisadores existe indefinição, ou até mesmo uso errôneo. No estudo de Rottenberg (2005), a emoção é conceituada como reações rápidas que ocorrem quando os organismos encontram estímulos significativos que exigem respostas adaptativas. Enquanto que humor é descrito como estados de sentimentos difusos e de movimento lento que estão fracamente ligados a objetos ou situações específicas.

No que se diz respeito ao reconhecimento de emoção, existem duas principais perspectivas: o modelo bidimensional, que apresenta a emoção através de agrupamentos no plano dimensional e o categórico, que utiliza adjetivos discretos, mas que, também, podem ser agrupados. O modelo categórico é diversas vezes referenciado através do estudo de Hevner (1936). Nesse estudo, a autora tinha o objetivo de apresentar uma gama maior de opções para medir a relação entre música e emoção. Dessa forma, utilizou 66 adjetivos, organizados em 08 grupos. Apesar de serem adjetivos discretos, são tratados pela forma agrupada, como mostrado na Figura 377.

O modelo dimensional é baseado na caracterização da emoção definida através da sua localização em duas ou três dimensões. Porém, Russell (1980), apresentou evidências de que as dimensões são inter-relacionadas sistematicamente, e podem ser representadas através de um modelo espacial, de modo a formar um círculo que envolve excitação e valência. O estudo é baseado na lista de Hevner.

Logo após, o estudo realizado no trabalho de Thayer (1990), simplificou o modelo de Russel em duas dimensões, de forma que, utilizou estresse e energia para caracterizá-las. Energia é representada por excitação, enquanto que o estresse representa a valência, como retratado na Figura 378.

Figura 377 – Lista de adjetivos de acordo com 8 grupos relacionados

		6	
		divertido	
		feliz	
		alegre	
		contente	
		animado	
		brilhante	
7			5
	hilariente		cómico
	elevado		brincalhão
	triunfante		caprichoso
	dramático		fantasioso
	apaixonado		estranho
	sensacional		enérgico
	agitado		delicado
	emocionante		leve
	impetuoso		gracioso
8	inquieto		
	vigoroso		4
	robusto		emocional
	enfático		vagaroso
	martial		satisfeito
	passado		sereno
	majestoso		tranquilo
	engrandecido		quieto
			calmante
		1	
	espiritual		
	sublime		
	imponente		
	digno		2
	sagrado		patético
	solene		doloroso
	sóbrio		triste
	serio		fúnebre
			tragico
			melancólico
			frustrado
			deprimente
			sombrio
			pesado
		3	
		sonhador	
		docil	
		afetuoso	
		sentimental	
		saudade	
		anseio	
		suplicante	
		queixoso	

Adaptado de [Hevner \(1936, p. 249\)](#)

Figura 378 – Modelo Dimensional

Adaptado de [Yeh, Lin e Chang \(2009\)](#)

Segundo [Chanel, Ansari-Asl e Pun \(2007\)](#), a valência representa o modo como se julga uma situação, de

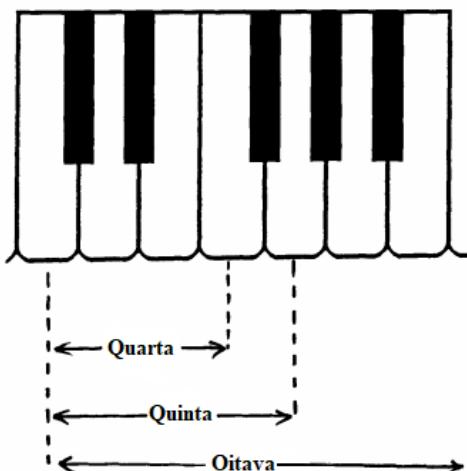
desagradável (emoção negativa) a agradável (emoção positiva). A excitação expressa o grau de entusiasmo que vai de calmo a emocionante. No trabalho de Lane, Chua e Dolan (1999), a valência é referida como à direção da ativação comportamental associada à emoção, seja em direção a emoção prazerosa ou emoção desagradável de um estímulo. O autor afirma que a excitação é proposta para ser ortogonal à valência e se refere à intensidade da ativação emocional, variando de excitada a calma.

O sistemas computacionais podem ser baseados num modelo de emoção, embora tais representações continuam sendo um tópico ativo da pesquisa em psicologia. Ambos os modelos são usados no reconhecimento da emoção da música como “verdade fundamental”, porém existe maior tendência no uso de forma categórica (KIM et al., 2010).

34.2.3 Características Musicais

No ponto de vista europeu, Pitágoras é considerado o primeiro teórico da música (PAPADOPoulos, 2002). Isso se deve pelo fato de ele ter identificado diferentes sons a partir da batida de um martelo em uma bigorna. Pitágoras percebeu que a nota musical que soava do martelo, dependia do seu peso e não do local aonde o martelo atingia a bigorna. Percebeu também, com o soar de martelos diferentes, representava períodos que na música grega eram intervalos de oitava, de quinta e de quarta, correspondem, em termos de pesos, à fração numérica de 2/1, 3/2 e 4/3 (GAFFURIUS, 2005). Dessa forma, foi descoberta as escalas musicais, que contém as notas conhecidas atualmente.

Figura 379 – Representação da Escalas musicais



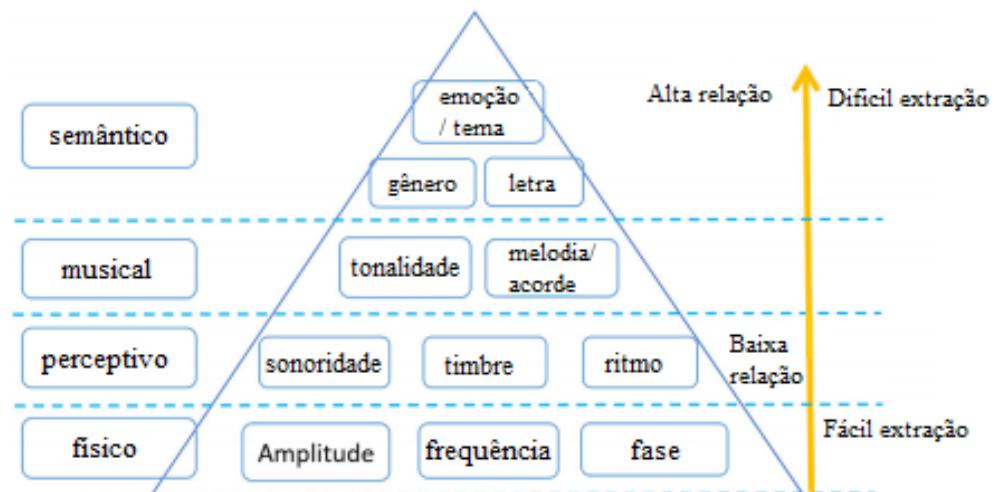
Adaptado de Papadopoulos (2002)

Durante séculos, a música foi representada por meio de transmissão auditiva e a partir de partitura. Atualmente, existem diferentes formas para representar e extrair as informações musicais. As formas mais conhecidas são MIDI e áudio. O formato MIDI, consiste em uma partitura digital, que serve para extrair informações do nível musical(apresentado a seguir). Em relação ao áudio, existem formatos, como por exemplo: MP3, WAV, FLAC, AAC, OGG. Esses e demais formatos são discutidos em Kumar e Goyal (2014).

Conforme Zhang et al. (2016), as características musicais variam entre os níveis: físico, perceptivo, musical e semântico, como apresentado na Figura 380. O nível físico, diz respeito ao nível mais baixo e é representado por características pertencentes ao sinal de áudio, como: energia do sinal, picos de frequência, amplitude. O nível de percepção é pertencente ao nível médio, é representado por características como timbre, ritmo, intensidade, volume. Em relação às características musicais, também pertencem ao nível médio e são representadas por: melodia, harmonia, acorde, tonalidade. Quanto ao nível semântico, é o nível mais alto, o qual a letra da música pode ser usada como exemplo.

Em relação ao nível de dificuldade de extração, as características de baixo nível são mais fáceis de ser extraídas e existe um conjunto de ferramentas que auxiliam em tal processo, como será abordado posteriormente. No entanto, é mais difícil estabelecer uma relação direta com a emoção que o indivíduo sente ao escutar a música. Já as características de alto nível são mais fáceis de serem conectadas a emoção. Porém, são mais difíceis de serem extraídas automaticamente (ZHANG et al., 2016).

Figura 380 – Relação entre características musicais e emoção



Adaptado de Zhang et al. (2016)

34.2.4 Ferramentas para Extração de Características

Como foi citado anteriormente no texto, o RIM permite a extração de características. Além disso, corresponde a estratégias para conceder níveis de acesso a coleções de músicas. O RIM tem sido objeto intensivo de estudo por laboratórios de pesquisa acadêmicas, indústrias e bibliotecas (LEW et al., 2006). Uma forma para avaliação de algoritmos e técnicas relacionadas ao RIM é o *Music Information Retrieval Evaluation eXchange* (MIREX) (DOWNIE, 2008).

A extração de característica do áudio é um dos pilares do desenvolvimento de processamento de sinal (MOFFAT et al., 2015). Uma diversidade de ferramentas foi desenvolvida com o intuito de facilitar a extração de recursos de áudio. Os formatos de ferramentas mais utilizados são: aplicações independentes, *plug-ins* para um aplicativo *host*, biblioteca de funções de software.

Para o processamento de sinais de áudio é importante ter ciência das informações sobre as configurações pertencentes a essa etapa. Algumas das informações são o taxa de amostra, sobreposição, Janelamento e Tamanho da Janela. O taxa de amostra é um conversor de taxa de amostragem para um sistema de áudio digital (ANDERSEN et al., 2007). sobreposição define a sobreposição dos canais/bandas de áudio. Janelamento é um padrão que pode ter diferentes cálculos, como *Hamming*, ou *Hanning*. O Tamanho da Janela define o espaçamento em cada banda de áudio.

34.3 Método de Pesquisa: Mapeamento Sistemático

O Mapeamento Sistemático consiste em um estudo baseado em trabalhos científicos, com a intenção de avaliar de forma crítica e imparcial tais trabalhos para obtenção de um aparato geral do tema pesquisado (KITCHENHAM, 2004a). A metodologia utilizada reside na submissão de *strings* de buscas em base de dados. Assim como, na divisão de estágios de leitura de artigos científicos, com o objetivo de eliminar, ou aceitar, conforme o nível de aderência. Os resultados alcançados nas etapas de planejamento, condução e extração,

descritas a seguir, foram validados por duas especialistas, uma no que diz respeito aos aspectos musicais e do mapeamento sistemático em si e outra no que diz respeito a extração de características do sinal de áudio.

34.3.1 Planejamento: Definição de Protocolo

A parte do planejamento corresponde à definição do protocolo de pesquisa. Sendo assim, o protocolo contém as seguintes informações:

- A. Questões de Pesquisa:** o objetivo desse estudo é identificar as características musicais utilizadas na literatura, para a classificação de emoções alegres e tristes na música. Pensando nisso, a primeira e principal questão de pesquisa é:

QP1 Como um sistema computacional consegue identificar a emoção transmitida por uma música?

Para a análise do conteúdo musical, se torna necessário a extração de informações musicais, pensando nisso, a seguinte questão foi definida:

QP2 Quais métodos de extração das características são utilizados para a análise de sentimentos na música?

Como forma de análise, em relação a seleção do conteúdo a ser extraído, a seguinte pergunta se tornou necessária:

QP3 Quais características musicais devem ser extraídas para determinar a emoção da música?

- B. Palavras-chave** A escolha foi baseada nas questões de pesquisas, com o propósito de identificar os métodos utilizados. Logo após, foram adicionados sinônimos a partir de uma pesquisa das palavras mais utilizadas em artigos de computação musical.

Foi possível observar em algumas bases utilizadas, que o termo “*music*” retornava títulos que abrangiam temas fora do assunto de interesse. Para restringir de modo que não interferisse de forma tendenciosa no resultado, foi preferível o uso dos termos pontuais: “*music feature*”, juntamente com “*elements of music*”.

Tendo em vista que, o objetivo deste trabalho é identificar as características musicais mais relevantes para análise da tristeza, utilizamos o termo principal “*sad*”, logo após chegar a conclusão que o termo “*sadness*” restringiria os resultados por causa das variações da palavra (*sadness*, *sad*). E para as diversas emoções acrescentamos os termos “*mood*” e “*emotion*”.

O presente estudo, tem uma parte relevante da pesquisa na forma de extração de característica utilizadas na literatura. Dessa forma, utilizamos as seguintes palavras-chave: “*recognition*”, “*analys*”, “*extraction*”, “*segmentation*”, “*feature extraction*”. Sendo que, o termo *analys*, foi escolhido para englobar as suas variações e a repetição de “*extraction*” em “*feature extraction*” é uma forma de delimitar o assunto referido.

- C. String de Busca** Com base nas palavras-chaves apresentadas anteriormente, foram definidas uma série de strings até chegar a utilizada no Mapeamento. A tabela 236 apresenta as strings que foram testadas.

Tabela 236 – Histórico de strings

Número de Versões	Strings de Busca
I	music* AND "mood music"AND feature* AND (sad* OR emotion OR "negative mood")AND (Recognition OR Analys* OR extraction* OR segmentation)
II	music* AND feature* AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation)
III	music* AND ("feature extraction"OR "music feature"OR "musical feature") AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation)
IV	music* AND "feature extraction"AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation)
V	music* AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR feature extraction*)
VI	music* AND ("music feature"OR "musical feature") AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation OR "feature extraction")
VII	music* AND ("music* feature"OR "elements of music") AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation)
VIII	("music* feature"OR "elements of music") AND (mood OR sad* OR emotion) AND (Recognition OR Analys* OR extraction* OR segmentation)
IX	("music features"OR "elements of music") AND (mood OR sad* OR emotion) AND (recognition OR analys* OR extraction* OR segmentation)

D. Fontes de Pesquisa: Para obter os primeiros resultados, foi utilizado repositórios que contém pesquisas nas áreas de computação musical e relacionadas a saúde. As bases relacionadas somente a saúde são: PubMed ¹, PubMed Central ². O uso das bases citadas é devido o estudo proposto ser intimamente ligado as áreas da psicologia. As bases relacionadas em ambas as áreas são: IEEE ³, Semantic Scholar ⁴, ACM ⁵, Springer Link ⁶ e Science Direct ⁷.

E Critérios de Seleção do Estudos: a fim de selecionar estudos que são relacionados ao tema, foram apresentados na Tabela 237, os critérios de Inclusão (I) e Exclusão (E) adotados neste trabalho.

Tabela 237 – Critérios adotados na seleção de artigos

Código	Critério
I.01	Relaciona emoção com a música.
I.02	Área da Ciências da Computação
I.03	Realiza a extração de características musicais a partir do áudio, partitura ou MIDI.
I.04	Realiza classificação/predição/reconhecimento do humor/emoção da música.
E.01	Artigo redigido em Inglês.
E.02	Título e resumo fora dos critérios de pesquisa.
E.03	Todo o texto fora dos critérios de pesquisa

Como foco do presente trabalho busca estudos que reconhecem automaticamente a música e emoção, os critérios I.01 e I.02 foram definidos. Tendo em vista o foco do trabalho, são considerados os trabalhos

¹ <https://www.ncbi.nlm.nih.gov/pubmed/>

² <https://www.ncbi.nlm.nih.gov/pmc/>

³ <https://www.ieee.org>

⁴ <https://www.semanticscholar.org>

⁵ <https://dl.acm.org>

⁶ <https://link.springer.com>

⁷ <https://www.sciencedirect.com>

que realizam a extração e classificação da emoção na música. Pensando nisso, os critérios I.03 e I.04 foram estipulados.

Além dos critérios E.02 e E.03 para exclusão, O critério E.01 foi definido uma vez que a maioria dos estudos e revistas utilizarem o idioma inglês. Vale destacar que não foi definido, como critério de exclusão, um limite para a data de publicação do trabalho, pelo fato de que os conceitos e métodos iniciais dos estudos são considerados neste trabalho.

34.3.2 Condução do Mapeamento Sistemático

A atividade de condução foi realizada através das etapas seguintes: 1. identificação dos estudos primários; 2. seleção dos estudos primários; 3. extração dos dados. As fases da condução estão dispostas nas seguintes subseções: subseção 34.3.2.1, corresponde a consulta nas bases de dados;a subseção 34.3.2.2 em que apresenta os métodos de seleção de características; e a subseção 34.3.2.3, que apresenta a extração dos dados dos artigos.

34.3.2.1 Consulta nas Bases de Dados

Nesta fase do protocolo ocorreu à busca de artigos científicos nas bases de dados de acordo com o protocolo definido. A seleção foi feita em 15 de fevereiro de 2018. A partir daí, foi realizado o *download* dos arquivos para diretórios divididos pelos nomes das bases de dados.

Pelo fato de alguns arquivos estarem indisponíveis, foi aplicado um filtro constatando esta ocorrência. O conjunto de arquivos antes do filtro aplicado era de 760 artigos, logo após o filtro ficaram 721 arquivos.

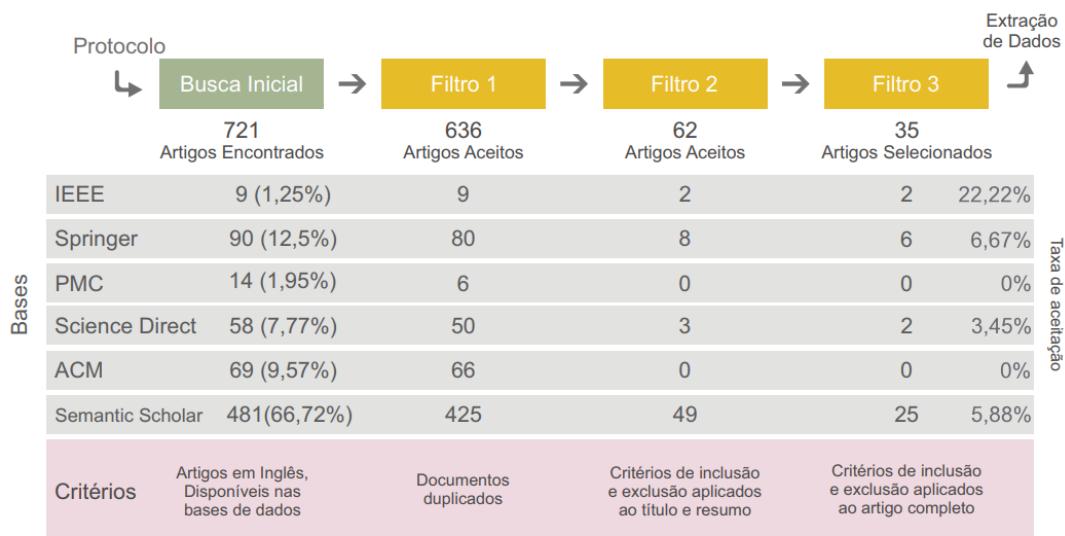
Devido o volume considerável de artigos, a organização em pastas foi inviabilizada. Portanto, adotamos o aplicativo *Mendeley*⁸ para organizar e compartilhar documentos científicos. O *Mendeley* conta com ferramentas para detecção automática de artigos duplicados, auxiliando no processo de triagem dos trabalhos.

34.3.2.2 Triagem de Trabalhos

Nessa etapa, ocorreu a escolha de trabalhos a partir da aplicação de três filtros. O primeiro filtro corresponde à exclusão dos arquivos duplicados. O segundo filtro corresponde à leitura dos títulos e resumos dos artigos de forma a aplicar os critérios de seleção definidos. O terceiro filtro foi efetuado a leitura completa dos estudos aceitos no segundo filtro e a aplicação dos critérios de seleção. Na figura 381, são apresentadas as etapas para a seleção dos artigos.

⁸ <https://www.mendeley.com>

Figura 381 – Aplicação dos filtros na condução do mapeamento



Os artigos que são o resultado final da aplicação dos filtros foram divididos em tabelas conforme o modelo de emoção utilizados. A Tabela 238 apresenta os artigos que adotaram o modelo tridimensional e os artigos que associaram o modelo categórico com bidimensional. A Tabela 239 Os artigos que adotaram o modelo categórico. A Tabela 240 apresenta os artigos que adotaram o modelo bidimensional.

Tabela 238 – Artigos que utilizaram o modelo tridimensional e a junção de modelos

Autor	Título
(WIECZORKOWSKA et al., 2005)	Creating Reliable Database for Experiments on Extracting Emotions from Music
(TROHIDIS et al., 2011)	Multi-label classification of music by emotion

Tabela 239 – Artigos que utilizaram o modelo categórico

Autor	Título
(ZHU et al., 2006)	An integrated music recommendation system
(CABREDO et al., 2012)	An Emotion Model for Music Using Brain Waves.
(SONG; DIXON; PEARCE, 2012)	Evaluation of Musical Features for Emotion Classification
(XING et al., 2013)	Human-centric music medical therapy exploration system
(ZHANG; SUN, 2013)	Web music emotion recognition based on higher effective gene expression programming
(VYAS; DUTTA, 2014)	Automatic mood detection of indian music using mfccs and k-means algorithm
(LIN et al., 2016)	Music Emotion Recognition Based on Two-Level Support Vector Classification

Tabela 240 – Artigos que utilizaram modelo bidimensional

Autor	Título
(SHI et al., 2006)	A tempo feature via modulation spectrum analysis and its application to music emotion classification
(CHENG et al., 2008)	Automatic Chord Recognition for Music Classification and Retrieval
Rho, Han e Hwang (2009)	SVR-based Music Mood Classification and Context-based Music Recommendation
(CHEN et al., 2010)	Affective understanding of online songs and speeches
(SINGH et al., 2012)	Architecture for Automated Tagging and Clustering of Song Files According to Mood
(SCHULLER; DORFNER; RIGOLL, 2010)	Determination of nonprototypical valence and arousal in popular music: Features and performances
(HAN et al., 2010)	Music emotion classification and context-based music recommendation
(YANG; CHEN, 2011a)	Prediction of the Distribution of Perceived Music Emotions Using Discrete Samples
(YANG; CHEN, 2011b)	Ranking-based emotion recognition for music organization and retrieval
(SCHULLER; WENINGER; DORFNER, 2011)	Multi-Modal Non-Prototypical Music Mood Analysis in Continuous Space: Reliability and Performances
(UJLAMBKAR; ATTAR, 2012)	Automatic mood classification model for Indian popular music
(BRINKER; DINOTHER; SKOWRONEK, 2012)	Expressed music mood classification compared with valence and arousal ratings
(EEROLA, 2012)	Modeling Listeners' Emotional Response to Music
(YEH et al., 2014)	Popular music representation: chorus detection & emotion recognition
(LIN; YANG; JUNG, 2014)	Fusion of electroencephalographic dynamics and musical contents for estimating emotional responses in music listening

(REN; WU; JANG, 2015)	Automatic Music Mood Classification Based on Timbre and Modulation Features
(XING et al., 2015)	Emotion-driven Chinese folk music-image retrieval based on DE-SVM
(ZHANG et al., 2016)	Bridge the semantic gap between pop music acoustic feature and emotion: Build an interpretable model
(BAI et al., 2016)	Dimensional music emotion recognition by valence-arousal regression
(CHO et al., 2016)	Music emotion recognition using chord progressions
(KOSTEK; PLEWA, 2016)	Rough sets applied to mood of music recognition
(GREKOW, 2017)	Audio features dedicated to the detection of arousal and valence in music recordings
(HU; YANG, 2017)	Cross-Dataset and Cross-Cultural Music Mood Prediction: A Case on Western and Chinese Pop Songs
(ZHANG et al., 2017)	Feature selection and feature learning in arousal dimension of music emotion by using shrinkage methods
(HSIAO; CHEN; LEE, 2017)	Methodology for stage lighting control based on music emotions
(BAI et al., 2017)	Music Emotions Recognition by Cognitive Classification Methodologies

34.3.2.3 Extração de Dados do Mapeamento Sistemático

Além das características musicais de baixo e médio nível, foram extraídas informações consideradas importantes. Essas informações servem para atender as questões de pesquisa aqui levantadas. Dentre os dados extraídos, estão:

1. O algoritmo usado para classificar
2. Acurácia
3. O modelo de emoção
4. A fonte de extração
5. Ferramentas de extração
6. Seleção de características
7. Técnicas para extração
8. Disponibilidade da base de dados

As informações referentes ao país aonde foi desenvolvido o estudo, assim como, a disponibilidade das bases de músicas, a fim de, referenciar bases de dados disponíveis publicamente foram selecionadas, uma vez que, por vezes, nichos específicos de músicas estão associados a determinadas culturas e, muitas vezes, a comunidade interessada em classificar músicas necessitam de bases de dados rotuladas. Desse modo, tais informações serão compiladas e disponibilizadas para amplo uso da comunidade científica.

Para identificar a emoção transmitida por uma música, em alguns casos os trabalhos aplicam métodos de seleção de características a fim de reduzir a dimensionalidade dos dados extraídos. Em seguida, as

informações são apresentados à um classificador que identifica o emoção da música. Desse modo, extraímos do estado da arte, métodos de seleção e algoritmos de classificação com sua respectiva acurácia a fim de oferecer uma base de referência para aqueles que desejem realizar classificação da emoção da música (QP1).

No que se diz respeito à fonte de extração de características, está relacionada à forma de apresentação do conteúdo (MIDI, partitura ou áudio). A partir daí, essa informação pode ser associada aos dados sobre as ferramentas de extração e suas configurações (*Sample rate, overlap, windows size.*) (QP2).

Existem diferentes tipos de modelos para relacionar a emoção da música, conforme discorrido na Seção 34.2.2. Pensando nisso, foi mapeada informação sobre o modelo de emoção abordado nos estudos selecionados e associação das emoções a características musicais. Esta informação é referente à forma de correlacionar música a uma emoção, ou ao um conjunto de emoções semelhantes (QP3).

Cada uma das informações usadas para responder as questões de pesquisas estão na Tabela 241.

Tabela 241 – Informações Extraídas dos artigos

Identificação	Dados extraídos
Informações Gerais	Disponibilidade da base de dados e Localização dos artigos.
QP1	Ferramentas para seleção de características, Classificadores, acurácia.
QP2	Fonte de extração, duração do áudio, ferramentas para extração de Características, características extraídas.
QP3	Modelo de emoção, características com valores emocionais.

34.4 Resultados

Foram encontrados 35 artigos que respondem às questões de pesquisas definidas e se distribuem nos anos de 2005 a 2017. De forma que, todos apresentam características musicais e classificam, ou reconhecem a emoção da música. Um total de 15 artigos apresentam ferramentas utilizadas para extração de característica. E apenas 08, apresentam um conjunto de características que representa emoção.

Os resultados foram distribuídos e detalhados nas subseções seguintes. A Seção 34.4.1 apresenta algumas informações gerais pertinentes a pesquisa. A Seção 34.4.2 apresenta os métodos utilizados na literatura para extração de características. A seção 34.4.3 discute as formas utilizadas para identificação da emoção musical; A Seção 34.4.4 apresenta um conjunto de características musicais, de forma a relacionar a emoções. E, por fim, é apresentada uma tabela com as características extraídas que foram levantadas nos artigos.

34.4.1 Informações Gerais

Nesta seção serão apresentados alguns dados levantados a partir do Mapeamento Sistemático, afim de, mostrar informações relevantes para estudos futuros. Dessa forma, a subseção 34.4.1.1, contém informações sobre a base de dados e a subseção 34.4.1.2, apresenta um mapa com a os locais aonde foi desenvolvido os estudos.

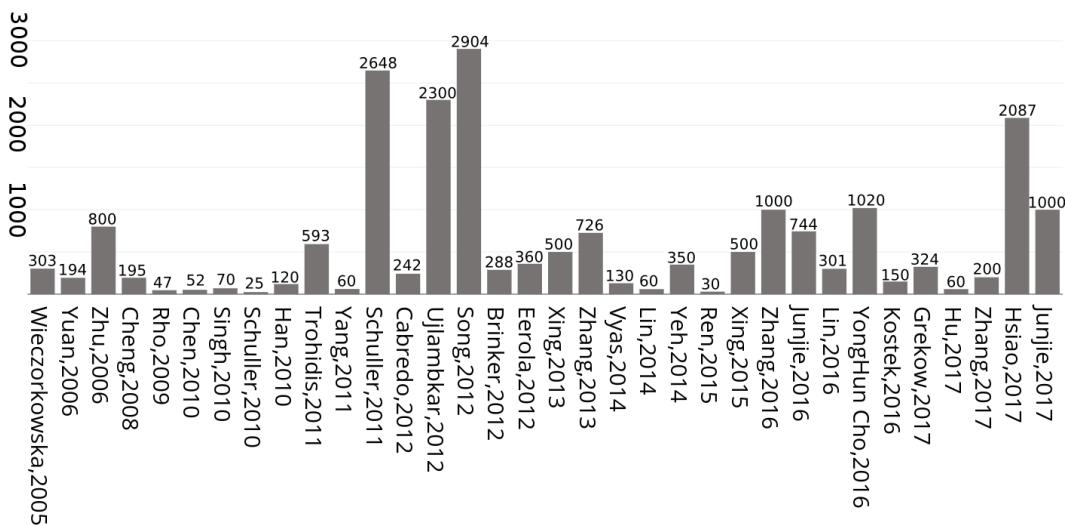
34.4.1.1 Base de Dados

Para analisar os métodos utilizados na classificação, é relevante associar a quantidade média de músicas usadas nas bases de dados. Nos artigos selecionados, apenas um trabalho disponibilizou publicamente a base. O trabalho de (SONG; DIXON; PEARCE, 2012), dispôs uma base de dados com um total de 2904 músicas, rotuladas através de *tags* sociais. Porém, ao acessar o *link* disponibilizado pelo autor, foi verificado

que a página havia sido removida. Em relação quantidade de músicas utilizadas nas bases, vão de 25 a 2904 músicas, como retratado na Figura 382.

As bases de dados apresentadas foram criadas manualmente pelos autores dos artigos. Um outro modo, também, apresentado é através das bases *online* que são rotuladas por *tags* sociais, como é o caso do *Last.fm* (HU; DOWNIE, 2007) e do SOUNDTRACK (EEROLA; LARTILLOT; TOIVIAINEN, 2009).

Figura 382 – Quantidade de músicas usadas em cada artigo

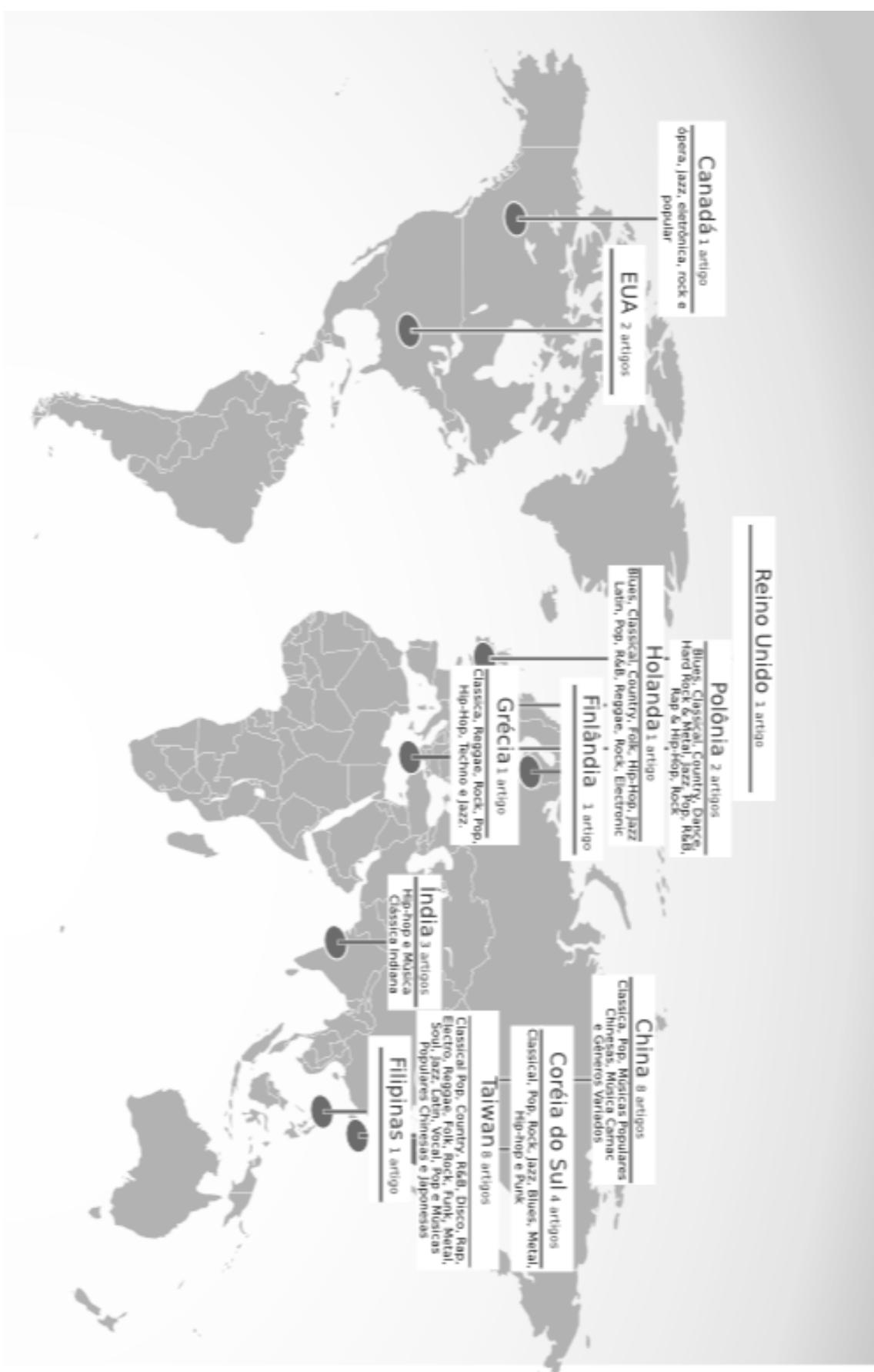


Uma das dificuldades relatadas na literatura e observada no presente trabalho é o conjunto de dados distintos criados por cada autor, por conta da falta de bases de dados públicas. O que afeta visivelmente na precisão dos resultados, de forma que, impede a comparação dos resultados entre os estudos.

34.4.1.2 Localização dos Artigos

Uma das teorias ainda discutidas entre os pesquisadores é a influência cultural sobre o estudos da música e emoção (para mais informações, Seção 34.2.1) Dessa forma, um dos dados extraídos nesta revisão é o país em que foi realizado a pesquisa. Juntamente com a informação do país, estão os gêneros musicais usados. Alguns dos artigos não evidenciaram os gêneros. Como pode ser observado na Figura 383, a maior parte dos estudos estão concentrados no oriente.

Figura 383 – Mapa dos países onde foi realizado o estudo



34.4.2 Métodos para Extração de Características

Entre os artigos, apenas dois artigos associaram o formato MIDI ao áudio, isso deve ao fato de que os autores, além de extraírem características de baixo nível, também extraíram o acorde, característica de médio nível. Apenas o artigo ([CABREDO et al., 2012](#)), utilizou somente o MIDI como fonte de extração.

Na etapa de pré-processamento, nos artigos foi considerado a duração do áudio. A maioria dos artigos consideraram entre 10 a 45 segundos iniciais para extração de características. Somente o trabalho de ([YEH et al., 2014](#)), extraiu os dados a partir da informação do refrão da música. O autor relatou que a música contém mais de uma emoção. A forma utilizada no trabalho para obter um maior número de informações emocionais, foi extraíndo as características diretamente do refrão.

A partir da seleção do tempo de duração do áudio, identificamos em 16 artigos, informações sobre a suas configurações. As informações são: janelamento, tamanho da janela, taxa de amostragem e a sobreposição. As técnicas de janelamento encontradas são *Hamming* e *Hanning*, nas quais, apresentaram a taxa de amostragem de 22,050 Hz e 44,1Hz em diferentes artigos. Os tamanhos das janelas utilizadas são 1024 e as sobreposições variam de 20 ms a 50 ms.

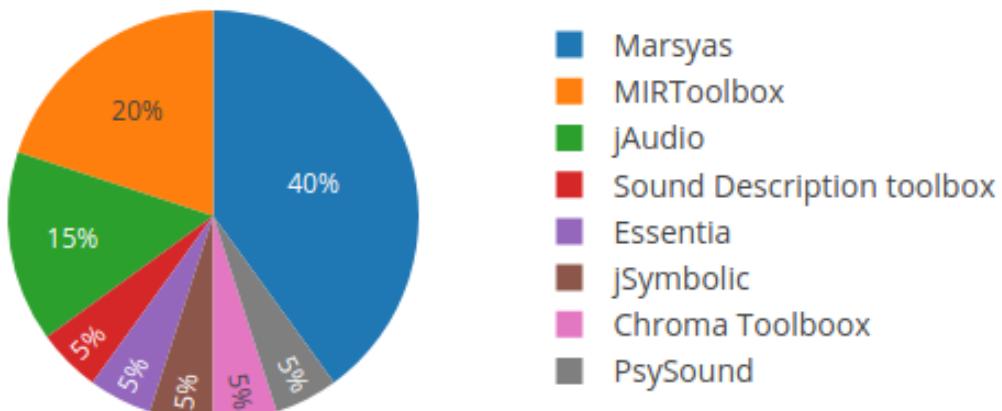
Apesar de ser uma informação essencial, poucos trabalhos citam os parâmetros que são fundamentais para extração de características. Um dos trabalhos encontradas na fase de condução deste mapeamento foi o ([YANG; DONG; LI, 2018](#)). Esse artigo consiste em uma revisão sistemática sobre os métodos de reconhecimento emocional a partir de dados, porém, também, não apresenta informações sobre os parâmetros de extração.

A partir dos trabalhos resultantes do mapeamento sistemático, nós verificamos que apenas 19 artigos informaram as ferramentas utilizadas na extração de características. Ainda, observamos que, em alguns casos, é necessário a utilização de mais de uma ferramenta para comparação. Como é o caso de ([BAI et al., 2016](#)), ([CHO et al., 2016](#)), ([HU; YANG, 2017](#)) e ([GREKOW, 2017](#)). As ferramentas utilizadas são apresentadas na Tabela 242. A ferramenta que se destacou com a frequência de uso nos artigos foi a *Marsya*, apresentado na Figura 384.

Tabela 242 – Ferramentas para extração de características

Ferramenta	Descrição
Essentia	Uma biblioteca C ++ de código aberto para análise de áudio e recuperação de informações de música (BOGDANOV et al., 2013).
jAudio	É um <i>framework</i> independente desenvolvida em Java com interface gráfica para usuário. Projetado para processamento em lote para saída no formato XML ou ARFF (MCKAY; FUJINAGA; DEPALLE, 2005).
jSymbolic	Biblioteca que extrai características de arquivos MIDI (MCKAY; FUJINAGA, 2006).
Marsyas	É uma <i>framework</i> independente de processamento de áudio. Inclui uma ferramenta de extração de recurso de baixo nível construída em C ++ (TZANETAKIS; COOK, 2000).
MIR Toolbox	API de processamento de áudio para extração <i>off-line</i> de recursos no Matlab. Inclui pré-processamento, classificação e funcionalidade de agrupamento (LARTILLOT; TOIVIAINEN, 2007).
PsySound Toolbox	Uma ferramenta extrai não apenas a medição física de sinais de áudio de música (por exemplo, tensão e amplitude), mas também percepções humanas de som (CABRERA et al., 1999).
Chroma Toolbox	uma ferramenta desenvolvida com foco em recursos de áudio baseados no tom (MÜLLER; EWERT, 2011).
Sound Description	extrai vários descritores padrão MPEG-7 como e outros conjuntos de recursos de arquivos de áudio WAV (BENETOS, 2006).

Figura 384 – Frequência de uso das ferramentas de extração nos artigos



34.4.3 Reconhecimento da Emoção

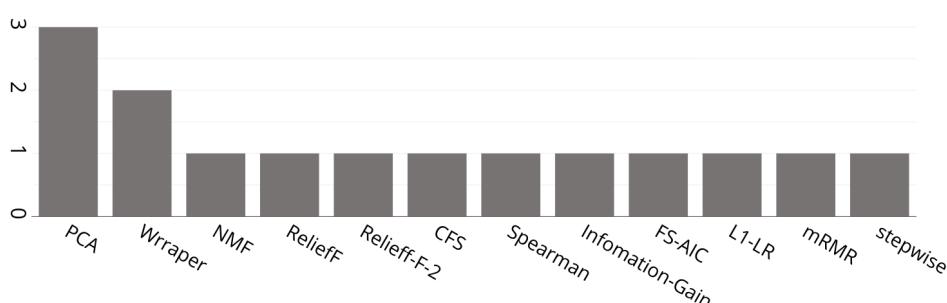
A principal questão de pesquisa, se refere à forma para identificar a emoção em uma música: **QPI** Como um sistema computacional consegue identificar a emoção transmitida por uma música? Nos artigos analisados, podem ser observadas algumas técnicas utilizadas para detecção automática de emoções. Os passos seguidos para a execução reconhecimento de emoção são descritos no fluxograma da Figura 385 e apresentado em detalhes no decorrer desta seção.

Figura 385 – Etapas para o Reconhecimento de Emoções



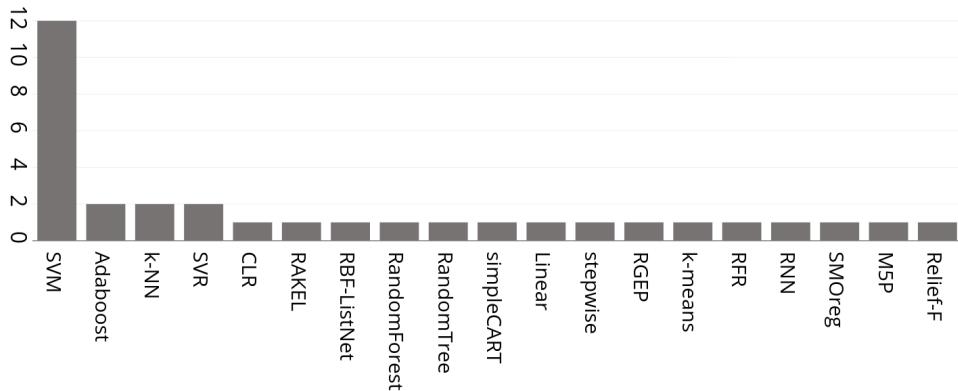
Com objetivo de selecionar as características que melhor representam a emoção na música e, ainda, conseguir reduzir a dimensionalidade dos dados, foram apresentados nos artigos os métodos para a seleção de características. Apesar de 11 artigos evidenciarem os métodos de seleção usados, de forma que, em alguns trabalhos foram utilizados mais de um método. Os métodos de seleção usados são: *Wrapper*, NMF, PCA, ReliefF, ReliefF-1, ReliefF-2, F-score, Correlation-based (CFS), Spearman, Information-gain, FS-AIC, L1-LR, mRMR, Stepwise. Os métodos de seleção PCA e Wrapper são os mais usados, como pode ser observado na Figura 386.

Figura 386 – Frequência de uso dos métodos de seleção



Após a seleção de características, a forma automática usada para reconhecimento e classificação de emoções é através de algoritmos de classificação e predição. Os algoritmos utilizados são: K-NN, Adaboost, SVR, SVM, CLR, RAKEL, RBF-ListNet, RandomForest, RandomTree, simpleCART, BP, Linear, RGEP, K-means, RFR, RNN, SMOreg, REPTree, M5P, FKNN, LDA. Podem ser observados na Figura 387. Entre os algoritmos, o mais utilizado nos estudos encontrados foi o SVM. Em relação a acurácia estabelecida pelos classificadores, foi identificada uma grande variação nas taxas apresentadas, que variam de 55,2% a 92,8%.

Figura 387 – Frequência de uso dos classificadores e preditores



O trabalho (CHEN et al., 2010), que atingiu a precisão de 55,2 % apresentou uma base com 52 músicas e utilizou o classificador SVM. O autor alegou que a baixa precisão é devido à confusão em relação à música de fundo e a interação do cantor. No trabalho é exemplificado esse fato com o estilo de música rap, que contém um ritmo de música lenta, porém a taxa de reconhecimento é degradada por que o cantor executa a música de uma maneira intensa.

No trabalho (SHI et al., 2006), foi alcançado a precisão de 92,8%. O autor utiliza uma base de dados com 194 músicas com os gêneros ópera, jazz, eletrônica, popular e rock. O classificador utilizado é o Adaboost. É proposto no estudo o uso do método de extração de características de tempo baseado na análise de espectro de modulação. E aplica o coeficiente de frequência de modulação em escala de *log* (CFML), que é um recurso eficiente e para caracterizar a dinâmica de longo prazo da música, na qual a frequência do tempo e seus harmônicos são apresentados. Dessa forma, eles alcançaram uma alta acurácia na classificação.

34.4.4 Relação das Características e a Emoção

No presente estudo, foram extraídas dos artigos 85 características musicais. Porém, foram encontradas limitações. Alguns estudos, não definem as características em seus estudos, forçando a busca na literatura conforme referência apresentada com a definição da característica neste trabalho, de forma que algumas descrições de características não foram encontradas. Além disso, alguns autores utilizam nomes diferentes para a mesma característica ou cunham novos termos, o que dificultou a associação do nome ao conceito.

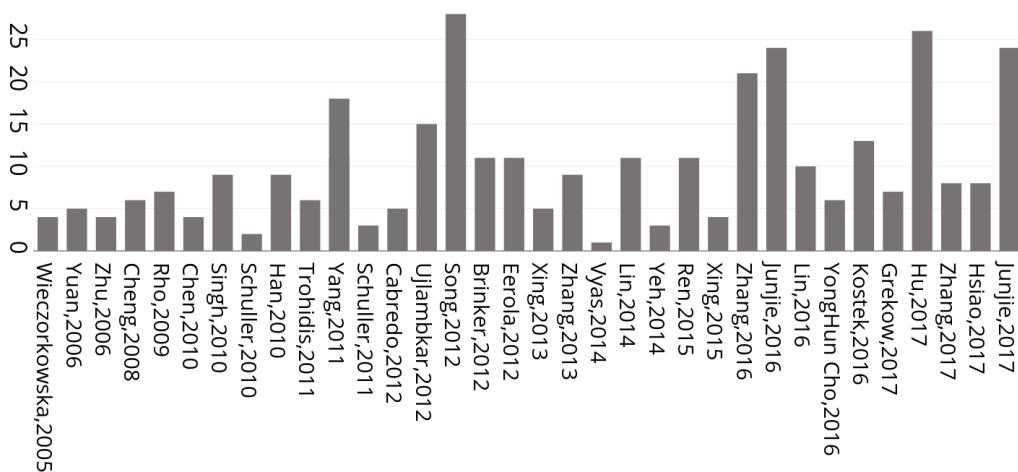
Como forma de organização e para maior compreensão do leitor, alguns autores categorizaram as características de duas maneiras. O primeiro modo corresponde a categorização a partir do elemento principal de extração. Como observado nos artigo (YANG; CHEN, 2011a), (SCHULLER; WENINGER; DORFNER, 2011), (EEROLA, 2012), (CABREDO et al., 2012), (SONG; DIXON; PEARCE, 2012), (BRINKER; DINOTHER; SKOWRONEK, 2012), (LIN; YANG; JUNG, 2014), (LIN et al., 2016), (HSIAO; CHEN; LEE, 2017), (ZHANG et al., 2017), (GREKOW, 2017), (HAN et al., 2010), (HU; YANG, 2017). Enquanto que em (ZHANG et al., 2016), as características são representadas em categorias por níveis de extração, que são: físico, perceptivo, musical e semântico.

A forma de categorização das características utilizada neste trabalho corresponde a primeira forma apresentada, por ser a mais utilizada, além de proporcionar organização. A Tabela 245, apresenta as características extraídas do timbre. A Tabela 246, apresenta as características extraídas do ritmo. A Tabela

[247](#), apresenta a característica extraídas da energia. A Tabela [248](#), apresenta as características extraídas da tonalidade. A Tabela [249](#), apresenta as características extraídas do tom. A Tabela [250](#) apresenta as características extraídas do acorde. A Tabela [251](#), apresenta as características extraídas da harmonia. A Tabela [252](#), apresenta as características extraídas da intensidade. A Tabela [253](#), apresenta as características extraídas da dinâmica. E por fim, a Tabela [254](#), apresenta as características extraídas da articulação.

As características obtidas pelo Mapeamento se dividem em médio e baixo nível. A maioria dos estudos extraíram características de timbre e ritmo. As características de Timbre que mais foram repetidas são: Coeficiente Cepstrais de Alta Frequência, Rollof, Fluxo Espectral e Centroide Espectral. Enquanto que, o tempo foi a característica que mais se repetiu na classe ritmo. A Figura [388](#), apresenta a quantidade de características extraídas de cada artigo.

Figura 388 – Quantidade de características extraídas por artigo



Para responder a questão de pesquisa **QP3 Quais características musicais devem ser extraídas para determinar a emoção da música?**, foram selecionados artigos que apresentam extração de características musicais a fim de caracterizar uma determinada emoção. Porém, apenas 08 trabalhos selecionados utilizam a acurácia de classificação de um conjunto de características de áudio para reconhecer a emoção da música. Todos utilizam o modelo bidimensional e apenas 07 fornecem como resultado características que representam valência e excitação, conforme apresentados na Tabela [243](#). No artigo de ([SCHULLER; DORFNER; RIGOLL, 2010](#)), apesar de usar o modelo dimensional, não especifica as características associadas, mas apresenta uma tabela de acordes de associando a emoções discretas, Tabela [244](#).

Tabela 243 – Associação de características de áudio a emoção

Artigo	Valência	Excitação
(LIN; YANG; JUNG, 2014)	Dissonância Espectral e o Modo	Nitidez, Fluxo Harmônico e Frequência Mel do Coeficiente Cepstral.
(ZHANG et al., 2016)	Não especificado	Baixo SONE, Raiz Média Quadrada e Fluxo de Sonoridade.
(CHENG et al., 2008)	Acorde, Espectro de Nivelamento, Frequência Mel do Coeficiente Cepstral, Espectro de Crista.	volume e Tempo.
(HU; YANG, 2017)	Volume, Regularidade, Irregularidade, Taxa de Cruzamento Zero, Centroide Espectral, Flux Espectral, Brilho Espectral, Nivelamento Espectral, Rollof.	Força do Ritmo, Clareza do Pulso, Média do Tempo, Regularidade do Ritmo, Função de Detecção de Alteração Harmônica, Modo, Clareza da Chave, Tom, Cromá.
(LIN et al., 2016)	Modo e Harmonia	Tempo, Tom, Volume, e Timbre.
(BRINKER; DINOTHER; SKOWRO-NEK, 2012)	Tempo lento, Volume, Excentricidade do Cromá, Ritmo rápido e Inclinação Espectral.	Croma, Variabilidade de percussão entre bandas, Medida de relação entre tempos rápidos e lentos e Espectro de Modulação Harmônico.
(EEROLA, 2012)	Tempo de Ataque	Clareza de Pulso

Tabela 244 – Tipos de acordes e as emoções associadas

Tipo de Acorde	Emoções associadas
Maior	Felicidade, alegria, confiança, satisfação, radiância
Menor	Tristeza, escuridão, mau humor, apreensão, melancolia, depressão, mistério
Sétimo	Natural, moderado, inquieto
Sétimo Maior	Romance, suavidade, serenidade, satisfação, tranquilidade
Sétima Menor	Tranquilidade, melancolia
Nono	Abertura, otimismo
Diminuto	Medo, choque, susto, suspense
Quarto Suspenso	Tensão agradável
Sétima, Nona Menor	Sinistro, medo, escuridão
Nono Adicionado	Inflexível, severo

Adaptado de Schuller, Dorfner e Rigoll (2010)

Tabela 245 – Descrição das características extraídas do timbre

Id	Característica	Descrição
01	Amplitude Espectral	É a medida de quanto longe, e em que direção, essa variável difere de zero (LYONS, 2011).
02	Aspereza Espectral	É obtido computando os picos do espectro e tomando a média de toda a dissonância entre todos os possíveis pares de picos (SETHARES, 1998).
03	Atenuação de Espectro	Uma técnica que consiste nas etapas de estimativa do espectro de ruído e a estimativa da fala (KIM et al., 2003).
04	Baricentro Temporal	É a média temporal da energia envelop (YANG; CHEN, 2011a).
05	Brilho Espectral	É calculado como a centroide do espectro de magnitude de Fourier de curta duração, novamente armazenado como uma frequência de log. É uma medida do conteúdo de frequência mais alta do sinal.
06	Centroide Espectral	Indica o brilho ou a nitidez de um som e caracteriza o centro de gravidade do espectros (HAN et al., 2010).
07	Coeficientes de Transformação de Cosseno Discreto Modificado	É o componente básico de processamento para compressão de áudio de alta qualidade nos padrões internacionais de codificação (BRITANAK; RAO, 2001).
08	Contraste Espectral	Considera a intensidade dos picos espetrais e dos vales espetrais em cada sub-banda separadamente, de modo que possa representar as características espetrais relativas e refletir aproximadamente a distribuição de componentes harmônicos e não harmônicos (JIANG et al., 2002).
09	Crista Espectral	Fornece uma medida para quantificar a tonalidade do sinal. Esta medida é útil para discriminar entre sinais de banda larga e banda estreita indicando o pico relativo de uma sub-banda (HOSSEINZADEH; KRISHNAN, 2007).
10	Curtose Espectral	O fluxo espectral é definido como a diferença quadrática entre as magnitudes normalizadas de janelas sucessivas (ANTONI, 2006).
11	Dissonância Espectral	Mede a dureza ou aspereza do espectro acústico (CABRERA et al., 1999).
12	Distorção Espectral	Descreve a assimetria da distribuição de frequência ao redor do centroide espectral (GOUYON; HERRERA, 2003).
13	Energia Espectral	É a energia de cada sub-banda normalizada com a energia combinada do espectro (HOSSEINZADEH; KRISHNAN, 2007).
14	Entropia Espectral	Pode ser usada para quantificar o pico de uma distribuição (MISRA et al., 2004).
15	Espectro	indica se a distribuição é suave ou pontiaguda e resulta da relação simples entre a média geométrica e a média aritmética (LARTILLOT, 2011)
16	Espectro de Força	Fornece um gráfico da porção da potência de um sinal (energia por unidade de tempo) pendendo para o interior de determinadas posições de frequência.
17	Espectro de Magnitude	É a medida de como, independentemente da direção, sua quantidade difere de zero. Então, as magnitudes são sempre valores positivos (LYONS, 2011).
18	Estrutura Novidade	O grau de repetição temporal de qualquer característica particular, como espectro ou cromato ao longo do tempo com base na detecção de bordas dentro da diagonal da matriz de auto-similaridade (FOOTE; COOPER, 2003).
19	Fluxo Espectral	É definido como a diferença quadrática entre as magnitudes normalizadas de janelas sucessivas (PORIA et al., 2016).
20	Frequência de Início	Refere-se à detecção do início de eventos discretos em sinais acústicos. Uma percepção de um início é causado por uma evidente mudança na intensidade, timbre do som.
21	Frequência mais Forte via Cruzamento Zero	Uma estimativa do componente de frequência mais forte de um sinal, em Hz, encontrado através do número de cruzamentos zero (MCKAY; FUJINAGA; DEPALLE, 2005).

22	Frequência Mel do Coeficiente Cepstral	Caracteriza a forma espectral do som tomando os coeficientes da transformada de cosseno discreta dos espectros de <i>log-power</i> expressos em uma escala de frequência Mel relacionada com a percepção linear.
23	Histograma Coeficiente Wavelets	Representam informações locais e globais através do cálculo de histogramas nos Coeficientes <i>Wavelet Daubechies</i> em diferentes sub-bandas de frequência com diferentes resoluções.
24	Inarmonicidade	Estima a quantidade de conotações que se desviam dos múltiplos da frequência fundamental (LIN et al., 2016).
25	Irregularidade	É o grau de variação dos picos sucessivos do espectro (LARTILLOT, 2011)
26	Largura de Banda	A largura de banda é calculada como a média ponderada da magnitude das diferenças entre os componentes espetrais e o centroíde (LARSEN; AARTS, 2005).
27	Largura Timbral	É definida como a largura do pico, a intensidade da distribuição espectral, enquanto que o volume é derivado com base na força relativa entre a intensidade total e nitidez.
28	Média do Início	A média do início a cada pico do áudio (YANG; CHEN, 2011b).
29	Nivelamento Espectral	indica se a distribuição é suave ou pontiaguda e resulta da relação simples entre a média geométrica e a média aritmética (LARTILLOT, 2011).
30	Percussão	Caracterização e classificação de <i>onsets</i> por banda (BRINKER; DINOTHER; SKOWRONEK, 2012).
31	Pico Espectral	Representa o componente harmônico (REN; WU; JANG, 2015).
32	Propagação Espectral	Indica o espelhamento médio do espectro e está relacionado ao seu próprio Centroíde Espectral (HAN et al., 2010).
33	Suavidade Espectral	Se refere à expectativa de que os envelopes espetrais de sons reais tendem a variar lentamente em funções da frequência (KLAPURI, 2003).
34	Taxa de Cruzamento Zero	Mede o ruído do sinal, é calculado pela contagem o número de valores de sinal que cruzam o eixo zero em cada quadro (YANG; CHEN, 2011a).
35	Timbre	É conhecido como qualidade de som da psicoacústica, o que faz com que dois sons com o mesmo tom e volume soem diferentes (ZHANG et al., 2016).
36	Vale Espectral	Corresponde ao componente não harmônico ou ruído no espectro (REN; WU; JANG, 2015).
37	Variabilidade Espectral	Consiste no desvio padrão do espectro de magnitude. Uma medida de quão variada é o espectro de magnitude de um sinal (MCKAY; FUJINAGA; DEPALLE, 2005).

Tabela 246 – Descrição de características extraídas do ritmo

Id	Característica	Descrição
01	Batida	Representa uma duração periódica de 1/4 de nota, enquanto que o andamento é geralmente definido como as batidas por minuto para representar o recurso rítmico (LIN et al., 2016).
02	Batida mais Forte	É definido como a batida mais forte em um sinal, em Batidas Por Minuto (BPM) e é encontrado localizando a caixa mais forte no Histograma de Batida (PORIA et al., 2016).
03	Clareza do pulso	Quão claro e estável é o pulso ou batida na música, também chamado de força da batida (LARTILLOT et al., 2008).
04	Clareza do Ritmo	Autocorrelação da curva de detecção de início (HU; YANG, 2017).
05	Densidade	Estima a frequência média de eventos, ou seja, o número de eventos detectados por segundo (LARTILLOT, 2011).
06	Histograma de Batida	É uma curva que descreve a força da batida como uma função de um intervalo de valores de andamento e permite a extração das propriedades mencionadas (BURRED; LERCH, 2003).
07	Histograma de Ritmo	Agrega os valores de amplitude de modulação das bandas críticas individuais calculadas em um padrões de ritmo e é, portanto, um descritor de menor dimensão para características rítmicas gerais em uma peça de áudio (LIDY et al., 2007).
08	inter-onset	Forma múltiplos agentes de batimento (com tempo e fase variados) que competem com base em quão bem cada um pode prever localizações de batidas (DAVIES; PLUMBLEY, 2007).
09	Média do Espectro de Ba-tida	É uma medida de auto-similaridade acústica versus tempo de latência, calculado a partir de uma representação de similaridade espectral (FOOTE; UCHIHASHI, 2001).
10	Relação da Sonoridade da Batida por Banda	A intensidade do sinal nas janelas centrada em torno dos locais de batida (BOGDANOV et al., 2013).
11	Ritmo	É um importante traço musical que representa o andamento ou o pulso de uma peça musical (HU; YANG, 2017).
12	Regularidade de Ritmo	Um histograma de batida no qual os picos são espaçados periodicamente (BURRED; LERCH, 2003).
13	Soma das Batidas	A soma de todos os valores no histograma de batida (MCKAY; FUJINAGA; DEPALLE, 2005).
14	Tempo	Geralmente definido como as batidas por minuto (BPM) e é usado para representar característica rítmica da música (RHO; HAN; HWANG, 2009).
15	Tempo Médio	O divisor comum dos picos da curva de autocorrelação (LIU; LU; ZHANG, 2003).
16	Variação do Comprimento da Batida	Analisa a variação do ritmo para identificar a diversidade de emoções (HAN et al., 2010).

Tabela 247 – Descrição da característica extraída da energia

Id	Característica	Descrição
01	Poder do Áudio	Descreve a potência instantânea temporariamente suavizado sobre cada quadro (SMITH; CHENG; BURNETT, 2010).

Tabela 248 – Descrição de características extraídas da tonalidade

Id	Característica	Descrição
01	Tonalidade	É usado, principalmente, para denotar um sistema de relações entre uma série de passos (formando melodias e harmonias) tendo um tônico ou tom central (GUTIÉRREZ et al., 2006).

Tabela 249 – Descrição de características extraídas do tom

Id	Característica	Descrição
01	Chave	Oferece uma estimativa ampla das posições dos centros tonais e sua respectiva clareza (LARTILLOT, 2011).
02	Cromagrama	É calculado no espectrograma log-scaled para representar as frequências em escalas musicais (LIN et al., 2016).
03	Histograma de Tom	São definidos e propostos como uma forma de representar o conteúdo do tom dos sinais musicais, tanto na forma simbólica quanto na forma de áudio (TZANETAKIS; ERMOLINSKIY; COOK, 2002).
04	Modo	É um tipo de escala na música que implica em maior e menor para estimar a modalidade baseada em componentes de Croma (LIN et al., 2016).
05	Modo da chave	É definido como chave maior e chave menor (GUTIÉRREZ et al., 2006).
06	Tom	É o atributo auditivo do som que pode ser solicitado em uma escala de baixa a alta (HU; YANG, 2017).
07	Tom Saliente	O tom típico em Hz ou semitonos estabelecido por base de métodos cromáticos (BARTSCH; WAKEFIELD, 2005).

Tabela 250 – Descrição de características extraídas do acorde

Id	Característica	Descrição
01	Acorde	O acorde consiste em harmonia formada por múltiplas notas ou arremessos (CHENG et al., 2008).
02	Força da chave	Encontra a força-chave calculando os coeficientes de correlação entre o cromatograma retornado pelo cromagrama, empacotado e normalizado e perfis semelhantes representando todos os possíveis candidatos a tonalidade (LARTILLOT; TOIVIAINEN, 2007).
03	Histograma de Acordes	Esboça uma imagem da estrutura harmônica de uma música (CHENG et al., 2008).
04	Perfil de Classe de Passo Harmônico	É um vetor mede que a intensidade de cada um dos doze semitonos da escala diatônica, e é obtido mapeando cada intervalo de frequência do espectro para uma dada classe de notas (GUTIÉRREZ et al., 2006).
05	Subsequência de Acordes Comum mais Longa	Indica a similaridade de acordes em ordem através de um algoritmo de programação dinâmica que foi usado em bioinformática para comparação de sequência de DNA (CHENG et al., 2008).

Tabela 251 – Descrição de características extraídas da harmonia

Id	Característica	Descrição
01	Centro de massa do Croma	Calcula o vetor centroide tonal de 6 dimensões a partir do cromatismo. Corresponde a uma projeção das cordas ao longo de círculos de quintas, de terços menores e de terços maiores (HARTE; SANDLER; GASSER, 2006).
02	Escala	É uma regra geral da formação tônica da música. No artigo referido, foi definido como um conjunto de chave, modo e tonalidade (RHO; HAN; HWANG, 2009).
03	Fluxo harmônico	Indica diferença no conteúdo harmônico entre quadros consecutivos, como alterações de acorde, melodia forte ou movimento da linha de baixo (LIN; YANG; JUNG, 2014).
04	Harmonia	Refere-se ao relacionamento entre dois ou mais arremessos simultâneos na música (HU; YANG, 2017).
05	Melodia	uma equação de arremessos, os mais detectados em regiões de média e alta frequência, em oposição à linha de baixo, que pode ser encontrada em bandas de baixa frequência (GÓMEZ; Klapuri; MEUDIC, 2003).

Tabela 252 – Descrição de características extraídas da intensidade

Id	Característica	Descrição
01	baixo-SONE	A relação entre as dez primeiras faixas sonoras de banda crítica e a intensidade total (ZHANG et al., 2017).
02	Centro de Massa da Sonoridade	Centro de massa do volume/intensidade (ZHANG et al., 2017).
03	Energia Baixa	É a porcentagem de sinal com um nível abaixo da média (INTROINI; PRESTI; BOCCIGNONE, 2016).
04	Fluxo da sonoridade	Quantidade de mudança de volume (ZHANG et al., 2017).
05	Intensidade	É a energia média da sequência global das ondas (RHO; HAN; HWANG, 2009).
06	Intensidade Sonora	É a intensidade perceptiva do som (ZHANG et al., 2017).
07	Média do volume	Média do volume total (ZHANG et al., 2017).
08	Volume	É uma parte integrante da distribuição espectral da sensação de intensidade (LIN; YANG; JUNG, 2014).

Tabela 253 – Descrição de características extraídas da dinâmica

Id	Característica	Descrição
01	Dinâmica	Está relacionado ao intervalo dinâmico e à quantidade de flutuação no volume presente em uma gravação (STREICH et al., 2006).
02	Inclinação Espectral	Representa a quantidade da amplitude espectral. É calculado pela regressão linear (GUNASEKARAN; REVATHY, 2011).
03	Raiz Média Quadrada	Estima o recurso de energia de acordo com o volume de uma sílaba (EEROLA, 2012).

Tabela 254 – Descrição de características extraídas da articulação

Id	Característica	Descrição
01	Articulação	É baseada na relação média de silêncio, indicando a porcentagem de quadros que têm uma energia RMS significativamente menor do que a energia RMS média de todos os quadros (LARTILLOT, 2014).
02	Tempo de Ataque	Logaritmo de duração entre a hora em que o sinal começa e a hora em que o sinal atinge o seu valor estável. (EEROLA, 2012)

34.5 Conclusão

Identificamos na literatura que é possível reconhecer a emoção através da música. Porém, a tarefa de detectar automaticamente a emoção da música é bastante desafiadora por diversos motivos:

- I. A emoção percebida por um indivíduo enquanto escuta uma música é algo subjetivo e difícil de mensurar, por isso, os trabalhos selecionados se detêm às características de baixo e médio níveis;
- II. Muitas vezes uma mesma música apresenta mais de um sentimento ([YEH et al., 2014](#)), podendo ser eles antagônicos, o que leva os trabalhos à realizarem a tarefa de classificação em um trecho específico da música;
- III. Ausência de bases públicas rotuladas, os trabalhos confeccionaram sua própria base, com tamanho arbitrário e músicas de complexidade e gêneros distintos. Nenhum trabalho disponibilizou a base utilizada publicamente, o que previne o treinamento de classificadores, a reproduzibilidade dessas pesquisas e a comparação dos resultados alcançados por diferentes técnicas;
- IV. Grande volume de características, uma diversidade de características podem ser utilizadas para extrair informações da música e alcançar sua classificação. Neste, trabalho elencamos 85 características de baixo e médio nível a partir dos trabalhos selecionados. Muitas vezes, é necessário um grande conjunto de ferramentas com diferentes parâmetros de configuração para alcançar uma maior variedade na extração de características;

V. área interdisciplinar, muitas vezes para conduzir uma pesquisa neste ramo é preciso de conhecimento de computação aliado a teoria musical e, dependendo, do nível das características que se deseja extrair, é necessário conhecimento na área de psicologia. Tais razões se refletem diretamente na taxa de acurácia dos classificadores que assume um valor médio de 78,97%. Mesmo quando essa classificação ocorre, existe uma dificuldade em estabelecer uma relação direta entre as características da música em uma emoção específica, por exemplo, a tristeza. A contribuição principal a relação indireta das emoções com as características, no uso do modelo bidimensional. De forma que, trazem resultados de forma geral para valência e excitação.

Esse cenário indica que este é um campo com espaço para melhorias e proposição de novos fluxos de classificação e para contribuições no campo da música e psicologia.

Identificamos que o caminho tomado por grande parte da literatura para detecção da emoção da música se baseia na extração de características, seguindo um fluxo clássico de processamento de sinais, composto pelas etapas de (i) pré-processamento, (ii) extração de características, (iii) seleção de características e (iv) classificação da emoção da música. A principal fonte de extração de característica apresentada nos dados extraídos dos artigos é o áudio. Entre os artigos, 34 utilizaram áudio pelo fato de extraírem características de baixo nível que são mais fáceis de extrair, porém, difíceis de associar diretamente com a emoção. Apesar de a maioria dos estudos usarem o áudio, poucos se dedicaram a etapa de pré-processamento do sinal e, muitas, vezes informações importantes para a reprodução desta etapa não são disponibilizadas. Este contexto, abre espaço para investigação de técnicas de pré-processamento do sinal de áudio que podem impactar positivamente na acurácia do classificador.

Mineração de dados espaço-temporais do Twitter no contexto de desastres naturais

Autores: Ana Caroline de Almeida Alves; Deborah Maria Vieira Magalhães.

Resumo: A análise do comportamento de multidões no cenário de catástrofes naturais, exerce um papel importante no gerenciamento de desastres, bem como no planejamento da evacuação. Melhores estratégias de gerenciamento de desastres e planejamento da evacuação, podem ser desenvolvidas, baseadas no comportamento apresentado pela multidão. Nesse contexto, as mídias sociais assumem um papel importante, pois fornecem dados espaço-temporais em tempo real para orientar políticas de evacuação. A extração de dados espaço-temporais a partir de mídias sociais, tais como, hora, data e localização, é uma tarefa complexa. Além disso, um grande volume de dados não estruturados precisa ser filtrado e estruturado de modo a comunicar informação útil. Nesse contexto, será construída uma base de dados proveniente da Interface de programação de aplicações (API) do Twitter, a fim de identificar um padrão de localização da multidão durante o desastre natural, sendo utilizado como estudo de caso o terremoto e erupção do vulcão Kilauea no Havaí. Assim, este trabalho visa formalizar um padrão de evacuação e fornecer diretrizes afim de auxiliar no processo de evacuação em situações de desastres naturais.

Palavras-chaves: desastre natural, Havaí, mineração de dados, dados espaço-temporais.

35.1 Introdução

É notável que as mídias sociais tem se popularizado na vida das pessoas, tendo alterado a forma como as elas se comunicam. A mídia social tem sido reconhecida como um dos principais canais de comunicação em situações de crises e desastres (ANBALAGAN; VALLIYAMMAI, 2016a). O uso das redes sociais vem crescendo continuamente e, com isso, é criado um enorme volume de dados constantemente. O Facebook¹ tem mais de 750 milhões de usuários ativos, onde um usuário publica em média 90 postagens de conteúdo por mês. Os usuários do Twitter² enviam cerca de 140 milhões de tweets por dia³. Ao contrário da mídia tradicional,

¹ <https://www.facebook.com/>

² <https://twitter.com/>

³ <https://www-03.ibm.com/press/us/en/pressrelease/35633.wss>

essas plataformas permitem a coleta de dados em grande escala, documentando a reação pública aos eventos que se desdobram nos mundos virtual e físico (KRYVASHEYEU et al., 2016).

As redes sociais na gestão de desastres têm atraído grande interesse público e da comunidade científica. Em particular, a plataforma de *microblogging* Twitter tem sido especialmente útil durante eventos de emergência (KRYVASHEYEU et al., 2016). Nas mídias sociais são utilizados serviços baseados em geolocalização, onde as pessoas criam dados com data, horário e localização, fornecendo dados espaço-temporais. Tais dados podem ser aproveitados no monitoramento e avaliação de desastres. Desse modo, a extração de dados espaço-temporais no cenário de desastres naturais pode ser útil na análise do percurso, como o local onde as pessoas se deslocam e como elas respondem aos desastres. Assim, é permitido a formação de diretrizes para o auxílio no desenvolvimento de políticas públicas de evacuação.

Para a gestão de emergências e desastres, a análise do comportamento, como a forma que as pessoas se preparam e respondem aos desastres, é importante para o planejamento da evacuação (CHAE et al., 2014). Políticas de planejamento de evacuação são uma das principais medidas tomadas para evitar perdas de recursos humanos durante desastres naturais. A evacuação quando bem planejada, exerce um papel importante na diminuição de vulnerabilidades e na busca por melhores rotas de fugas, podendo assim poupar vidas.

De acordo com a ONU, em 2018, os desastres climáticos e geofísicos, como terremotos e tsunamis, mataram 1,3 milhão de pessoas nos últimos 20 anos e deixaram mais 4,4 bilhões de feridos, desabrigados ou em necessidade de ajuda de emergência⁴. Desse modo, utilizar as mídias sociais como forma de auxiliar as pessoas no momento da evacuação desponta como alternativa. Chae et al. (2014) afirmam que a análise do comportamento público, por exemplo, como as pessoas se preparam e respondem a desastres, desempenha um papel importante na resposta à desastres e planejamento de evacuação.

O tratamento de dados é uma tarefa complexa, pois, será necessário formalizar um padrão de evacuação, através de métodos de filtragem. Para Compieta et al. (2007) os conjuntos de dados espaço-temporais costumam ser muito grandes e difíceis de analisar e exibir. Pois é necessário passar por processos de mineração e limpeza de dados. Além disso, Chae et al. (2014) afirma que a exploração de tempo e dados geográficos referenciados são uma questão desafiadora, inclusive pelo fato da dificuldade de adquirir bases de dados georreferenciadas.

A fim de endereçar tal desafio, neste trabalho, propomos a construção de uma base de dados espaço-temporais. Nessa base, são utilizados dados provenientes de tweets durante a ocorrência de um terremoto no Havaí em maio de 2018. Tal terremoto ocorreu próximo à uma região vulcânica o que desencadeou a erupção do vulcão Kilauea. Após a aquisição dos dados, estes foram agrupados e passaram por um processo de filtragem. Onde, em seguida, as rotas de evacuação foram traçadas para auxiliar na construção de diretrizes a fim de ajudar em outros eventos de desastres que venham a ocorrer nessa região e no mundo. A Figura 1 descreve cada passo da metodologia adotada, com a finalidade de alcançar o objetivo proposto. O passo 1 representa a revisão da literatura que foi realizada a fim de selecionar os artigos de maior contribuição para este trabalho. A segunda etapa consiste em fazer uma síntese dos trabalhos selecionados e abstrair os melhores conceitos afim de serem postos em prática. Na fase de desenvolvimento foram aplicadas todas as técnicas necessárias para obtenção dos resultados deste trabalho. Por fim, foram formuladas diretrizes afim de ajudar no processo de evacuação em situação de desastre. As diretrizes foram formuladas com base nos resultados obtidos nas etapas anteriores.

Figura 389 – Etapas de contribuição do trabalho.



⁴ <https://nacoesunidas.org/relatorio-da-onu-alerta-para-aumento-dramatico-das-perdas-economicas-provocadas-por-desastres/>

35.1.1 Objetivos

O objetivo geral deste trabalho reside em extraír padrões de comportamento da multidão, através de dados espaço-temporais coletados a partir de eventos de desastre. Além de construir uma base de dados que auxiliam cientistas, analistas, defesa civil e comunidades afetadas por desastres. Os objetivos específicos deste trabalho são:

1. Formalizar um padrão de evacuação e sugerir rotas de fuga que auxiliem na evacuação da multidão;
2. Desenvolver um conjunto de diretrizes para orientar políticas públicas baseados nos padrões extraídos dos dados coletados durante o desastre.

35.2 Referencial Teórico

Esta seção aborda tópicos relevantes ao contexto deste trabalho: desastres naturais, mídias sociais, dados espaço-temporais e, por fim, políticas públicas de evacuação em cenários de desastre.

35.2.1 Desastres naturais

Nos últimos anos vem ocorrendo desastres por causas naturais com maior frequência devido às mudanças climáticas. Portanto, furacões, inundações e tornados, provavelmente se tornarão mais comuns e intensas e, consequentemente, mais dispendiosas. Tais desastres promovem perda de recursos humanos, naturais e financeiros, levando as pessoas a situações de extrema calamidade. Os desastres naturais levam 24 milhões de pessoas por ano à pobreza⁵. Martín, Li e Cutter (2017) afirmam que catástrofes naturais são custosas em termos de propriedade, estabilidade política e vidas perdidas.

Vários trabalhos estão sendo realizados neste sentido, segundo Kuhnen (2009) tais trabalhos envolvem infra-estruturas mais fortes, edificações mais seguras e melhoria do sistema de aviso. Na área da computação, também há pesquisas sendo desenvolvidas para buscar meios de auxílio em situações desastrosas, especialmente por meio das mídias sociais, amplamente utilizada para fins de comunicação. Martín, Li e Cutter (2017) afirmam que o conceito de informática de crise tem sido usado na literatura para se referir a dados e informações sobre a resposta de emergência dos respondentes oficiais e do público.

De acordo com Kryvasheyev et al. (2016) durante eventos catastróficos, as pessoas se voltam para plataformas de redes sociais para comunicação. Chae et al. (2014) certificam que a análise do comportamento público, como a forma como as pessoas se preparam e respondem às catástrofes, desempenha um papel importante no gerenciamento de crises, na resposta a desastres e no planejamento de evacuação. Nos tópicos subsequentes serão explorados as formas de auxílio no contexto de desastres naturais através das mídias sociais e como essas informações podem colaborar, utilizando conceitos de mineração de dados.

Segundo MacEachren et al. (2011) as mídias sociais têm sido reconhecidas por muitos na pesquisa de gestão de crises e nas comunidades de prática como uma ferramenta para comunicação e coleta de informações.

35.2.2 Análise de dados espaço-temporais

As mídias sociais possibilitam interações sociais, tornando-as um dos principais meios de comunicação entre as pessoas. A cada interação novos dados estão sendo criados, sendo gerado um enorme volume dados constantemente. Todos os dias criamos 2,5 quintilhões de bytes de dados, tanto que 90% dos dados do mundo, foram criados apenas nos últimos dois anos⁶.

Para Avvenuti et al. (2014) as mídias sociais representam uma maneira poderosa de investigar preferências, gostos e atividades de grupos de usuários. Atualmente as pessoas compartilham continuamente comentários e conteúdo multimídia sobre suas vidas, interesses, sentimentos e opiniões. Portanto os

⁵ <https://nacoesunidas.org/artigo-genero-gestao-de-desastres-e-mudanca/>

⁶ <https://www-03.ibm.com/press/us/en/pressrelease/35633.wss>

usuários de redes sociais podem ser considerados como sensores capazes de transmitir informações valiosas sobre situações e fatos, como afirmado pelo paradigma de detecção social.

[Chae et al. \(2014\)](#) discutem a respeito das mídias sociais tornaram-se populares e as pessoas as utilizam para comunicação, não só em suas vidas diárias, mas também em situações desastrosas. Para [Martín, Li e Cutter \(2017\)](#) as plataformas de redes sociais, são canais de comunicação bidirecionais cada vez mais utilizado pelas autoridades, empresas e pesquisadores para compreender rapidamente a opinião pública e a atividade.

Para [Martín, Li e Cutter \(2017\)](#) situações extremas atraem a atenção pública com mais intensidade, o que gera ainda mais dados sobre uma determinada situação. Já [Avvenuti et al. \(2014\)](#) afirma que a quantidade de informações compartilhadas nas mídias sociais aumenta após as convergências de massa ou os eventos de emergência.

Segundo [Martín, Li e Cutter \(2017\)](#) no gerenciamento de desastres, exemplos de conteúdo de redes sociais espaciais são encontrados em terremotos, incêndios florestais, ciclones tropicais ou eventos de inundação. Além disso, estudos mostraram que as pessoas fisicamente próximas de um evento de desastre tendem a se envolver mais em conteúdo relacionado a desastres nas mídias sociais.

A análise desses dados, como a forma como as pessoas se preparam e respondem às catástrofes, desempenha um papel importante no gerenciamento de crises, na resposta a desastres e no planejamento de evacuação. Segundo [MacEachren et al. \(2011\)](#) as mídias sociais têm sido reconhecidas por muitos na pesquisa de gestão de crises e nas comunidades de prática como uma ferramenta para comunicação e coleta de informações.

35.2.3 Dados espaço temporais

Um dado reside em um elemento de informação ou representação de fatos ou instruções em forma apropriada para o armazenamento, processamento ou transmissão por meios automáticos. Vários dados correlacionados transformam-se em informação útil [FERREIRA \(1999\)](#).

Para [Martín, Li e Cutter \(2017\)](#) atualmente as mídias sociais estão disponíveis em tecnologias móveis, geralmente com receptores integrados do Sistema de Posicionamento Global (GPS). As atuais tecnologias, tais como, GPS e sensores proveem recursos que facilitam no acesso à determinados dados, como por exemplo, o *geotagged*. Segundo [Chae et al. \(2014\)](#) os *geotagged* oferecem uma precisão confiável em relação aos dados. A nível de software, há dois objetos JSON 'nível-raiz' usados para descrever o local associado a um Tweet⁷: (i) Coordenadas geográficas, latitude e longitude; e (ii) lugar referente ao nome do local.

Para [Martín, Li e Cutter \(2017\)](#) a disponibilidade de redes sociais baseadas em localização com dados de redes sociais *geotagged* melhorou significativamente a pesquisa e a prática de gestão de emergências, através do monitoramento de desastres e da reação das pessoas. Além disso, pesquisas usando dados de redes sociais *geotagged* e a identificação de padrões espaciais de usuários de redes sociais é generalizada, ou seja, podem ser encontrados conteúdos de mídia social acerca de terremotos, incêndios florestais, ciclones tropicais, eventos de inundação, dentre outros.

[Chae et al. \(2014\)](#) observam que um número crescente de pessoas estão usando serviços de redes sociais baseados em localização, como microblogs, onde são criados dados geo-localizados com marca de tempo. Os dados espaço-temporais possuem um grande potencial para melhorar a consciência situacional durante a situação de crise, fornecendo informações sobre o evento em evolução, a resposta pública e os possíveis cursos de ação.

35.2.4 Processamento de Linguagem Natural

Segundo [Kanakaraj e Guddeti \(2015\)](#) minerar opiniões e analisar sentimentos a partir de dados de redes sociais auxiliam em vários campos, como a previsão de eventos, a análise do clima geral do público em uma questão social específica, dentre outros aspectos. Para [Rajman e Besançon \(1998\)](#) as técnicas de Mineração de Texto são dedicadas à extração de informações a partir de dados textuais não estruturados, além disso, Processamento de Linguagem Natural (PNL) é tido como uma ferramenta interessante para

⁷ <https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object>

o aprimoramento de procedimentos de extração de informações. [Kanakaraj e Guddeti \(2015\)](#) afirmam que a análise do sentimento de texto envolve a aplicação de aptidão computacional na compreensão do sentimento implícito no texto, assim, o Processamento de Linguagem Natural envolve a tarefa de encontrar o significado semântico do conteúdo do texto e assim, analisar a informação do texto.

O Processamento de Linguagem Natural começou nos anos 50 como a subárea da inteligência artificial e da linguística, além disso, a PLN era distinta da recuperação de informações de texto que consiste em empregar técnicas altamente escalonáveis baseadas em estatísticas para indexar e pesquisar grandes volumes de texto. [Nadkarni, Ohno-Machado e Chapman \(2011\)](#).

Para [Maldonado et al. \(2016\)](#) a PLN analisa a estrutura da linguagem de acordo com cinco níveis: pragmático, morfológico, sintático, fonológico e semântico. Assim, o resultado da análise de opinião e da mineração de opinião depende em grande parte, do uso de ferramentas para executar diferentes tarefas da PNL [Zvarevashe e Olugbara \(2018\)](#).

35.2.5 Políticas Públicas de evacuação

Um bom planejamento de políticas de evacuação no cenário de catástrofes naturais desempenha um papel importante na prevenção de perdas de vidas humanas. Para [Kuhnen \(2009\)](#) as etapas fundamentais para se planejar num evento são: Prevenção, Reação, Recuperação e Mitigação.

Segundo [Kuhnen \(2009\)](#) a prevenção envolve desenvolver uma resposta de emergência e gerenciamento de capacidade antes de um desastre. Prevenir abrange sistemas de detecção e aviso, identificação de rotas de evacuação e abrigos, mantimentos de emergência e sistema de comunicação, procedimento para notificação e mobilização de pessoas especializadas e pré-estabelecimento de acordo com comunidades próximas.

Segundo [Kuhnen \(2009\)](#) a reação refere-se à ação deliberada imediatamente antes, durante e após um desastre. Já a recuperação envolve atividades tais como restaurar sistemas, além, do retorno à rotina normal e mitigação refere-se às políticas e ações que buscam reduzir a vulnerabilidade de danos de uma área para futuros desastres.

Tais atividades são fundamentais para o processo de evacuação, inclusive as medidas preventivas que podem ser desenvolvidas embasadas no âmbito da computação. Para [Kryvasheyev et al. \(2016\)](#), além das atividades, é importante haver uma análise de desastres naturais anteriores pois contribui para tomadas de decisão mais eficientes em situações futuras, auxiliando no combate a vulnerabilidades. Outro ponto que deve ser observado é o comportamento da multidão nessas situações para um melhor planejamento da rota.

Para [Kuhnen \(2009\)](#) metodologias que auxiliem a visualizar os níveis de interação entre a população e o meio são as mais indicadas. Portanto, explicar o desenvolvimento de uma situação de crise, bem como, integrar fatores para melhorar as capacidades de resposta da população em caso de futuras crises, são considerados métodos adequados no auxílio da evacuação. Segundo [Martín, Li e Cutter \(2017\)](#) a maior parte do uso das redes sociais em um contexto de desastre está focada nas fases de preparação e resposta do ciclo de gerenciamento de emergência.

35.3 Trabalhos Relacionados

Na Seção 3.1 é apresentado o protocolo de Mapeamento Sistemático. Na Seção 35.3.2 são abordados os trabalhos correlatos, onde os seguintes critérios foram avaliados para seleção desses trabalhos:

1. Define métodos para verificar se as pessoas evacuaram;
2. Define métodos para a extração de localização;
3. Utiliza um método/critério/abordagem para filtrar os dados relativos à ocorrência do desastre.

35.3.1 Revisão sistemática

A fim de realizar um apanhado no estado da arte foi realizado um mapeamento sistemática da literatura, a fim de extrair informações relevantes para o desenvolvimento deste trabalho. Desse modo, foi

buscado materiais com teor científico para o embasamento teórico através do protocolo de busca e etapas de planejamento.

Na etapa de construção do protocolo foram definidas questões de pesquisa, base de dados utilizada, string de busca e critérios de inclusão/exclusão.

Foram selecionados artigos publicados em anais de eventos, ou em revistas. Tais artigos foram buscados nas seguintes bases de dados: IEEE, PubMed, Science Direct, através da aplicação manual da seguinte String de busca: ("disaster management"OR "disaster environment") and ("social media"OR "social media data analysis"OR "spatiotemporal data analysis"). A fim de extrair informações que pudesse orientar o desenvolvimento do trabalho. Ao aplicar a string de busca nas bases de dados foram obtidos os resultados exibidos na Tabela 1:

Tabela 255 – Resultados nas Bases de Dados

Bases	Resultados
IEEE	1
PubMed	16
Science Direct	305

Ao todo foram retornados 322 artigos, contudo 51 deles, não estavam disponíveis, restando 271 artigos, conforme visto na tabela 2.

Tabela 256 – Detalhamento dos Resultados nas Bases de Dados

	Artigos encontrados	Artigos não disponíveis	Artigos disponíveis
Resultados	321	51	271

A quantidade de artigos disponíveis por base podem ser visualizados na Tabela 3.

Tabela 257 – Artigos Disponíveis por Base de Dados

Bases	Artigos encontrados	Artigos disponíveis
IEEE	1	1
PubMed	16	16
Science Direct	305	254

A seleção de artigos foi mediante a extração de informações consideradas relevantes para o desenvolvimento do atual trabalho. Foram desenvolvidas questões de pesquisa para responder se o artigo possui relação ao atual trabalho, e se contribui de alguma forma. A questão principal é: Como a literatura está abordando o gerenciamento de desastres a partir de dados das mídias sociais?

Além da questão principal foram formuladas três questões de pesquisa secundária, onde o artigo a ser selecionado deve responder às questões secundárias:

1. O trabalho aplica métodos de extração de informações de geolocalização?
2. O trabalho indica a fonte (entrevista, redes sociais, dataset) onde os dados de geolocalização são extraídos?
3. O trabalho define método/critério/algoritmo/abordagem para filtrar as informações relacionadas ao desastre?

Foram definidos critérios de Aceitação e Rejeição com o propósito de filtrar os artigos que respondem às questões formuladas.

Aceitação: Serão aceitos artigos na área de ciência da computação que tenha como objetivo a classificação e análise de dados relacionados ao cenário de desastres naturais.

1. Área da Ciências da Computação;

2. Realiza a análise de dados espaço-temporais;
3. Uso de filtros de palavras para classificação dos dados.

Rejeição: Serão rejeitados artigos duplicados, ou seja, caso haja o mesmo artigo em duas ou mais base de dados. Além de serem rejeitados resultados que não sejam artigos de jornal, anais ou conferências.

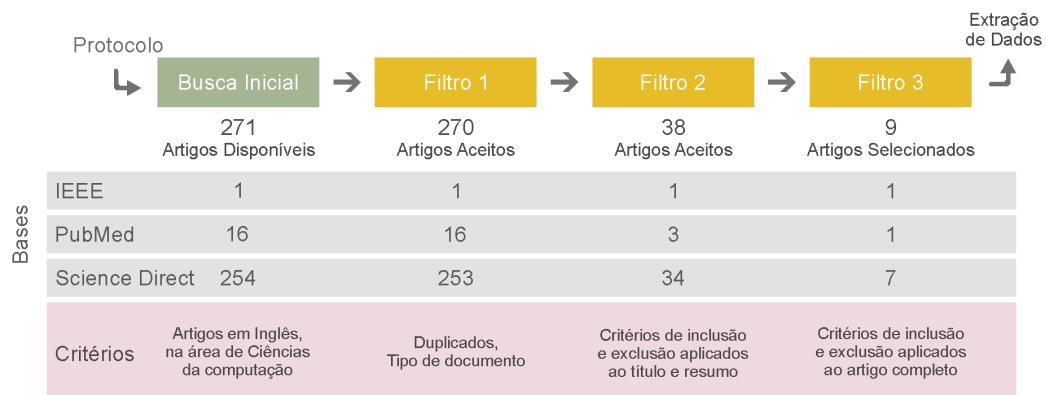
A leitura e organização dos artigos foi realizada com o apoio da ferramenta Mendeley⁸. Os artigos disponíveis foram baixados e importados para a ferramenta Mendeley, onde automaticamente são indicados os artigos duplicados, que faz parte do primeiro filtro. Nesta etapa foi identificado somente um artigo duplicado que estava disponível nas bases Science Direct e PubMed, totalizando em 270 artigos.

O segundo filtro aplicado residiu na leitura do título e resumo dos artigos. Esta etapa resultou em 38 artigos relacionados, correspondendo a 14,07% de artigos aceitos para o segundo filtro. Os artigos eliminados estavam fora dos critérios de pesquisa.

Por fim, foram lidos os 38 artigos restantes por completo, onde foram extraídas todas as informações necessárias para verificar a permanência do artigo como contribuinte para o atual trabalho. Desse modo, 9 artigos continham informações de interesse, sendo classificados como aceitos.

A Figura 2 mostra em detalhes todas a etapas da revisão sistemática:

Figura 390 – Etapas da Revisão Sistemática



Os artigos lidos abordaram o gerenciamento de desastres através da coleta de dados georreferenciados a partir das mídias sociais, a maioria dos trabalhos selecionados utilizou o Twitter para aquisição dos dados, em seguida, eles aplicaram filtros de palavras através da contagem da frequência de palavras ou análise sentimentos visando verificar a relação entre o desastre e os dados. Entre os trabalhos elencados, apenas 1 artigo define métodos que indica a evacuação da população, e nenhum define alguma orientação/guideline indicando o que fazer dado que ocorreu o desastre? (deslocar socorro, água, dinheiro, comida), alguns mencionam fazer, mas não é definida de que forma foi realizado, portanto, foi considerado que nenhum dos trabalhos mostrou guidelines para futuras evacuações.

Q1: O artigo menciona de onde são extraídos os dados? Dos 38 artigos lidos por completo, 73,68% citam a fonte dos dados, entre eles temos: Twitter, Facebook, entrevistas, ou dados extraídos através de uma revisão sistemática. Dos artigos que mencionaram a fonte dos dados, 89,28% extraíram dados do Twitter. Apenas 17,85% usaram dados do Facebook, 10,71% usou outros meios, é essencial expor que em alguns casos foi usado Twitter e Facebook no mesmo trabalho, mas o cálculo estatístico foi feito para cada mídia social utilizada.

Q2: O artigo define métodos para a extração de localização? Dos 38 artigos, 18 artigos definem métodos para a extração, ou seja, 47,36% responderam à questão. Dos 18 artigos que demonstraram métodos para a

⁸ <https://www.mendeley.com/>

extração da localização 61,1%, usou a metodologia de extrair somente dados com geolocalização ativada. Tal procedimento é possível por meio das bibliotecas de geocodificação disponível na API Twitter (mídia social mais usada), que possibilitam obter atributos de localização do usuário, incluindo país, estado, município, dentre outros tipos mais específicos de localização. Os 38,9% restantes, utilizaram outros meios, tais como Informações Geográficas Voluntárias (VGI) ou SIG(Sistema de Informação Geográfico).

Q3: Define método/critério/algoritmo/abordagem para filtrar as informações relacionadas ao desastre? Entre os 38 artigos lidos 39,4% definiram métodos de filtragem para selecionar apenas as fontes que tratavam do desastre. Desses artigos 86,66% utilizaram a técnica de contagem da frequência de palavras chaves a fim de construir uma *bag of words*. A análise de sentimentos foi usada em 33,33% dos trabalhos, é importante ressaltar que, a maioria dos trabalhos que utilizaram análise de sentimentos também usaram a contagem da frequência de palavras. A metodologia de quem usou ambos, era aplicar a princípio, contagem da frequência de palavras chave, ao ter conhecimento das palavras mais utilizadas, eram aplicadas técnicas de Processamento de Linguagem Natural (PLN) em cima dessas frases contendo as palavras mais frequentes. Esses trabalhos que utilizaram PLN, no geral, tinham como objetivo analisar o sentimento ou a opinião pública sobre o ambiente de desastres. O atual trabalho não fará o uso de técnicas de análise de sentimento, pois, o objetivo do atual trabalho não é análise do sentimento público.

35.3.2 Trabalhos selecionados

[Chae et al. \(2014\)](#) fornecem uma análise visual e temporal para comportamentos públicos durante o evento de desastre, sendo geradas trajetórias da distribuição dos usuários do Twitter usando a localização geográfica através de mensagens. Para tal tarefa foram coletados tweets com marcação geográfica e foram extraídos palavras chave em uma área específica usando a *Latitude Dirichlet Allocation*(LDA). Também foram identificadas as palavras mais comuns e incomuns dentro da área selecionada. As palavras de Tweets dentro da área selecionada, são ordenados por suas contagem de anormalidade. Esse trabalho definiu métodos para verificar se as pessoas evacuaram do local de desastre, tal tarefa foi feita através da verificação da distribuição dos usuários de acordo com cada data consecutiva, sendo plotado em mapas de calor. Foi utilizada também, a visualização baseada em glifos (figuras geométricas para representar os dados) para descrever os aspectos de localização e tempo do registro de dados independente, usando dois recursos visuais, sendo estes formas geométricas e cores.

[Kirilenko, Molodtsova e Stepchenkova \(2015\)](#) examinaram se as pessoas que vivem nos Estados Unidos (EUA) relacionam suas postagens com a temperatura local às mudanças climáticas. A análise foi realizada nos dois níveis de agregação nacional e local. A coleta de dados foi através do Twitter, onde, foi desenvolvido um software que busca as palavras chave. Para verificar a relação dos tweets com o ocorrido, foi feita uma contagem da frequência de palavras chave, além de aplicar uma plataforma Natural Language Toolkit⁹, onde foi realizada uma análise de opinião automatizada em um pequeno subconjunto dos tweets coletados.

[Kryvasheyeu et al. \(2016\)](#) apresentaram uma análise multiescala da atividade do Twitter antes, durante e depois do furacão Sandy, sendo, examinada a resposta on-line de 50 áreas metropolitanas dos Estados Unidos e foi encontrada uma forte relação entre a proximidade com o caminho de Sandy e a atividade de mídia social relacionada a furacões. Desse modo foram coletados tweets geomarcados. Após agregar os tweets por localização, foram utilizados carimbos de hora para análise temporal. Também foram analisadas as frequências das palavras que foram mais prevalentes entre mensagens positivas e negativas. Nesse trabalho foi mostrado que as palavras-chave relacionadas ao furacão foram aumentando lentamente com um pico forte no dia do desembarque do furacão, e em seguida houve um gradual declínio no nível de atividade do tweet. O trabalho usou mapas de calor para demonstrar as cidades pela sua proximidade com o furacão, e mostrou que palavras relacionadas aumentaram a atividade aumenta com a proximidade do furacão.

[Anbalagan e Valliyammai \(2016b\)](#) fizeram o mapeamento de crises durante desastres usando mídias sociais usando como estudo de caso Jakarta na Indonésia. Sendo utilizado Informações Geográficas Voluntárias (VGI) para capturar os locais de abrigos de evacuação por inundação com base nas preferências

⁹ www.nltk.org

dos residentes durante os períodos de cheias. Foi acessada a API do Twitter, e extraídos tweets com hashtags geomarcadas, desse modo, foram gerados mapas geo-codificados para identificar a precisão do nível da localização. Os tweets coletados foram analisados em um corpus para análise eficaz do texto usando a PNL. As etapas a seguir são executadas para limpar o corpus e prepará-lo para mais análises.

[Kusumo, Reckien e Verplanke \(2017\)](#) pesquisaram a utilidade das Informações Geográficas Voluntárias (VGI) para o planejamento de abrigos de evacuação de inundação. Usando Jakarta, na Indonésia, como um estudo de caso, foi usado o VGI para capturar os locais de abrigos de evacuação por inundação com base nas preferências dos residentes durante os períodos de cheias e compará-los com os locais dos abrigos oficiais. Inicialmente foi verificada a validade ou relevância dos dados geolocalizados do Twitter, posteriormente foi aplicada uma análise de classificação de conteúdo de tweets relevantes. No referido trabalho haviam 306 tweets referentes a abrigos de evacuação, assim, dos 306 tweets, 215 estavam localizados em abrigos. Os resultados foram mostrados a partir de mapas, onde era possível observar os pontos de localização de cada tweet e os pontos de abrigo. Contudo, em alguns casos, o evacuado e o voluntário não foram diferenciados.

[Murthy e Gross \(2017\)](#) buscaram estender a pesquisa em ciências sociais sobre o uso de tecnologia de mídia social durante eventos de desastre, além de explorar mudanças no comportamento das pessoas através da mídia social Twitter, durante eventos de desastres. Assim, foram coletados tweets identificados geograficamente antes e depois do furacão Sandy. Para apurar o relacionamento entre os tweets e o desastre, foi verificada a frequência de co-ocorrência em tweets com duas ou mais hashtags. Cinco dos dez principais pares de hashtags incluem uma hashtag relacionada a tempestades com uma *hashtag* relacionada à localização. Esse trabalho ilustra quantitativamente que as pessoas dos estados próximos ao caminho do furacão, publicavam tweets em frequências mais altas.

[Becken et al. \(2017\)](#) monitoraram o ambiente e sentimento humano em relação à grande barreira de corais na Austrália, foi usada a API do Twitter para recuperar dados. Durante a coleta havia restrições para capturar tweets com marcação geográfica postados da região de interesse. As palavras-chave foram extraídas usando uma técnica de busca insensível a maiúsculas e minúsculas, e variações da mesma palavra (por exemplo, "mergulho", "mergulhando") foram compiladas como a mesma palavra-chave. Os números de ocorrências para cada palavra-chave foram contados. Além das frequências de palavras-chave, os tweets foram analisados com relação à polaridade positiva ou negativa. Por fim, foi mostrado um mapa de calor de tweets georreferenciados.

[Chong et al. \(2018\)](#) apresentaram um estudo conceitual sobre o uso potencial das mídias sociais, sendo usado o Twitter, neste caso, para entender melhor a resiliência da infraestrutura em Chennai, na Índia, usando os dados extraídos da inundação de 2015. É aplicada a *geotagging* e a mineração de texto para explorar fontes ricas de dados de mídia social, como vídeo, website, texto e código de resposta rápida (QR code). Desse modo, foram coletados tweets com marcação geográfica do local de específico. Para a obtenção de informações de interesse, foram aplicadas técnicas de mineração de texto. Assim, foram contadas a frequência da ocorrência de palavras chave. Esse estudo mostra que essas técnicas são úteis na identificação de locais, defeitos e intensidades de falha de infraestrutura, através do uso de metadados de localização de geotags, palavras que contêm os locais e as frequências de tweets de cada local.

[Wu e Cui \(2018\)](#) realizaram uma análise com base em múltiplos recursos de dados, combinando dados de mídia social, perdas econômicas e informações geográficas. Foram verificadas as funções desempenhadas pelas mídias sociais antes, durante e depois de um desastre natural. Para obter e retificar informações geográficas mais precisas, foram usadas ferramentas de programação por meio das bibliotecas de geocodificação Google Map API e OpenStreetMap para reverter o geocódigo do Twitter. Além disso, foi aplicada a análise de sentimento, a fim de medir a emoção ou o humor expresso em cada tweet. Devido à insuficiência dos métodos de análise de sentimentos, alguns tweets foram atribuídos à classe errada. Para superar tal problema, foi necessário utilizar a outros métodos de análise de palavra, tais como, contagem da frequência de palavras-chave. Esse trabalho mostra gráficos que indicam que desastres, situações especiais e problemas emocionais têm correlação significativa com a atividade do Twitter, enquanto o índice de sentimento tem uma correlação negativa.

Tabela 258 – Comparação de Trabalhos Correlatos

Artigo	Aplica métodos de extração de geolocalização	Define método/critério/algoritmo para filtrar as informações relacionadas ao desastre	Define métodos para verificar se as pessoas evacuaram?	Propõe guidelines
(CHAE et al., 2014)	x	x	x	
(KIRILENKO; MOLODTSOVA; STEPCHENKOVA, 2015)	x	x		
(KRYVASHYEYU et al., 2016)	x	x		
(ANBALAGAN; VALLIYAMMAI, 2016b)	x	x		
(KUSUMO; RECKIEN; VERPLANKE, 2017)	x	x		
(MURTHY; GROSS, 2017)	x	x		
(BECKEN et al., 2017)	x	x		
(CHONG et al., 2018)	x	x		
(WU; CUI, 2018)	x	x		
Atual Trabalho	x	x	x	x

35.4 Proposta

Este trabalho consiste em limpar e filtrar espaço-temporais coletado a partir do Twitter a fim de identificar rotas de evacuação em cenários de desastre. Primeiro, apresentaremos o cenário de desastre tratado no trabalho. Em seguida discutiremos o processamento dos dados coletados para a construção da base de dados.

35.4.1 Cenário de desastre

O terremoto ocorreu no Havaí como resultado de fissuras na lateral do sul do vulcão Kilauea, na zona leste do rifte, no qual consiste em grandes fraturas tectônicas. Para fr (2018) a abertura do solo, a uma profundidade de 50 metros, foi contínua e esteve associada à atividade sísmica. As áreas de erupção foram na Zona Leste do Rift e no topo do Kilauea. Segundo a USGS, os moradores e visitantes perto das fissuras dos fluxos de lava e da área de colapso da cúpula devem se manter atentos aos avisos do Departamento de Defesa Civil e do Parque Nacional do Havaí¹⁰.

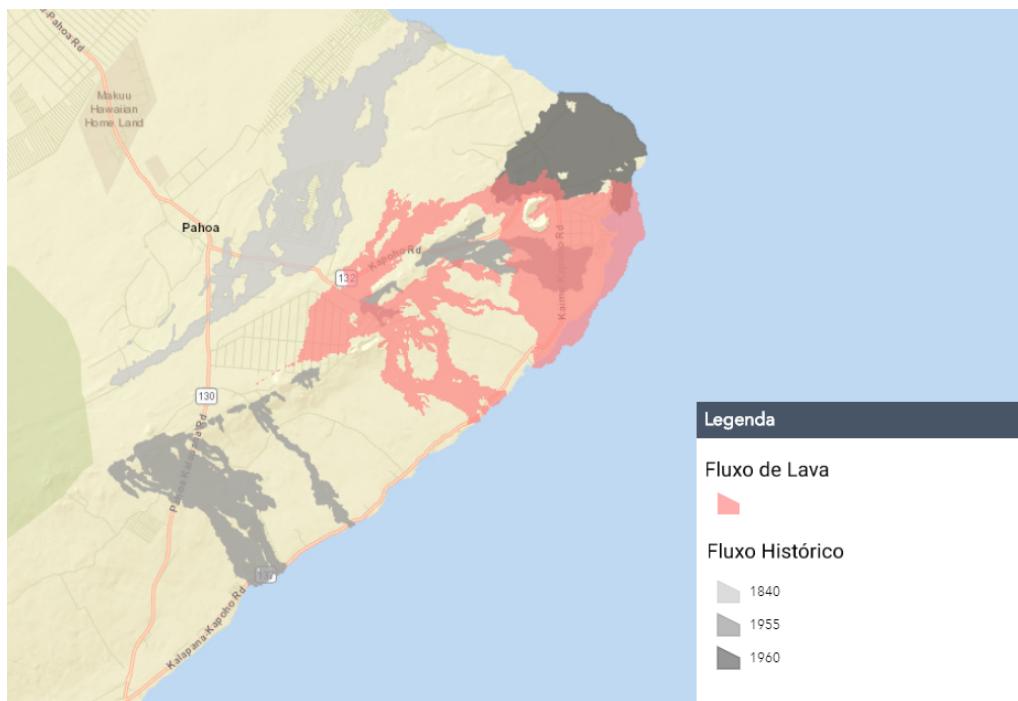
As Figuras 3, 4 e 5, foram obtidas a partir do site do County of Hawaii¹¹ no qual possui dados geológicos disponibilizados pela USGS. Tais figuras servem como apoio para relacionar o fluxo de lava com o trajeto de evacuação sugerido na conclusão e justificar por qual motivo as pessoas devem se deslocar para determinados locais durante o processo de fuga. Tais figuras mostram as áreas mais afetadas, onde escorreu a lava e a zona leste do rifte, as áreas seguras próximas às áreas afetadas, bem como, os locais onde ocorreram abalos sísmicos.

Conforme observado na Figura 3, a mancha vermelha representa o fluxo de lava que cobriu o Leste do Havaí, especialmente nas cidades de Kapoho e Leilani States. Ademais, tais áreas afetadas, possuem o histórico de serem atingidas em outras ocasiões de desastre, conforme visto nas manchas em cinza representando dados de fluxos históricos.

¹⁰ <https://volcanoes.usgs.gov/volcanoes/kilauea/status.html>

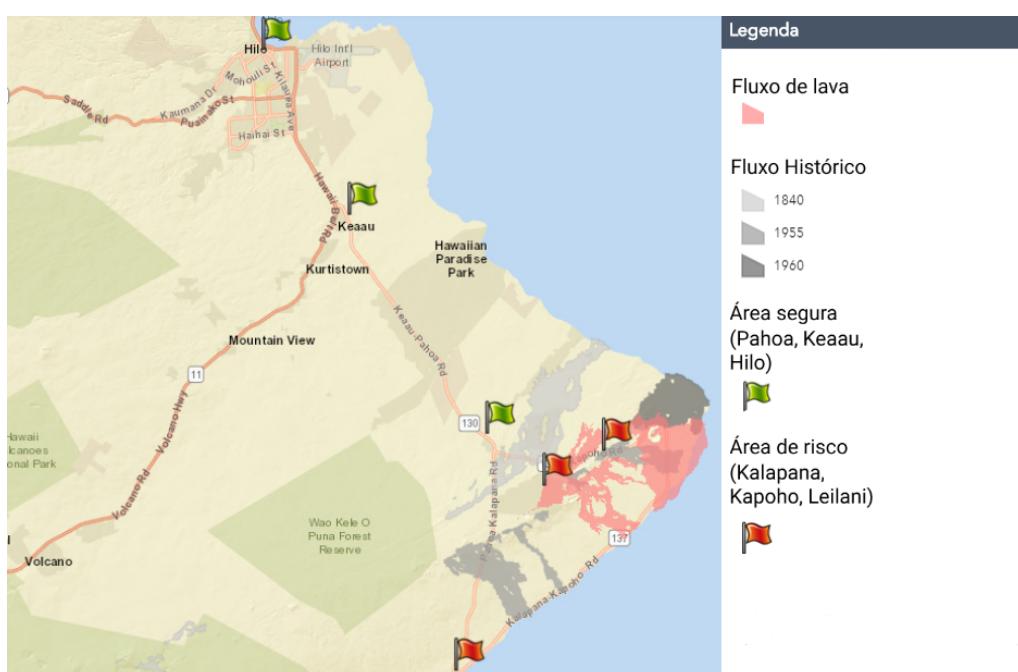
¹¹ <https://hawaiicountygis.maps.arcgis.com>

Figura 391 – Mapa do fluxo de lava. Em vermelho corresponde ao ano de 2018 e em cinza fluxos históricos



A Figura 4 mostra as áreas de risco e áreas seguras onde estão disponíveis os abrigos mais próximos às áreas de risco. As áreas de maior risco de receber fluxo e gases vulcanicos são: Kalapana, Kapoho e Leilani States, no qual estão representadas com bandeira vermelha nesta figura. Tais regiões possuem alto risco de serem afetadas pela lava e gases vulcanicos, pois ficam situadas no rifte e já possuem histórico de fluir a lava para tais regiões.

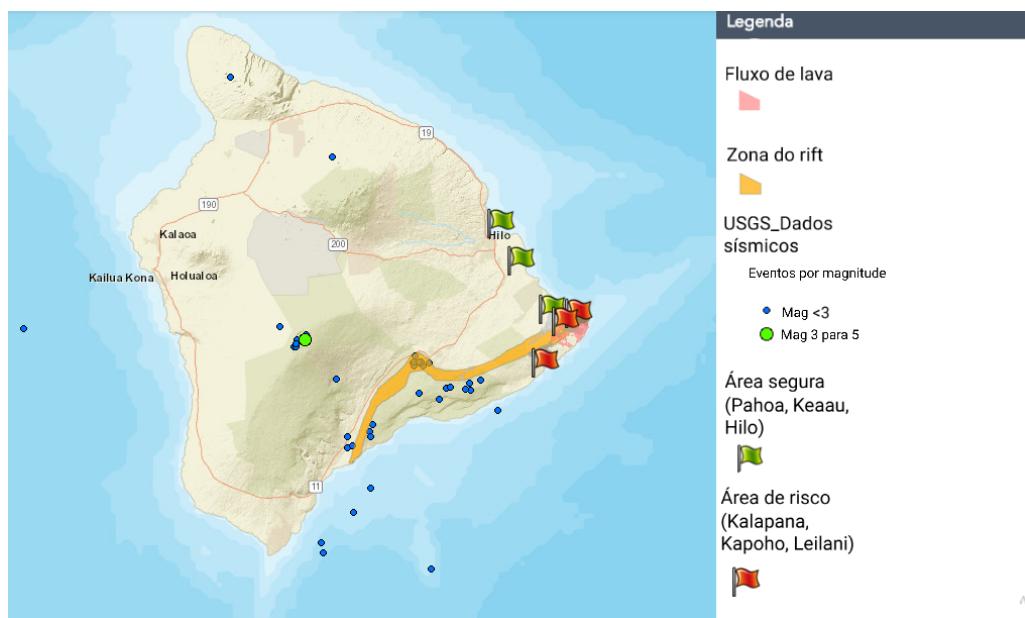
Figura 392 – Áreas de risco e áreas seguras mais próximas às áreas de risco



A Figura 5 mostra o rifte existente no leste do Havaí. Juntamente com dados de abalos sísmicos na ilha do Havaí. Toda a área amarela, onde possui um rifte é área de risco, pois, a mesma possui placas

tectônicas fragmentadas, no qual, a lava escorre por tal região abrindo fissuras no solo e colocando a lava para fora. Assim, a região do rife possui altas chances de deixar as vias e ruas comprometidas. Além disso, são mostrados dados acerca de abalos sísmicos.

Figura 393 – Área do rife e dos dados de terremotos



35.4.2 Mineração dos dados

A Figura 6 apresenta a metodologia seguida para alcançar os resultados obtidos.

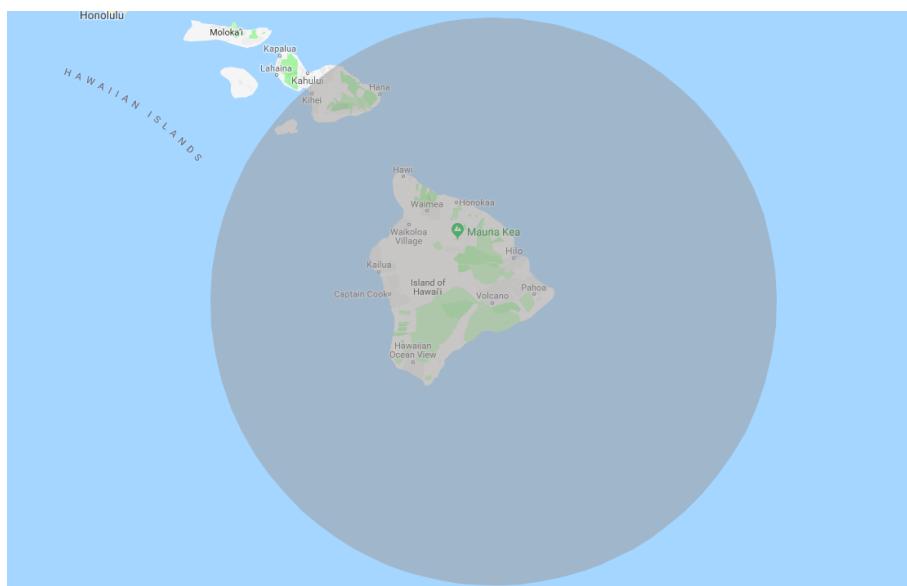
Figura 394 – Metodologia de processamento dos dados



O critério de seleção dos dados é a localização geográfica dos tweets. Portanto, foram coletados somente tweets com a localização habilitada e provenientes da região da ilha do Havaí, com latitude: 19.420120, longitude: -155.253137 e raio de 200 Km. Foram selecionados tweets postados durante o terremoto e erupção do vulcão Kilauea, que ocorreu por volta de 3 de maio à 11 de junho de 2018. A Figura 7 mostra a área correspondente à coleta de dados. Na etapa de *aquisição* dos dados, a coleta ocorreu, diariamente durante 12 horas por dia, ao longo da ocorrência do desastre natural na maior ilha do Havaí no período que inicia em dia 27 de abril de 2018 à 23 de maio do mesmo ano. Foram considerados apenas os tweets dos usuários cuja a informação de localização está habilitada. Ao todo foram coletados 6023 tweets.

Os dados obtidos, estão devidamente organizados e, todos os tweets resultantes possuem relação direta com o desastre em questão. Tendo em vista que o Havaí é uma pequena ilha pertencente aos Estados Unidos, a ilha possui uma população pequena, contudo há uma quantidade considerável de usuários de mídias sociais. O conjunto de dados possui o seguintes campos de interesse: ID (número de identificação do tweet), Data, Latitude, Longitude e Horário. As localizações foram disponibilizadas por latitude e longitude em um formato específico definido pelo Twitter. Existem dois objetos JSON usados para descrever o local associado

Figura 395 – Área de abrangência dos dados



a um *tweet*: coordenadas e local¹². A Figura 8 apresenta um exemplo de *tweet* em formato *JavaScript Object Notation*(JSON).

Figura 396 – Exemplo de como são disponibilizados os tweets

```
{
  "created_at": "Thu Apr 06 15:24:15 +0000 2017",
  "id": 850006245121695744,
  "id_str": "850006245121695744",
  "text": "1/ Today we're sharing our vision for the future of the Twitter API platform!nhttps://t.co/XweGngmx1P",
  "user": {},
  "entities": {}
}
```

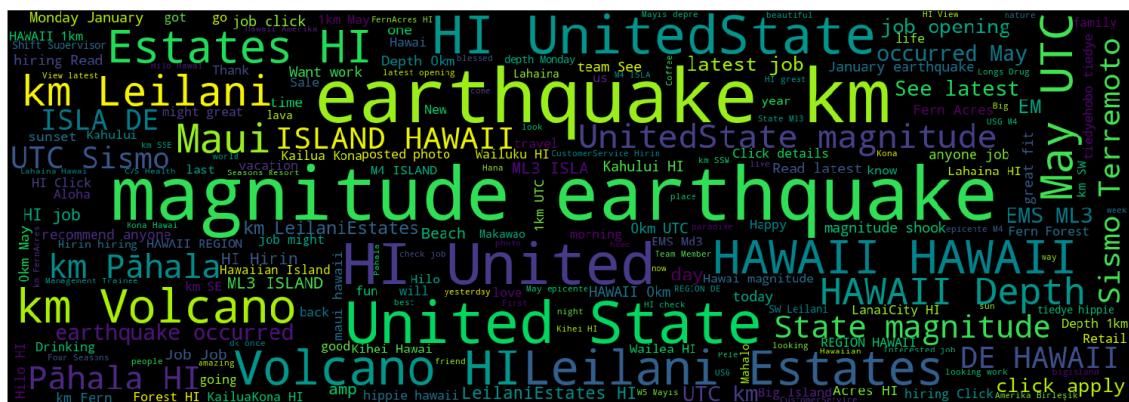
Na fase de limpeza dos dados, foram aplicadas técnicas de Processamento de Linguagem Natural (NLP) para a normalização do texto do tweet, foram usadas técnicas de tokenização, *stemming* e remoção de ruídos. Diante do exposto, foi aplicada a tokenização no qual consiste em decompor a mensagem inteira em termos/palavras isoladas.

Em seguida nós utilizamos a *stemming* mantendo somente o sentido principal da palavra. A técnica de *stemming* foi feita utilizando a biblioteca do Python NLTK (*Natural Language Tool Kit*), com a finalidade de criar algoritmos que funcionam com linguagem natural.

E, por fim, removemos o que consideramos ruídos, tais como: números, caracteres especiais, e *stopwords* que consistem em palavras muito comuns que não são significativas para nosso dataset. Em relação a caracteres especiais houve exceção para *emojis*, que foram agrupados para análise de sentimento. Posteriormente foi executado um filtro para remoção de conteúdo duplicado, possíveis spams. Durante o processo de remoção de conteúdo duplicado foram identificados perfis com conteúdo similar para todas as suas postagens, perfis de divulgação e impulsionamento de conteúdo e marcas, perfis de postagens dinâmicas sobre hora, tempo e empregos, postagens que não são interessantes para este trabalho. Conforme visto no Seção 3, a técnica mais utilizada nos trabalhos relacionados para verificar a relevância de algum dado é a contagem da frequência de palavras. Diante do exposto, verificamos quais as palavras mais frequentes nos tweets obtidos e foi gerada a nuvem de palavras da Figura 9.

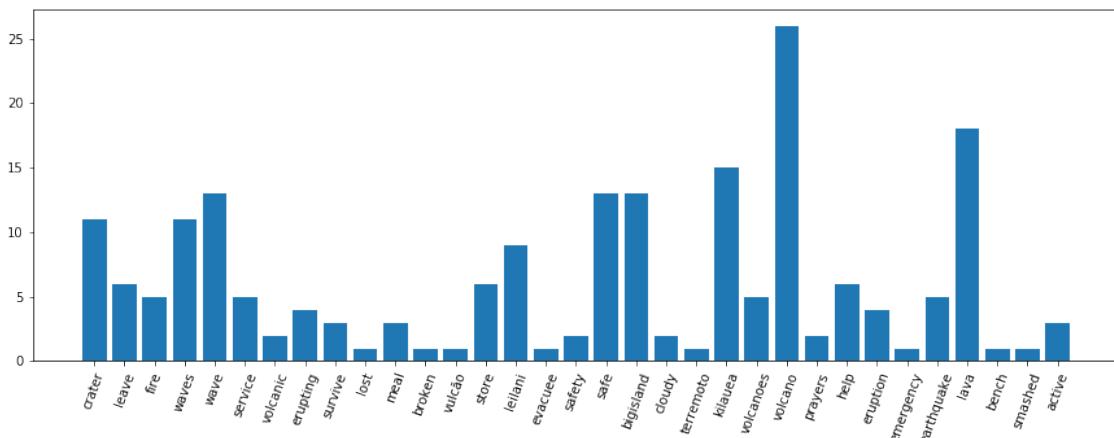
¹² <https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/geo-objects.html>

Figura 397 – Palavras mais usadas nos tweets durante o evento de desastre no Havaí



Contudo, foi observado que algumas palavras não faziam sentido para nosso trabalho, portanto foi desenvolvido o dicionário de palavras relacionadas afim de filtrar somente os Tweets que se relacionam ao desastre. O dicionário foi construído inspirado no trabalho de [Murthy e Gross \(2017\)](#) e adaptado ao contexto do desastre do Havaí¹³. O dicionário foi organizado classificando as palavras por categorias, tais como: Earthquake, Volcane, Locations, Weather, Emotion, Relief and donation, Social effects, Social movements, Disaster related damage. Para verificar a relação entre os dados e o desastre foi realizada a contagem da frequência de palavras mais utilizadas nas postagens. À vista disso, as palavras mais utilizadas no Havaí durante o desastre, podem ser visualizadas na Figura 10.

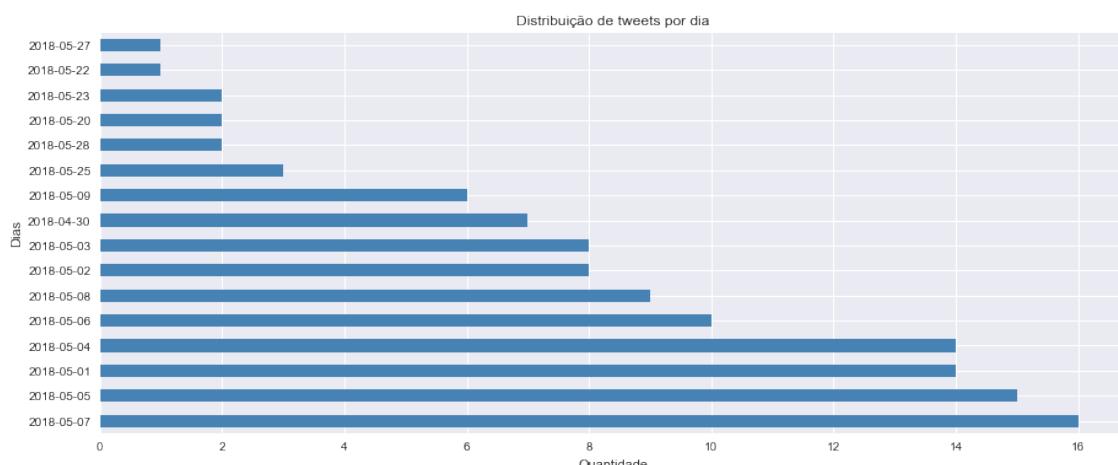
Figura 398 – Palavras relacionadas ao desastre mais usadas nos tweets durante o evento de desastre no Havaí



Foi realizada uma contagem da quantidade de tweets que foram postados por dia. Conforme visto na Figura 11, os dias com maior quantidade de postagens foram no início do mês de maio de 2018.

¹³ Dicionário de palavras-chave: <https://docs.google.com/spreadsheets/d/1nWsmAn2jvDyOxmljRlyLrfca8dlydN-J4iuHG-BBBQO/edit?usp=sharing>

Figura 399 – Quantidade de tweets por dia



Através dos resultados do deslocamento da multidão e um estudo sobre a região, foi possível desenvolver diretrizes. Tais diretrizes são referentes às orientações que irão ser úteis em futuros casos de desastres. As diretrizes foram desenvolvidas com base nos dados obtidos os quais estão relacionados à escolha da população em evacuar para determinados locais e também nos fundamentamos em estudos mais específicos sobre a geografia do Havaí.

Ao todo foram coletados 6023 após os todos os processos de limpeza e filtragem foi obtido um total de 118 tweets.

35.5 Diretrizes

Esta seção visa apresentar diretrizes para orientar o processo de evacuação em casos de desastres naturais semelhantes ao ocorrido no Havaí, tratado no estudo de caso desse trabalho.

(i) Conhecimento sobre o local e sua geografia: a primeira diretriz diz respeito ao conhecimento sobre o local e sua geografia, ou seja, as autoridades e pesquisadores devem fazer estudos acerca das possíveis ameaças mediante estudos geofísicos e meteorológicos. A partir de estudos geofísicos é possível identificar ameaças como terremotos que consequentemente podem ocasionar vulcões, assim como ocorreu no Havaí, inicialmente houve abalos sísmicos que consequentemente ocasionou a erupção do vulcão Kilauea. Os estudos meteorológicos facilitam na previsão de tornados, furacões, chuvas intensas, bem como, outros fenômenos naturais que possam ser previstos através do estudo da atmosfera terrestre. Em nosso estudo de caso, terremotos e vulcões são recorrentes, as comunidades locais já possuem conhecimentos acerca de tais fenômenos e de como se preparar e se recuperar de tais catástrofes.

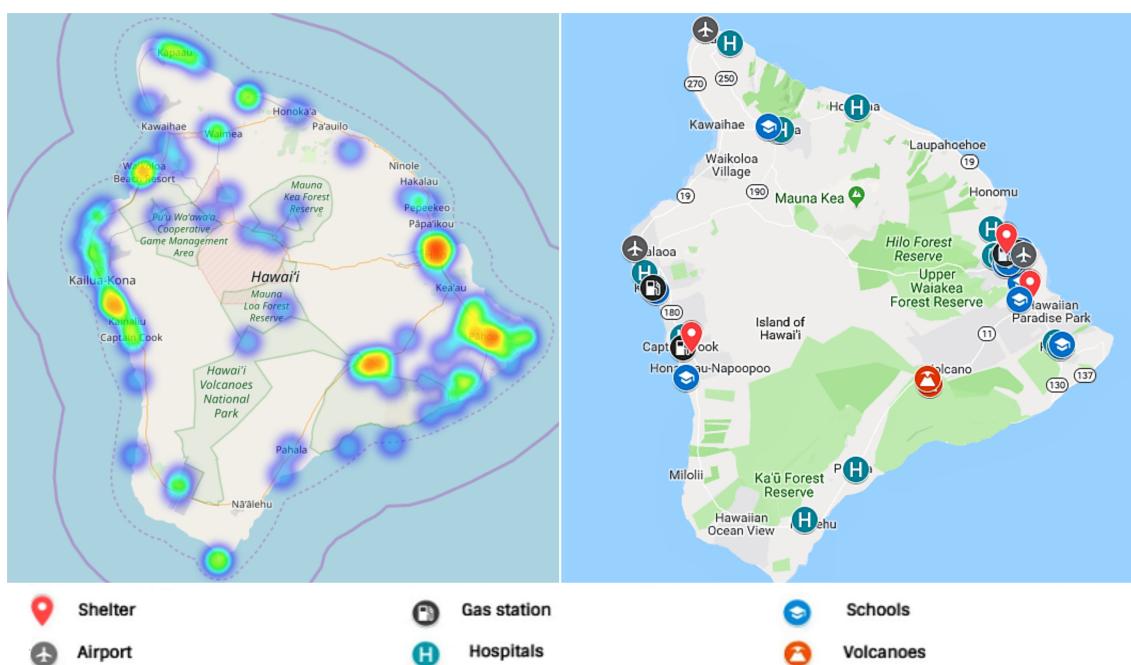
(ii) Densidade populacional: tendo em vista que são identificados os locais mais vulneráveis, é necessário obter informações acerca da densidade populacional em cada área de risco. Desse modo, o tamanho da população deve ser refletida no tamanho da frota de veículos que irão trafegar. Além disso, há grupos de pessoas especiais com mobilidade limitada, tais grupos populacionais são dependentes do trânsito e isto deve ter atenção especial. O objetivo final desta etapa é definir um número de veículos que saem das áreas de risco em direção às áreas seguras. Segundo Stepanov e Smith (2009) o principal provedor de dados (o programa do Censo dos EUA) possui informações exatas sobre a distribuição da população, ou seja, são informações sobre onde as pessoas vivem, assim, é possível estar atualizado acerca da densidade populacional das regiões pertencentes aos Estados Unidos.

(iii) Recomendação de rotas: uma vez que são identificados os locais vulneráveis e tamanho da população de cada área específica, é necessário que as autoridades desempenhem o papel de recomendar os residentes da área de risco a evacuarem para um local mais seguro, onde os dados gerados nesse trabalho exercem papel fundamental. Desse modo a terceira diretriz depende das primeiras. Por exemplo, o governo Havaiano recomendou que os residentes próximos da cidade de Kapoho e Kalapana fugissem para o centro comunitário

de Pahoa ou para o centro comunitário de Keaau. Tendo em vista que Kalapana fica muito próxima à um vulcão, é sabido que os moradores de Kalapana poderiam ter sérios problemas respiratórios por causa da emissão de gases do vulcão, e em casos mais extremos, moradores dessa localidade poderiam ser atingidos pela lava e serem vitimizados por um possível incêndio. Além disso, ambas localidades ficam próximas ao rife, portanto, a lava escorre para tais regiões ocasionando fissuras no solo, comprometendo vias de trânsito e moradias. Segundo a defesa civil havaiana, nesses abrigos estavam disponíveis água, comida e cuidados para animais¹⁴. Desse modo, o planejamento de rotas, bem como, recomendações para locais de abrigo mais próximos se faz necessário para as comunidades que vivem em locais vulneráveis a desastres naturais. É importante notar que a cidade de Kalapana fica à direita da reserva natural do Kilauea, onde estão localizados os principais vulcões. Conforme mencionado no trabalho de Fujihara e Miwa (2012) os evacuados normalmente conhecem alguma área de refúgio recomendada oficialmente pelo governo local, contudo, se a área de refúgio conhecida não estiver disponível por algum motivo, então os evacuados não saberão para onde ir. Neste caso, é necessário sugestões de abrigo, bem como rotas alternativas. No caso de um vulcão no Havaí, o treinamento sugerido seria que os residentes próximos ao rife e a locais com históricos de fluxo de lava se desloquem para os locais seguros mais próximos de forma distribuída afim de evitar congestionamentos. De posse dos dados coletados e dos estudos realizados, foram ilustradas algumas possíveis rotas que podem ser seguidas no caso específico de terremoto e ou vulcão no Havaí.

É notável que há relação direta entre os locais de abrigo e os locais com maior densidade populacional. Conforme visto na Figura 12, o mapa de calor à esquerda representa os locais com maior densidade de pessoas que publicaram no twitter na época correspondente ao desastre. O mapa da direita representa os locais de abrigo no Havaí, conforme exibido na legenda da figura, os abrigos representados são: escolas/universidades, hospitais, postos de gasolina, aeroportos, abrigos, bem como o local da cratera vulcânica.

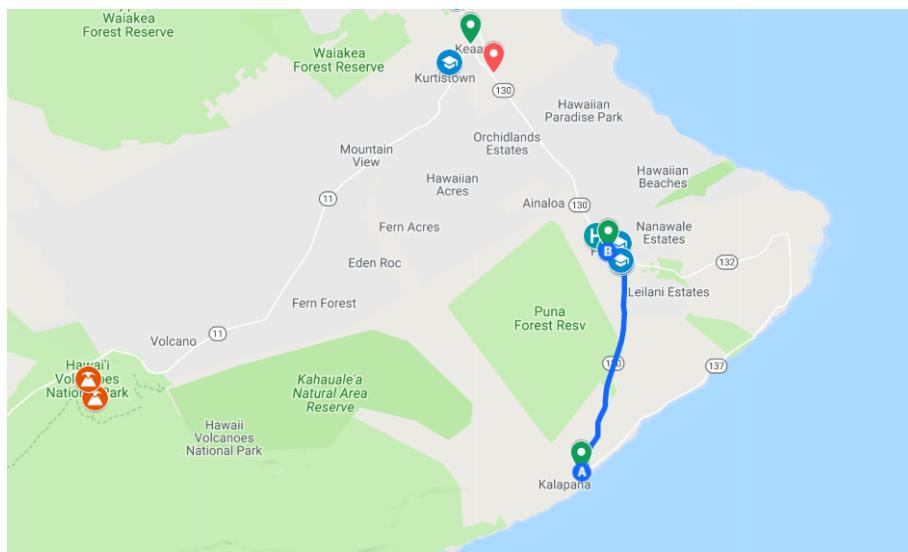
Figura 400 – Locais com maior quantidade de tweets/ Locais com abrigos



As rotas ilustradas nas Figuras 13, 14 e 15 foram criadas com base nos mapas de calor gerados com a localização das pessoas, bem como, na análise das vias de trânsito, principais locais de abrigo e áreas de risco do Havaí. A Figura 13 ilustra uma rota para de escapar da área de risco e dos gases do vulcão. A rota de Kalapana à Pahoa possui aproximadamente 20,3 km de distância. Por ser uma das áreas de risco mais distante dos abrigos seria conveniente que os residentes de Kalapana permanecessem em Pahoa do qual é a área mais próxima fora de risco e com abrigos disponíveis.

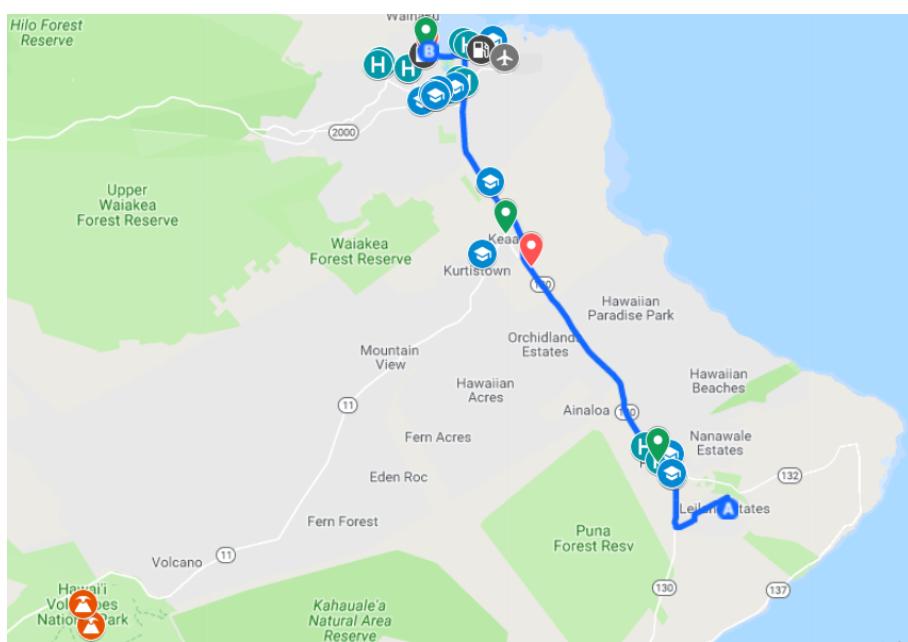
¹⁴ <http://dod.hawaii.gov/hiema/update-from-hawaii-county-civil-defense/>

Figura 401 – A rota de Kalapana a Pahoa, no qual possui 20,3 km de distância.



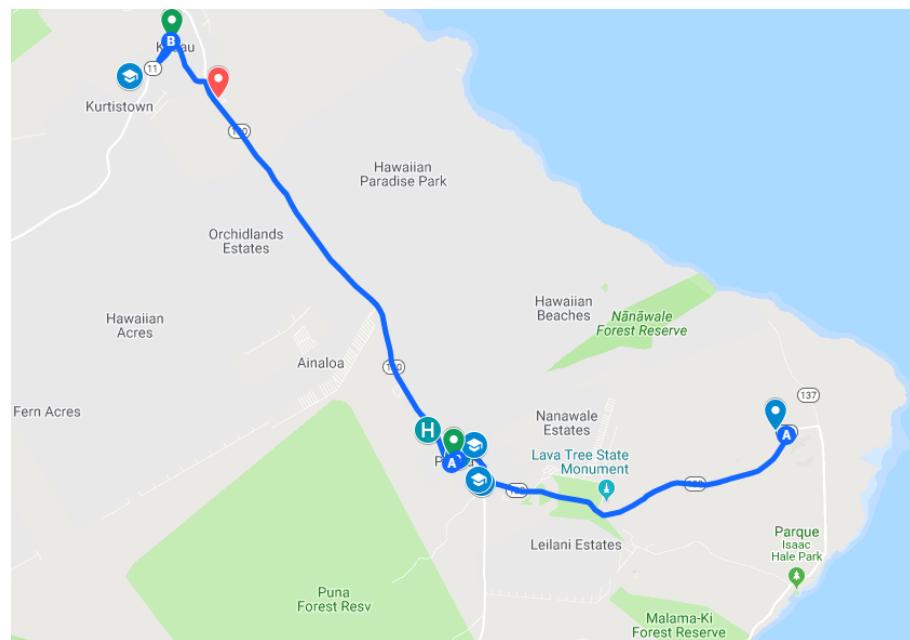
Outra localidade que é bastante vulnerável em casos de vulcões é Leilani States, que fica no leste do Havaí, tal área juntamente com Kapoho foram as mais atingidas pela lava. É necessário haver campanhas de evacuação nesta área, pois, Leilani fica justamente na região do rife onde lava escoa ou abre crateras, conforme visto na Figura 5. Desse modo, na Figura 14 é sugerida um rota de Leilani para Hilo, onde o caminho possui aproximadamente 27 km. Sugerimos apenas uma rota, pois, esta região é a mais afetada por lava vulcânica, e todas as outras rotas analisadas são áreas de extremo risco e ficam totalmente comprometidas pela lava. Embora tal localidade fique bem próxima a Pahoa, a população de Leilani seria mais bem distribuída para Keau. A distância de Leilani para Hilo é apenas 27 km, desse modo, Pahoa ficará disponível para os locais mais distantes, tais como Kalapana. Uma outra sugestão é que os residentes de Leilani evacuem imediatamente a Pahoa que possui apenas 8,9 km de distância, e que após receberem recursos tais como água, alimentos e atendimento médico, os evacuados prossigam até Hilo, deixando espaço para os evacuados dos locais mais distantes que provavelmente chegarão após os residentes de Leilani desocuparem os abrigos de Pahoa.

Figura 402 – Sugestão de rota de Leilane a Hilo



Outra região bastante afetada pela lava é Kapoho. Esta deve ser a região em que os residentes devem ser evacuados com maior urgência possível. Pois todas as ruas disponíveis ficam tomadas pela lava e esta fica localizada na área do rife. Os residentes desta área devem procurar evacuar assim que obter as previsões meteorológicas e geológicas, conforme menciona a primeira diretriz, além de orientações da defesa civil. Desse modo, evitam pegar a estrada tomada por lava e congestionada. Conforme visto na Figura 4 a área de Kapoho ficou totalmente tomada pela lava. Portanto os residentes de Kapoho devem seguir para Pahoa, no qual possui aproximadamente 13 km de distância. Caso seja necessário prosseguir a Keau, para uma melhor distribuição da população, a distância é de aproximadamente 30,3 km de distância.

Figura 403 – Sugestão de rota de Kapoho a Pahoa e Kapoho a Keau



Lamentavelmente não há rotas alternativas para as três áreas descritas, pois todas elas ficam com estradas comprometidas pela lava vulcânica. A melhor forma de distribuição encontrada foi direcionar os residentes de cada local a um abrigo distinto.

(iv) Mitos e informações equivocadas: a quarta diretriz é relacionada a mitos ou informações equivocadas que dificultam a eficácia do planejamento de emergência. Embora a resposta rápida seja crucial no momento de emergência, é necessário que as informações obtidas sejam verídicas, pois, segundo [Perry e Lindell \(2003\)](#), informações equivocadas causam pânico podendo levar as autoridades a tomar atitudes fracassadas em relação à proteger as pessoas. Além disso, esses mitos de desastre dificultam a eficácia do planejamento de emergência, desorienta a alocação de recursos e a disseminação de informações [Perry e Lindell \(2003\)](#). Além disso, há perfis que aproveitam da situação para divulgar produtos e promoções através de hashtags referentes ao desastre. Neste trabalho foram identificados 179 bots, dos quais apresentavam postagens com padrões repetitivos em curto intervalo de tempo.

(v) Processo de planejamento de emergência: a quinta diretriz para o processo de planejamento em emergências, é que devem ser fornecidos testes para operações de respostas a desastres de forma contínua. Um planejamento eficaz reside no processo contínuo, onde as mudanças devem ser incorporadas em todos os aspectos do sistema de gerenciamento de emergências. Tais mudanças dizem respeito às tecnologias aplicadas, equipamentos de teste, comunicação, bem como, treinos para a comunidade local. Devem ser realizados treinamentos para a comunidade de acordo com estudos e riscos identificados, conforme visto na primeira diretriz. No trabalho de [Fujihara e Miwa \(2012\)](#) foi observado que na maioria das vezes a rota de evacuação foi determinada de acordo com os conhecimentos dos evacuados.

Um planejamento de desastre abrange todos os aspectos de recuperação: organização de gestão, material, equipamentos, logística, mão de obra e comunicações [Lindsey \(2008\)](#). Programas de treinamento e

avaliações pós-evento que apoiam a melhoria contínua são essenciais para um programa bem-sucedido.

Conforme os dados minerados, seria necessário deslocar água, alimentos e roupas, aos locais mais próximos ao vulcão. Os locais próximos ao vulcão devem ser priorizados a partir de um abalo sísmico, pois esses abalos podem ocasionar a erupção do vulcão, portanto, os residentes deste local precisam de ajuda redobrada para enfrentar os tipos de desastre naturais.

Os locais próximos ao vulcão também devem ter apoio médico, pois, os residentes podem obter graves doenças respiratórias, irritação nos olhos e na pele, o que pode dificultar a evacuação dessas pessoas para os abrigos. As comunidades também devem estar atentos aos recursos que devem ser estocados. Um exemplo é que ao ser previsto um abalo sísmico, deve-se estocar água e alimentos, e ficar atento às recomendações da defesa civil através do meio de comunicação disponível no momento.

Para Stepanov e Smith (2009) o comportamento de evacuados é um fenômeno complexo, por exemplo, muitas pessoas não saem da região mesmo após serem avisadas. Desse modo, devem ser desenvolvidas políticas de atenção a tais pessoas. Conforme mencionado na segunda diretriz, a defesa civil deve estar atenta à quantidade de pessoas de cada região, assim, será perceptível quando algumas pessoas não tiverem se deslocado. Desse modo, a defesa civil e autoridades, devem deslocar recursos, tais como água e alimentos às regiões afetadas, onde provavelmente há pessoas que não se deslocaram. O deslocamento de água e alimentos deverá ser realizado usando transportes aéreo com a finalidade de evitar congestionamento das vias.

35.6 Conclusão

A base de dados e formação de diretrizes disponibilizadas, permite que os cientistas, analistas de dados, defesa civil e comunidades afetadas por desastres, possam verificar e examinar situações anormais, facilitando criar novas estratégias para evacuação. Neste trabalho, realizamos a mineração de dados referentes ao desastre natural do Havaí, onde foram empregadas análises de comportamento de resposta em eventos de desastres usando dados de mídia social. Através da base de dados construída, alcançamos informações úteis em relação ao comportamento das pessoas e em relação ao desastre, e correlacionamos tais informações à geografia da ilha do Havaí. A construção da base de dados se deu por meio da coleta de tweets, e esta passou por várias fases de limpeza com a finalidade de se tornar uma base de dados útil para este trabalho.

Além da análise de dados das mídias sociais, também foi realizado um estudo sobre a geologia do Havaí, tais como: as áreas que possuem um rifte, ou que possuem dados históricos de abalos sísmicos e lava, bem como, aspectos locais de vias e ruas que podem ser trafegadas afim de escapar da área de risco. Com base nos dados minerados e nos estudos geológicos sobre a ilha, foram criadas diretrizes formalizando o padrão de evacuação mais adequado nesse tipo de situação extrema. Assim, com a finalidade de evitar congestionamentos foram sugeridas rotas de modo que as pessoas fossem distribuídas para determinada região a fim de evitar congestionamentos.

Conforme foi observado, havia uma grande concentração de pessoas na região próxima aos abrigos, escolas, aeroportos e hospitais, tais regiões são as recomendáveis para evacuar antes ou durante uma atividade vulcânica para fins de abrigo.

Em perspectivas futuras, nós coletaremos informações de outros tipos de desastres a fim de validar as diretrizes apontadas neste trabalho. Ainda, como consequência natural do trabalho, nós vislumbramos a construção de um sistema de recomendação de rotas de fuga, que considere diferentes aspectos além da distância, como vagas em abrigos, recursos na rota, alocar pessoas conhecidas em um mesmo abrigo, entre outros.

Mapeamento Sistemático das Técnicas de Reconhecimento Facial para Sistemas de Controle de Acesso

Autores: Felipe de Sousa Santos; Flávio Henrique Duarte de Araújo.

Resumo: Proteger dados, informações e ambientes físicos com técnicas cada vez mais eficazes de biometria tem sido um grande desafio, uma vez que em um mundo globalizado, onde ao mesmo tempo em que se aperfeiçoam sistemas de segurança, também se aperfeiçoam ferramentas de ataques que visam roubo de informações e fraldes, é necessário que se pesquise cada vez mais sobre aplicações de segurança. No entanto, para que se crie e aperfeiçoe novos sistemas de segurança envolvendo reconhecimento facial, é necessário entender através de estudo e análise, quais modelos e técnicas tem sido utilizadas dentro destes campos. Visando esta análise, este trabalho realiza um mapeamento sistemático das técnicas de reconhecimento facial aplicadas em sistemas de controle de acesso através da análise de 19 artigos que utilizam o reconhecimento facial dentro do campo de segurança. Das técnicas mais utilizadas os métodos PCA, LBP, Viola-Jones e SVM tiveram adesão em vários trabalhos pesquisados além de outros modelos que utilizam técnicas baseadas em inteligência artificial. As técnicas utilizadas foram mais simples, uma vez que muitas das aplicações foram feitas para controlar portas e portões em casas, o que exige que sejam utilizadas técnicas mais simples para que sejam comportadas por processadores de baixo custo, como a placa *Raspberry* e consigam ter um bom desempenho. Foi observado que muitos dos trabalhos tiveram dificuldades e desafios com relação ao reconhecimento facial quando as condições de iluminação são baixas, ou dependendo do ângulo do rosto e até mesmo da distância com que a imagem é capturada. Esses desafios representam bons campos de pesquisa para o futuro.

Keywords: Reconhecimento facial; Controle de Acesso; Mapeamento Sistemático;

36.1 Introdução

Ao longo dos últimos anos com o avanço das tecnologias na área de Inteligência Artificial, a utilização da técnica de reconhecimento facial tem se expandido como uma importante área de pesquisa neste segmento

e seu uso se diversificou através do tempo, de forma que hoje, o reconhecimento facial está presente em diversos tipos de aplicações nos mais variados setores da sociedade.

Exemplos de aplicações que utilizam reconhecimento facial se encontram em ambientes corporativos, que utilizam a técnica para emitir mensagens e liberar o acesso cada vez que um rosto é reconhecido. A biometria facial também é utilizada nos smartphones através de tecnologias como o *Face ID do Iphone* e em aplicações semelhantes em sistemas *Android* e que permitem ao usuário desbloquear o aparelho, abrir aplicativos e realizar transações bancárias. Aplicações na área de fiscalização têm feito uso do reconhecimento facial para investigação de crimes. Outro importante setor que tem utilizado as aplicações de biometria facial é a medicina, mais especificamente para a realização de cirurgias plásticas (BOUGUILA; KHOCHTALI, 2020) onde a técnica se tornou auxiliar do cirurgião na reconstrução e correções de anomalias para melhorar a aparência. No entanto, uma das áreas mais populares onde a técnica tem sido amplamente utilizada é na área de segurança e controle de acesso. Um sistema eficaz é de extrema importância para que se garanta a segurança das corporações contra roubos, fraudes, vazamento de dados e outros tipos de crimes tanto a nível físico como a nível digital.

Devido ao contexto atual em que o mundo se encontra enfrentando a pandemia do Novo Coronavírus, sistemas que utilizam biometria digital passaram a ser até mesmo arriscados por conta do alto risco de contaminação por meio do contato com superfícies compartilhadas. Diversas empresas tiveram que mudar seus sistemas de controle de acesso de funcionários devido ao risco de contaminação por meio do contato com essas superfícies. Sistemas de acesso não invasivos passaram a ser requeridos por não necessitarem de contato com superfícies compartilhadas.

Com base no exposto, este estudo visa realizar um mapeamento sistemático envolvendo as técnicas de reconhecimento facial voltadas para sistemas de controle de acesso. Como objetivos específicos, temos:

1. Realizar um levantamento bibliográfico dos principais métodos de reconhecimento facial.
2. Relacionar os métodos encontrados com aplicações específicas de controle de acesso.
3. Análise das técnicas levantadas para identificação das principais vantagens e desvantagens de cada uma.
4. Identificar as técnicas mais adequadas para o reconhecimento facial aplicadas aos sistemas de segurança de acesso.

Os capítulos deste trabalho estão dispostos da seguinte forma: no Capítulo 36.2 são apresentados conceitos referentes à biometria, controle de acesso e reconhecimento facial. Já o Capítulo 36.3 apresenta a metodologia de pesquisa utilizada, que é o mapeamento sistemático, além de explicar sobre o protocolo da pesquisa e como foi feita a condução do trabalho. Logo em seguida, o Capítulo 36.4 apresenta a descrição dos trabalhos encontrados na revisão sistemática, aborda as aplicações de cada um e as técnicas utilizadas no desenvolvimento, também são levantadas algumas discussões a respeito do que foi observado durante a pesquisa. O Capítulo 36.5 apresenta as conclusões do trabalho bem como caminhos para trabalhos futuros.

36.2 Referencial Teórico

Este capítulo apresenta o referencial teórico que serve como base para o entendimento do conteúdo abordado neste trabalho. São apresentados alguns conceitos acerca de controle de acesso, tipos de biometria, reconhecimento facial e suas etapas.

36.2.1 Controle de Acesso

A tecnologia de computadores teve um crescimento gigantesco desde seu surgimento, passando posteriormente pela propagação da internet, que fez com que computadores que no começo eram utilizados apenas em laboratórios e corporações, chegassem nas casas das populações. É raro ver uma tecnologia que tenha se propagado ao redor do mundo inteiro com tamanha velocidade e adesão como esta. O fato é que praticamente todas as áreas e atividades passam pela via computacional, desde compras online, transações

bancárias, ouvir música, cursos online, escrever livros, videoconferências além de uma infinidade de outras atividades e aplicações que utilizam a computação.

Além disso, os computadores trouxeram um vasto número de benefícios em diversos campos e que vão desde as atividades mais triviais até campos de estudos do genoma humano, exploração espacial, inteligência artificial e etc. (BOSWORTH; KABAY; WHYNE, 2014).

Por outro lado, os computadores também são ferramentas utilizadas por aproveitadores para roubar dados e dinheiro, cometer fraudes, calúnias, difamações, falsidade ideológica dentre outros crimes. Muitos destes crimes são cometidos utilizando-se das redes sociais, já outros através de vírus de computador. Por conta destes e outros problemas, o crescimento da segurança dos sistemas de informação foi paralelo ao do próprio campo da computação (BOSWORTH; KABAY; WHYNE, 2014).

Dentro da área de segurança, mais especificamente no meio físico, os sistemas digitais podem ser utilizados em dispositivos ou em máquinas embarcadas para controle de acesso. Estes sistemas embarcados de controle de acesso são uma importante ferramenta do mundo corporativo e são utilizados basicamente para conceder ou restringir acesso ou níveis de acesso para um grupo específico de pessoas. Em sua forma básica, um sistema de controle de acesso eletrônico consiste de uma tranca eletrônica, um leitor (como aqueles utilizados em cartões) e um controlador eletrônico (DEUTSCH, 2018).

Os leitores são montados do lado de fora das portas ou catracas de acesso e é a única parte do sistema de controle de acesso que fica visível ao usuário. Em um sistema de controle de acesso moderno, os leitores são projetados para reconhecer códigos (algo que você conhece), credenciais (algo que você possui) ou biometria (algo que você é) (DEUTSCH, 2018).

Se o controle do acesso for através de código, o usuário digita um número de identificação pessoal (PIN) em um teclado para se identificar, ou se a identificação for feito por credenciais, pode-se usar cartões de acesso ou crachás. O problema destas técnicas de identificação é que são de baixa confiabilidade, podendo ser perdidas, roubadas ou hackeadas e, caso uma pessoa mal intencionada tenha acesso às credenciais ou ao cartão de identificação de alguém, ela poderia obter acesso a um ambiente restrito e, esta falha de segurança poderia resultar em uma situação grave envolvendo roubo de dados, roubo de informações sigilosas, furto de dinheiro e crimes envolvendo fraudes.

36.2.2 Biometria

Na era digital em que nos encontramos, as técnicas biométricas tem sido cada vez mais utilizadas como uma proteção contra o roubo de informações e de identidade. A premissa é que a biometria - uma característica física mensurável ou traço comportamental - é um indicador mais confiável de identidade do que sistemas legados, como senhas e PINs (WEAVER, 2006).

Existem diversos tipos de identificação biométrica, que estão destacadas a seguir:

- Impressão digital: Este tipo de identificação se utiliza de amostras das marcas que existem nas falanges dos dedos utilizando um sensor óptico, ultrassom ou medição de calor.
- Geometria de mão: Na geometria de mão, são mapeadas algumas características como o comprimento de dedos, posicionamento das linhas e o espaçamento dos dedos através de um sensor óptico.
- Reconhecimento da Íris: Nas tecnologias que se utilizam os olhos como ponto de foco, a íris é uma característica única e individual, assim como a biometria digital e pode ser identificada por scanners ou sensores.
- Reconhecimento de Voz: Este método capta as frequências sonoras emitidas pela voz e as transforma em sinais digitais para comparar com os registros existentes na base de dados.

Empresas e corporações passaram então a utilizar diversos destes métodos de identificação, sobretudo a biometria digital para liberar ou negar o acesso nos portões de entrada, também como forma de controlar o horário de entrada e saída dos funcionários, além de bancos que a utilizam nos caixas eletrônicos para transações financeiras.

A Figura 404 mostra um exemplo de um dispositivo de identificação biométrico.



Figura 404 – Imagem de usuário acessando o ponto biométrico.

A identificação biométrica é considerada invasiva e apresenta dificuldades de manutenção como por exemplo, se a pessoa tiver um pouco de suor nos dedos ou umidade devido a quaisquer outros fatores como chuva, isso já é suficiente para causar uma leitura incorreta da digital no leitor óptico.

Outro problema bem mais sério é devido à pandemia do Novo Coronavírus que ocorre no mundo todo. É evidência científica que a transmissão do coronavírus ocorre por gotículas respiratórias, pela via aérea e também pelo contato com superfícies contaminadas.

Este fato torna arriscada a utilização de leitores biométricos para identificação nas corporações, devido ao alto grau de disseminação do vírus, a contaminação pode facilmente ocorrer nas superfícies dos leitores ópticos, o que representa um sério risco à vida.

Com estas questões acima apontadas, este trabalho tratará na seção seguinte sobre o reconhecimento facial, uma categoria de identificação biométrica que tem ganhado bastante espaço nas corporações e na vida cotidiana e tem sido tema de vários trabalhos científicos.

36.2.3 Reconhecimento Facial

O primeiro sistema de reconhecimento facial foi desenvolvido em 1960 com o objetivo de localizar características de uma pessoa individual, mas somente na década seguinte foram criados sistemas automatizados de reconhecimento facial como dito em (BRAGA, 2013). Desde então através de novas pesquisas, foram aparecendo novas tecnologias que aperfeiçoaram a técnica, que passou a ser utilizada em diversos setores que vão desde a área comercial até a investigações criminais como mostrado na Tabela 259.

O reconhecimento facial é uma técnica biométrica em que a face humana é identificada em uma imagem digital por meio da análise e comparação de padrões. O rosto humano é carregado de diversas informações como identidade, gênero, raça, expressões faciais e estados de humor. A análise do rosto humano e do seu comportamento facial envolvem áreas multidisciplinares como psicologia, neurociência e engenharia (ARYA; PRATAP; BHATIA, 2015).

Ao contrário dos outros métodos de identificação biométrica, o reconhecimento facial não é invasivo pois não requer necessariamente a interação ou cooperação direta da pessoa com o sistema, sendo então um método mais discreto e portanto eficiente para aplicações de vigilância. Ele também pode ser baseado em características físicas estáticas e em características dinâmicas da face de uma pessoa, o que o torna adequado para biometria comportamental.

O reconhecimento facial encontra também algumas dificuldades como por exemplo, ao ser utilizado em sistemas livres de restrições, há problemas que envolvem o posicionamento da cabeça, iluminação do

Tabela 259 – Áreas de aplicação dos métodos de reconhecimento facial. Fonte: alterado de (TASKIRAN; KAHRAMAN; ERDEM, 2020)

Áreas de Aplicação	Aplicações específicas
Segurança, Saúde	Segurança da informação, autenticação de usuários, Login em dispositivos eletrônicos, Interação humano-robô, acesso a edifícios, Reconhecimento de rostos em vídeos de vigilância, Casas inteligentes, carros inteligentes, Assistentes robóticos
Leis	Monitoramento de fronteiras, análise de eventos ilegais, rastreamento de pessoas suspeitas, Passaportes, carteiras de identidade nacionais, carteira de motorista, Imigração
Entretenimento, Educação, Marketing	Jogos, realidade virtual, Gerenciamento de fotos, Análise de vídeo, recuperação de vídeo, Aprendizagem online, acompanhamento de alunos, engajamento de usuários, Campanhas publicitárias, moderação de redes sociais

ambiente e variações relacionadas à expressão facial. Há problemas também que envolvem a mudança de aparência como o uso de maquiagem, pelos faciais e a utilização de acessórios (óculos, bonés, lenços). Também existe a dificuldade quanto à semelhança de dois indivíduos, por exemplo irmãos gêmeos (ARYA; PRATAP; BHATIA, 2015).

O reconhecimento facial é abordado na literatura como um problema de identificação ou de verificação. A identificação é classificada como um problema de correspondência 1:N. Um rosto desconhecido é comparado com todos os rostos no banco de dados que contenham identidades conhecidas e a partir disso, uma decisão é tomada após o processamento de todas as comparações. Caso a imagem seja localizada no banco de dados, a tarefa é classificada como 'conjunto fechado', ou caso não seja localizada, é classificada como 'conjunto aberto'. A verificação facial é classificada como um problema de correspondência 1:1, ou seja, a identidade da face analisada é confirmada ou rejeitada após a comparação com a face armazenada no banco de dados.

36.2.4 Principais etapas do reconhecimento facial

Segundo (TASKIRAN; KAHRAMAN; ERDEM, 2020) o processo de reconhecimento facial é feito tradicionalmente em seis etapas, que são:

1. A imagem de entrada ou vídeo do rosto é adquirido.
2. O módulo *anti-spoofing* facial garante a segurança do sistema ao realizar detecção de ataques *spoofing* (por meio de testes que verificam se a imagem contém algum vírus malicioso).
3. É feita a detecção do rosto ou de pontos de referência na imagem ou em cada quadro de vídeo.
4. É executado o pré-processamento na imagem ou vídeo, que pode consistir em alinhamento, seleção de quadro de vídeo, redução de ruído, aumento de contraste ou operações semelhantes.
5. É feita a extração de características faciais da imagem ou vídeo.
6. Por último, é realizado o reconhecimento facial

Na Figura 405 é mostrada uma representação das etapas deste processo.

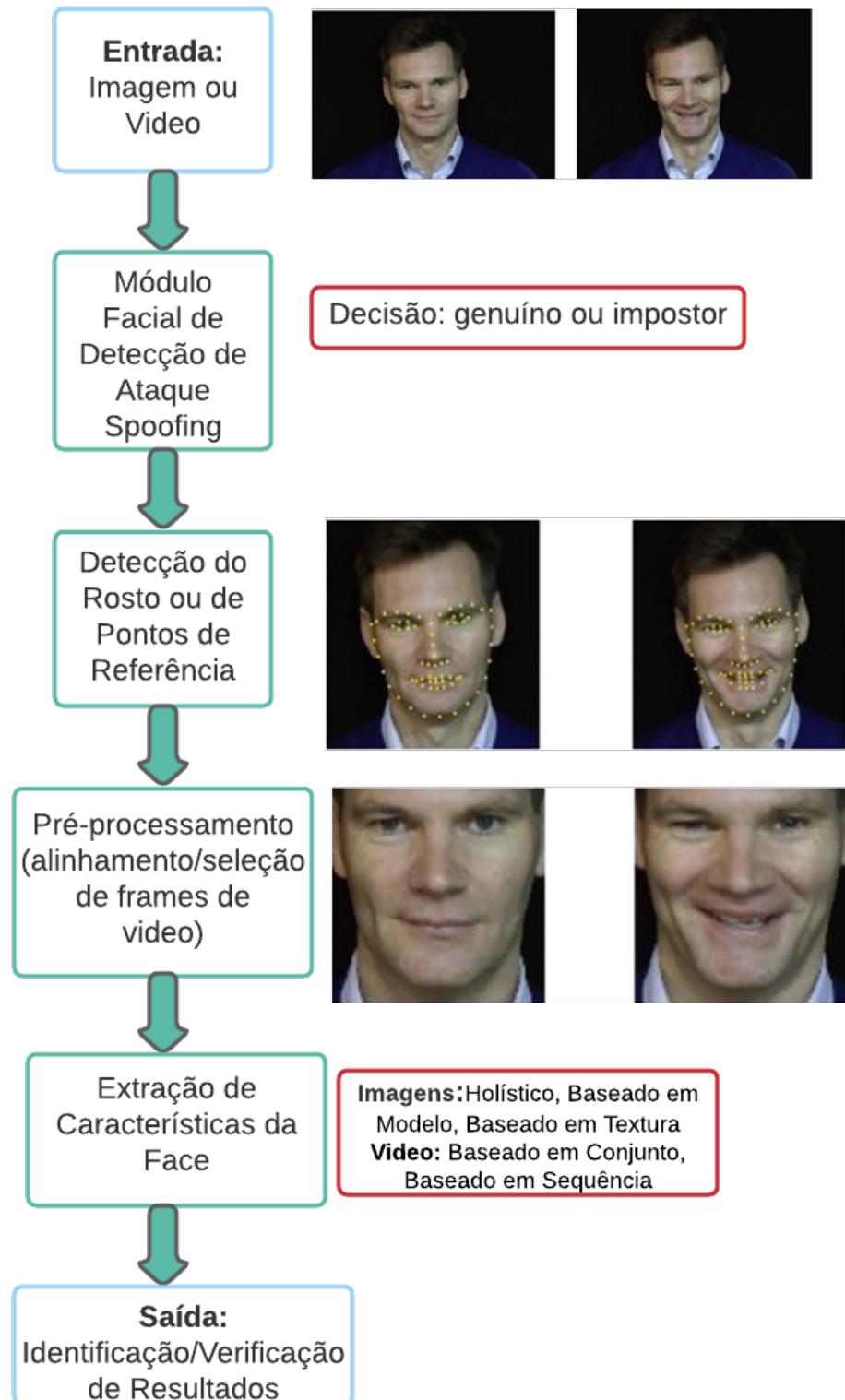


Figura 405 – As principais etapas dos sistemas de reconhecimento facial. Fonte: alterado de (TASKIRAN; KAHRAMAN; ERDEM, 2020)

36.2.5 Detecção Facial

Uma das etapas de um sistema que utiliza a biometria facial é a detecção de faces em uma imagem ou vídeo. É nesta etapa que são descartadas informações desnecessárias. Dependendo da imagem captada, podem existir vários rostos e, neste caso, o algoritmo irá detectar todos eles e extrair as faces extras para analisá-las individualmente. O algoritmo deve ser robusto para poder detectar a imagem com diferentes poses, escala e iluminação e eliminar o fundo o máximo possível. Outro ponto importante é que caso a entrada seja um vídeo, o fator tempo é levado em consideração e assim, é crucial que o algoritmo tenha uma boa velocidade de resposta, levando em conta que a detecção é feita em tempo real (BRAGA, 2013).

36.2.6 Aquisição das imagens

Para a avaliação do desempenho dos algoritmos utilizados no processo de reconhecimento facial, são utilizadas bases de dados que contenham imagens ou vídeos, dependendo do algoritmo responsável pelo processamento. Segundo (TASKIRAN; KAHRAMAN; ERDEM, 2020), os primeiros trabalhos foram realizados com bancos de dados de pequena escala, um dos primeiros bancos de dados baseados em imagens é ORL contendo 400 imagens de 10 indivíduos. Da mesma forma, um dos primeiros bancos de dados de faces baseados em vídeo lançado em 1997 consistia em 70 vídeos de 40 assuntos.

Atualmente, os bancos de dados utilizados para o reconhecimento de rostos se tornaram de grande escala, com milhares de imagens registradas sob condições não controladas além de milhares de vídeos gravados. Bases de dados em vídeo como IJB-A, JB-B, UMDFaces, IJB-C, YouTubeFace, PaSCVideo e bases de dados de imagens como CPLFW, MS-Celeb-1M(Ch3), VGG Face2, MegaFace, VGG Face e CASIAWebFace tem sido utilizados nos anos mais recentes a partir de 2010 (TASKIRAN; KAHRAMAN; ERDEM, 2020).

36.2.7 Filtragem anti fraude

Uma das fragilidades do reconhecimento facial é sua vulnerabilidade a ataques de *spoofing* feitos por fotos, vídeos ou máscaras 3D (TASKIRAN; KAHRAMAN; ERDEM, 2020). O termo *spoofing* é um jargão utilizado em tecnologia da informação para falsificação. Ataques *spoofing* acontecem quando um cibercriminoso finge ser uma pessoa ou uma rede conhecida para obter acesso indevido a informações a afim de roubar dados, invadir sistemas e espalhar *malwares*. Ataques como esse são comuns na gravação dos dados biométricos, extração de características ou na fase de decisão.

O *anti-spoofing* no reconhecimento facial geralmente está atrelado à tarefa de verificação de veracidade ou à detecção de outros ataques, o que pode ser feito através da identificação de movimentos fisiológicos, como movimentos da boca, piscar de olhos, alterações de expressões faciais e movimentos da cabeça.

Após a detecção do rosto e do pré-processamento, a próxima etapa no reconhecimento facial é a extração de características. A finalidade é localizar regiões da imagem que possuem características significativas. Estas regiões do rosto podem ser distinguidas por texturas, formas, intensidades, propriedades estatísticas e outros (BRAGA, 2013). Os pontos de referência do rosto (sobrancelhas, boca, canto dos olhos e canto do nariz) são demarcados para serem utilizados no alinhamento. Na Figura 405 é mostrado um exemplo de demarcação facial com pontos amarelos que estimam a área do rosto.

Na fase de extração de características, diferentemente da detecção facial, onde todas faces são categorizadas numa única classe, procura-se localizar pontos discriminantes que permitam que matematicamente, cada face seja diferenciada das demais faces. Essa tarefa é realizada utilizando descritores para extrair as características que representam uma imagem.

36.2.8 Reconhecimento

A última fase do reconhecimento facial consiste em analisar se as características faciais coletadas na entrada e posteriormente representadas por um modelo matemático condizem com as características de alguma face contida no banco de dados. O reconhecimento pode ser aplicado de dois modos: identificação ou verificação. Na verificação o sistema faz uma checagem para validar se o indivíduo realmente é quem diz ser; por exemplo, caso um indivíduo esteja portando um documento como um passaporte e o apresenta

como sua identidade. Então o sistema irá avaliar se o indivíduo é o mesmo do passaporte. A verificação é testada numa correspondência um para um, onde o rosto de entrada é testado somente contra um rosto da base de dados. (BRAGA, 2013). Na identificação, a correspondência é de um para vários, pois a face de entrada é comparada com todas as contidas no banco de dados a procura da face correspondente.

O capítulo seguinte trata da metodologia de mapeamento sistemático e da condução da pesquisa neste trabalho.

36.3 Método de Pesquisa: Mapeamento sistemático

Uma revisão sistemática da literatura é um meio de identificar, avaliar e interpretar as pesquisas relevantes disponíveis para uma questão de pesquisa específica (KITCHENHAM, 2004b). A metodologia utilizada no trabalho consiste na inserção de *strings* de buscas em bases de dados. A revisão sistemática tem diversas razões que a fazem necessária no meio científico, dentre elas:

- Para reunir as evidências existentes sobre um tratamento ou tecnologia, por exemplo, para reunir a evidência empírica dos benefícios e limitações de uma metodologia específica.
- Identificar quaisquer lacunas nas pesquisas atuais, a fim de sugerir áreas para futuras investigações.
- Fornecer uma estrutura/histórico para posicionar adequadamente novas atividades de pesquisa. Além disso, revisões sistemáticas também podem ser realizadas para examinar até que ponto a evidência empírica apoia/contradiz hipóteses teóricas, ou mesmo para auxiliar a geração de novas hipóteses.

36.3.1 Planejamento: Definição de Protocolo

No planejamento do trabalho, foi feita a definição do protocolo de pesquisa a ser utilizado para a realização do trabalho. Nesta definição foram levantados os seguintes pontos para serem observados:

1. Questões de pesquisa: o objetivo deste estudo é realizar um mapeamento sistemático nos trabalhos que utilizaram o reconhecimento facial aplicado à área de segurança para verificar as seguintes questões:
 - a) Qual a metodologia mais utilizada para uma aplicação de reconhecimento facial aplicada em controle de acesso?
 - b) Quais os algoritmos mais utilizados em cada uma das etapas da metodologia?
2. Palavras-chave: a escolha das palavras chave foi feita de acordo e com o objetivo de responder às questões de pesquisa feitas identificando os principais métodos e técnicas utilizadas.
Dessa forma, foram utilizadas as seguintes palavras-chave: "face", "recognition", "access" e "control". Para abranger um número maior de trabalhos, foram utilizadas palavras sinônimas dos termos principais da pesquisa como "facial", "identification", "verification" e "authentication" que são comumente utilizados na literatura.
3. *Strings* de Busca: baseado nas palavras-chaves apresentadas anteriormente, foram definidas uma série de *strings* a serem utilizadas nas bases de busca. As *strings* definidas são listadas na Tabela 260.
4. Fontes de Pesquisa: para a coleta dos trabalhos, foram utilizados repositórios de dados que contém pesquisas nas áreas de reconhecimento facial aplicadas ao controle de acesso. As bases de dados utilizadas na coleta dos artigos foram IEEE¹, Science Direct² e ACM³.
5. Critérios de Seleção: para coletar os estudos relacionados ao tema proposto, foram definidos critérios de Inclusão (I) e Exclusão (E) como mostrados na Tabela 261.

¹ <https://www.ieee.org>

² <https://www.sciencedirect.com>

³ <https://dl.acm.org>

Tabela 260 – *Strings* de busca nas bases de dados

Critérios de Inclusão e Exclusão	
I	<i>"face recognition" and "access control"</i>
II	<i>"facial recognition" and "access control"</i>
III	<i>"face identification" and "access control"</i>
IV	<i>"facial identification" and "access control"</i>
V	<i>"face verification" and "access control"</i>
VI	<i>"facial verification" and "access control"</i>
VII	<i>"face authentication" and "access control"</i>
VIII	<i>"facial authentication" and "access control"</i>

Tabela 261 – Critérios de Inclusão e Exclusão

Códigos de I/E	Critério
I.1	Relaciona reconhecimento facial com controle de acesso
I.2	Área da Computação
E.1	Artigo escrito em inglês
E.2	Título e resumo fora dos critérios de pesquisa
E.3	Todo o texto fora dos critérios

Como o objetivo é localizar os trabalhos que relacionem reconhecimento facial com controle de acesso, foi definido o critério de inclusão I.1. Para deixar o trabalho mais específico, foi definido o critério I.2, para que os trabalhos sejam da área de computação. Além disso, na avaliação dos trabalhos, foram observados artigos que descrevem uma metodologia de reconhecimento facial.

O critério de exclusão E.1 foi definido pelo motivo dos trabalhos mais relevantes desta área de pesquisa estarem escritos em inglês, dessa forma trabalhos escritos em outras línguas foram excluídos. Os critérios E.2 e E.3 foram definidos afim de eliminar os artigos não relacionados ao objetivo deste trabalho.

36.3.2 Condução do Mapeamento Sistemático

Na condução do mapeamento foram definidas algumas etapas para a realização deste trabalho: 1. Definição dos critérios de busca dos estudos primários; 2. Seleção dos artigos; 3. Extração dos dados contidos nos trabalhos. Nas subseções a seguir estão descritas as fases de condução.

36.3.3 Consulta nas Bases de Dados

Nesta etapa do processo ocorreu a coleta dos artigos científicos nas bases de dados selecionadas anteriormente. Nesta fase o trabalho consistiu em utilizar as *strings* de busca definidas na seção anterior e inseri-las no motor de busca das bases de dados. A partir daí, foram aplicados filtros para especificar a busca.

No primeiro momento quando as *strings* de busca foram inseridas, a base de dados foi configurada para localizar os termos das *strings* em qualquer parte do documento, porém mesmo utilizando os operadores lógicos AND e OR para refinar a busca, foram retornados um range muito grande de artigos. Embora esses artigos tivessem alguns dos termos da pesquisa, muitos deles se tratavam de assuntos mais abrangentes e que não condiziam com o objetivo aqui tratado. Inicialmente foram coletados alguns destes artigos e pela análise do título era possível identificar que alguns se tratavam do tema procurado. No entanto, um alto número de artigos com temas não relacionados ao objetivo desse mapeamento também foram retornados, mesmo filtrando apenas os trabalhos nos anos mais recentes. Por conta disto, foram feitas alterações nas configurações de busca das bases de dados para que ela filtrasse pelos termos da pesquisa apenas no título dos trabalhos publicados para que houvesse uma maior especificidade no resultado da busca.

Com relação ao ano, consideramos o mapeamento dos trabalhos que foram feitos nos últimos dez anos, ou seja, à partir de 2011 para cá. Com essa restrição, pudemos ver quais tecnologias vêm sendo utilizadas nas pesquisas mais recentes e, também, foi à partir desta década que a tecnologia de deep learning passou a ser mais amplamente utilizada nos estudos envolvendo inteligência artificial e aprendizado de máquinas. Na Tabela 262 encontram-se a quantidade de artigos retornados na busca inicial e após a filtragem dos termos apenas pelo título do artigo.

Tabela 262 – Quantidade de artigos retornados após busca na base de dados

Base de Dados	Artigos Encontrados na Busca Inicial	Artigos Encontrados Após a filtragem na Base de Dados
IEEE	8000	18
Science Direct	818	2
ACM	285	1

36.3.4 Triagem dos Trabalhos

Após a etapa de pesquisa dos trabalhos nas bases de dados, ocorreu a filtragem dos artigos selecionados por meio da leitura. O objetivo foi ler os trabalhos para verificar inicialmente se estavam dentro dos parâmetros anteriormente definidos e extrair os dados buscados.

Dentre os trabalhos selecionados, foi necessário eliminar dois deles da seleção pois, um deles não definia nenhuma metodologia de reconhecimento facial aplicada ao controle de acesso e o outro, apesar de inicialmente ter passado pelos filtros de pesquisa iniciais, foi observado que o título e o abstract estarem escritos em inglês porém, todo o restante do trabalho foi escrito no idioma chinês, inviabilizando a leitura e a compreensão do trabalho.

Na Tabela 263 encontram-se os artigos selecionados:

Tabela 263 – Artigos selecionados após a leitura

AUTOR	TÍTULO
(IBRAHIM; ZIN, 2011)	<i>Study of automated face recognition system for office door access control application</i>
(PAN, 2012)	<i>Research and Implementation of Access Control System Based on RFID and FNN-Face Recognition</i>
(TALEB; OUIS; MAMMAR, 2014)	<i>Access control using automated face recognition: Based on the PCA LDA algorithms</i>
(ZELJKOVIć et al., 2014)	<i>Personal access control system using moving object detection and face recognition</i>

AUTOR	TÍTULO
(MANTECÓN et al., 2015)	<i>Access control based on visual face recognition using Depth Spatiograms of Local Quantized Patterns</i>
(CHANG; LIAO, 2015)	<i>Combination of RFID and face recognition for access control system</i>
(SAHANI et al., 2015)	<i>Web-based online embedded door access control and home security system based on face recognition</i>
(YIN et al., 2016)	<i>Electronic Door Access Control Using MyAccess Two-Factor Authentication Scheme Featuring Near-Field Communication and Eigenface-Based Face Recognition Using Principal Component Analysis</i>
(COLE; EL-KHATIB, 2017)	<i>A Privacy Enhanced Facial Recognition Access Control System Using Biometric Encryption</i>
(BAKSHI; PRABHU, 2017)	<i>Face recognition system for access control using principal component analysis</i>
(LIN; HUANG, 2017)	<i>The implementation of the visitor access control system for the senior citizen based on the LBP face recognition</i>
(NAG; NIKHILENDRA; KALMATH, 2018)	<i>IOT Based Door Access Control Using Face Recognition</i>
(AI; CHENG, 2018)	<i>Research on embedded access control security system and face recognition system</i>
(SHAVETOV; SIVTSOV, 2020)	<i>Access Control System Based on Face Recognition</i>
(ORNA; BENÍTEZ; PÉREZ, 2020)	<i>A Low-Cost Embedded Facial Recognition System for Door Access Control using Deep Learning</i>
(PRAMANA; LESTYEKA; AMIRUDDIN, 2020)	<i>Development of a Secure Access Control System Based on Two-Factor Authentication Using Face Recognition and OTP SMS-Token</i>
(DEWANTARA; BACHTIAR; LANTANG, 2020)	<i>Door Access Control based on Illumination Invariant Face Recognition in Embedded System</i>
(WAHYUDONO; OGI, 2020)	<i>Implementation of Two Factor Authentication based on RFID and Face Recognition using LBP Algorithm on Access Control System</i>
(RAMESHWARI et al., 2021)	<i>Automated access control system using face recognition</i>

36.3.5 Extração dos Dados do Mapeamento Sistemático

Além das técnicas e algoritmos utilizados nos trabalhos selecionados, foram extraídas outras informações que também são importantes no escopo deste trabalho.

Dentre estas outras informações consideradas importantes estão:

1. Qual a aplicação de controle de acesso foi desenvolvida
2. Qual foi a base de imagens utilizada para criação e avaliação
3. Qual a tecnologia utilizada no controle de acesso
4. Qual a metodologia empregada na construção do trabalho

Estas informações são importante para saber nos últimos dez anos que tipo de aplicações de segurança foram desenvolvidas com a tecnologia de controle de acesso e também quais metodologias foram empregadas em cada uma delas. No capítulo seguinte são apresentados os resultados coletados após a leitura dos trabalhos e feitas algumas discussões a respeito deles.

36.4 Resultados

Foram selecionados 19 artigos que se enquadram nas questões de pesquisas anteriormente apresentadas e se localizam entre os anos de 2011 à 2021 de forma que todos executam uma aplicação de controle de acesso utilizando técnicas de reconhecimento facial.

Na seção a seguir é feita uma descrição dos trabalhos onde é citada em cada um a aplicação que foi desenvolvida e a tecnologia utilizada em cada um dos trabalhos.

36.4.1 Descrição dos trabalhos

- **1 - Combination of RFID and face recognition for access control system**

No trabalho desenvolvido por ([CHANG; LIAO, 2015](#)) foi desenvolvido um sistema de controle de acesso utilizado em uma empresa multinacional para controlar quem tem autorização de entrar no local. Este sistema integra a técnica de reconhecimento facial com a identificação por radio frequência (RFID).

Na metodologia proposta, o funcionário obrigatoriamente deve apresentar o cartão RFID e, após a apresentação do cartão é feito o processo de reconhecimento facial que é iniciado com a etapa de detecção. Na etapa de detecção facial foi utilizada a Tecnologia Robusta de Detecção Facial, a *Robust Real-Time Face Detection*.

Na etapa seguinte é executado o processo de extração da face, onde é feita a remoção do fundo e do penteado do rosto, deixando apenas a face interna. É realizado então um processo de normalização com o objetivo deixar as imagens detectadas com tamanho fixo. Na próxima etapa é utilizado o algoritmo *Local Binary Pattern* (LBP) para extrair características de textura e com isso reduzir a interferência da iluminação. A Figura 406 mostra os resultados do método após a normalização e a utilização do LBP, respectivamente.

As imagens LBP passam por mais um processo utilizando um método chamado *Complex Wavelet Structural Similarity* que tem o objetivo de medir a similaridade entre duas images. A última etapa é o registro de imagens, que é o processo de alinhar duas imagens tomadas em momentos diferentes na mesma coordenação. O algoritmo utilizado para isto é o *Speeded Up Robust Features* (SURF), que pesquisa os pontos de características na imagem integral. Nos testes realizados foram utilizadas imagens de 16 homens e 4 mulheres entre 22 e 26 anos, e as imagens foram capturadas pela *Logitech HD 720p web cam C310* sob um ambiente de iluminação estável. Foram realizados dois testes, um com o RFID pessoal de cada participante e outro com o RFID trocado. No primeiro teste o sistema reconheceu corretamente em 95% das vezes. No segundo experimento com o RFID trocado, foram realizados 380 testes e o sistema conseguiu uma acurácia de 100% no reconhecimento sob condições estáveis de iluminação.

- **2 - Access control based on visual face recognition using Depth Spatiograms of Local Quantized Patterns**

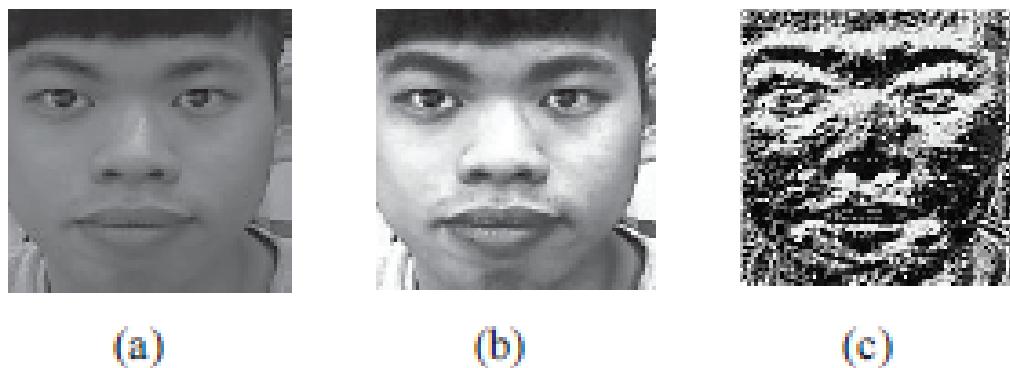


Figura 406 – (a) face interna, (b) resultado após a equalização, e (c) face LBP . Fonte: extraído de (CHANG; LIAO, 2015)

O trabalho de (MANTECÓN et al., 2015) consiste no desenvolvimento de um novo algoritmo biométrico de identificação facial para aplicações de controle de acesso. Nele é apresentado um descritor de características chamado *Depth Spatial Local Quantized Pattern descriptor (DS-LQP)*.

O DS-LQP utiliza as chamadas imagens de profundidade, que são imagens que se utilizam de representações de distâncias entre objetos na própria imagem a partir de um ponto de referência, fornecendo detalhes de profundidade com base em quanto perto ou longe se está de uma determinada parte da imagem.

A escolha por imagens de profundidade no trabalho é pelo motivo de estas apresentarem uma estruturação mais rica da imagem e terem maior robustez contra variações das condições de iluminação.

O DS-LQP proposto no trabalho é utilizado como entrada do algoritmo Máquinas de Vetores de Suporte (SVMs - *Support Vector Machines*), que é uma técnica de aprendizado de máquina utilizada para analisar dados e classificá-los, neste caso, o objetivo é classificar rostos e à partir do resultado do processamento, liberar ou não o acesso das pessoas à uma área restrita. A Figura 407 mostra um diagrama de blocos com a cadeia de processos da aplicação.

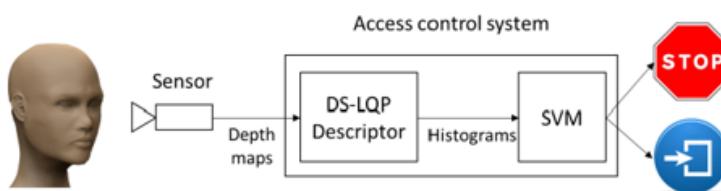


Figura 407 – Diagrama de blocos da aplicação de controle de acesso. Fonte: extraído de (MANTECÓN et al., 2015)

Nos testes foi utilizado o banco de dados de face de profundidade *HRRFaceD* com 200 imagens que foram divididos em 160 imagens de treinamento e 40 imagens de teste (80% e 20%, respectivamente). Os testes foram realizados em comparação com outros algoritmos, entre eles uma variação do descritor DS-LQP chamado H-LQP, também com o método original baseado em (LBP) e um método baseado em Transformação de Características Variáveis em Escala (SIFT - *Scale-invariant feature transform*). O DS-LQP conseguiu um resultado de 80% em comparação aos outros métodos, o que representou o melhor resultado entre os métodos comparados levando em consideração que o banco de dados contém imagens de pessoas em diferentes poses.

- **3 - IOT Based Door Access Control Using Face Recognition**

Em (NAG; NIKHILENDRA; KALMATH, 2018) foi feito um sistema de controle com reconhecimento facial para ser utilizado em portas, fazendo uso de um processador *Raspberry Pi* integrado com uma aplicação de

notificação via e-mail e Telegram. O processador *Raspberry* contém um sensor que inicialmente identifica a presença de uma pessoa na porta. Quando o humano é detectado, a câmera Pi captura a imagem da pessoa e envia para o usuário remoto através de email.

O algoritmo responsável pela detecção é o *Haar Cascade*, que é um classificador baseado em modelos cascata, que utilizam árvores de decisão que analisam conjuntos de atributos em cada níveis. Na Figura 408 é mostrado o protótipo do sistema.

Feito o reconhecimento, o usuário remoto pode controlar o mecanismo de travamento e destravamento da porta utilizando o aplicativo do Telegram.

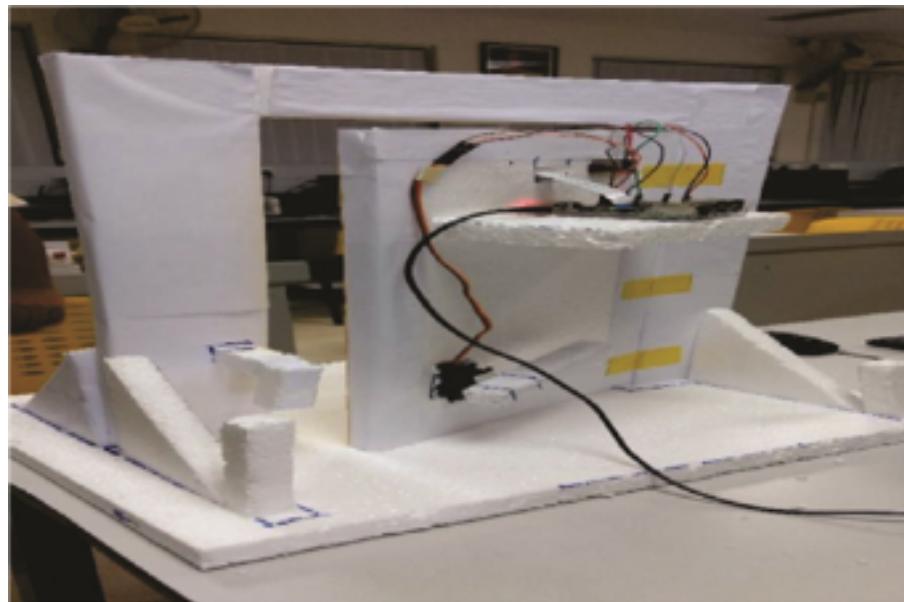


Figura 408 – Protótipo do Sistema de Acesso. Fonte: extraído de ([NAG; NIKHILENDRA; KALMATH, 2018](#))

Não foi especificado como os testes foram realizados e nem quantas imagens foram utilizadas. Também não foi especificado a acurácia do sistema, apenas foi dito que a eficiência do sistema foi analisada em termos de taxa de reconhecimento facial e que teve um ótimo desempenho.

• 4 - Research and Implementation of Access Control System Based on RFID and FNN-Face Recognition

Em ([PAN, 2012](#)) é introduzido um sistema inteligente que realiza a detecção facial baseada em modelos de pele, processamento de cartões RFID e reconhecimento facial baseado em Rede Neural Direta (*FNN-Feedforward Neural Network*). No sistema proposto, a pessoa que quer acessar uma área restrita inicialmente utiliza seu cartão de RFID para que o sistema leia as informações. Enquanto isso a câmera de vídeo é iniciada para tirar fotos da pessoa.

As informações de identidade no RFID são comparadas com as informações do banco de dados e os dados da face correspondente serão obtidos. Se as informações de identidade e os dados da face forem todos compatíveis com as informações do banco de dados, a pessoa será liberada, caso contrário não poderá entrar.

O modelo proposto é dividido em 5 subsistemas: subsistema de aquisição de informações, subsistema de processamento de dados, subsistema de armazenamento de informações e subsistema de gerenciamento de informações. No subsistema de aquisição de informações os dados originais do usuário são coletados, introduzidos no cartão RFID junto com a face do usuário. Neste módulo também é feita a identificação dos dados do cartão no caso de usuários já previamente cadastrados no banco de dados.

No subsistema de processamento de dados são realizadas as etapas de detecção da face, reconhecimento da face e comunicação com o sistema de armazenamento. Na detecção facial é utilizado um método próprio desenvolvido pelos autores que utiliza de cores RGB e faz uma varredura para analisar a distribuição dos

pixels da cor da pele para saber se um pixel é um ponto de cor de pele. Depois disso a imagem colorida é transformada em imagem binária, o fundo removido e é feita uma varredura para detectar os pontos de pixels que pertencem à face, após este procedimento a imagem é detectada.

No módulo de reconhecimento facial, é buscada no banco de dados as informações de identidade correspondente às passadas pelo cartão RFID. Então são extraídas as informações de cinco imagens faciais e feito o treinamento da rede neural direta FNN. Em seguida, as características da face utilizada no cartão RFID são coletadas e utilizadas como entrada da rede neural. Se o reconhecimento for aprovado a identidade é confirmada e o acesso liberado, caso contrário a imagem não é reconhecida e o acesso negado.

Nas Figuras 409, 410 e 411 é mostrada a interface de operação com três resultados possíveis, imagem corretamente detectada, imagem não consta no banco de dados e imagem não corresponde com a cadastrada no banco.

Não foram informados quantas imagens foram utilizadas no experimento e nos resultados obtidos, foi apenas informado que o sistema conseguiu uma taxa de 92% de acerto.



Figura 409 – Imagem corretamente detectada. Fonte: extraído de ([PAN, 2012](#))

• 5 - Access Control System Based on Face Recognition

No trabalho de ([SHAVETOV; SIVTSOV, 2020](#)) é feito um estudo onde é projetado um sistema de controle de acesso que realiza uma verificação adicional de segurança em que é verificada a vivacidade da imagem, minimizando o risco de controle não autorizado.

Inicialmente a imagem coletada passa por uma fase de pré-processamento onde são feitos poucos ajustes de brilho e luminosidade na imagem para que não atrapalhe o módulo de verificação de vivacidade.

O estudo realizou na etapa de detecção um estudo comparando algumas abordagens, incluindo o algoritmo Viola Jones e o algoritmo Redes Convolucionais em Cascata Multi-Tarefa (MTCNN - *Multi-task Cascaded Convolutional Networks*) para saber qual tem o melhor desempenho. Foi utilizado o banco de dados *Label Faces in the Wild* (LWF) que contém 13233 imagens de 5749 pessoas diferentes. No teste o Viola Jones teve um maior desempenho em termos de tempo na detecção de imagens.

Para o reconhecimento facial a abordagem testou 5 algoritmos que tiveram desempenhos parecidos, podendo ser utilizado qualquer um deles. Porém não foi descrito qual deles foi o escolhido para utilização no sistema real. A Tabela 264 mostra o resultado dos testes com os algoritmos utilizados.

Foi criado um módulo para detectar ataques *spoofing*, onde é feita uma verificação de vivacidade que filtra tentativas de ataque utilizando técnicas de falsificação, como mostrar o rosto da pessoa permitida a partir de uma foto impressa ou da tela de um dispositivo eletrônico. É utilizado um classificador de detecção



Figura 410 – Imagem não consta no banco de dados. Fonte: extraído de (PAN, 2012)



Figura 411 – Informações da face não correspondem. Fonte: extraído de (PAN, 2012)

Tabela 264 – Algoritmos testados no reconhecimento

Nome	Conjunto de Treinamento	Precisão
FaceNet	Google (500M,10M)	99.63
DeepFace	Facebook (4.4M,4K)	97.3
DeepID3	CelebFaces+ (0.2M,10K)	99.53
Cosface	CASIA-WebFace (0.49M,10K)	99.33
Arcface	MS-Celeb-1M (3.8M,85K)	99.83

de ataque de apresentação baseado na diferença entre histogramas de objetos reais e falsificados em espaços de cores diferentes. Para reforçar a precisão do sistema, foi utilizada a classificação baseada não apenas na estrutura atual, mas também nas anteriores. O sistema pode ser utilizado de forma automática, porém os

autores recomendam que o mesmo seja supervisionado para ter maior garantia na segurança.

- **6 - Access control using automated face recognition Based on the PCA - LDA algorithms**

No trabalho de (TALEB; OUIS; MAMMAR, 2014) é proposto um sistema de controle de acesso de veículos em estacionamento através de uma câmera instalada na entrada do estacionamento. O sistema se baseia nos algoritmos Análise de Componentes Principais (PCA - Principal Component Analysis) e Análise Discriminatória Linear (LDA - Linear Discriminant Analysis).

O sistema inicialmente detecta um objeto em movimento e, em seguida o resultado deste processamento é enviado para a etapa de detecção facial, que é feita utilizando o algoritmo Viola Jones que é baseado no algoritmo de aprendizagem Adaboost utilizando características Haar que detectam faces com alta precisão.

A extração facial no sistema proposto é feito utilizando PCA e após é utilizado o LDA para classificar os dados. Nos testes foi utilizado o banco de dados de faces GTAV, que inclui um total de 44 pessoas sob três iluminações diferentes (ambiente ou luz natural, fonte de luz forte a partir de um ângulo de 45º e, finalmente, uma fonte de luz quase frontal de força média). A resolução das imagens é de 240x320 e elas estão em formato BMP. Na Figura 412 encontra-se o resultado do reconhecimento após a utilização dos algoritmos.

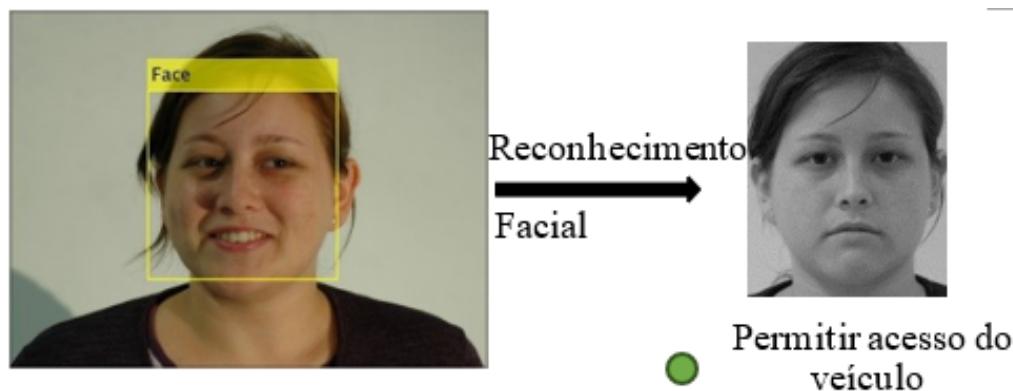


Figura 412 – Resultado do reconhecimento facial. Fonte: alterado de (TALEB; OUIS; MAMMAR, 2014)

Nos testes do sistema foi obtida uma taxa de 92% de acertos no reconhecimento das faces quando essas estavam na posição frontal. Porém quando são utilizadas outras poses a taxa de acerto cai consideravelmente. A variação de poses atualmente é um dos grandes problemas a serem resolvidos no reconhecimento facial.

- **7 - Study of automated face recognition system for office door access control application**

No trabalho de (IBRAHIM; ZIN, 2011) foi desenvolvido um sistema automatizado de reconhecimento facial para o controle de acesso às portas de um escritório. No sistema proposto é utilizado PCA e uma rede neural.

O desenvolvimento do sistema automatizado de reconhecimento facial é feito em duas etapas típicas de reconhecimento facial, que são a etapa de treinamento e a etapa de avaliação. Na primeira etapa, é capturado um número específico de imagens de treinamento do candidato de face. As características são extraídas da imagem das faces frontais humanas utilizando a análise dos componentes principais. O sistema então aprende sobre as características extraídas e armazena no banco de dados. No segundo estágio, o sistema reconhecerá novas faces de forma não supervisionada. A Figura 413 mostra a interface gráfica do sistema.

Foram utilizados dois tipos de banco de dados de imagens para treinar o sistema. O primeiro tipo consiste em imagens de rosto capturadas e recortadas de pessoas que irão ser reconhecidas. São utilizadas dez imagens para cada pessoa onde as variações das posições frontais da face são de vinte graus de rotação para a esquerda e direções perpendiculares à câmera. Estas imagens são capturadas usando a câmera, cortadas e depois são treinadas no sistema e mantidas dentro do banco de dados de faces do sistema. O



Figura 413 – Interface do Sistema. Fonte: extraído de (IBRAHIM; ZIN, 2011)

segundo banco de dados consiste das imagens frontais ad-hoc que são capturadas instantaneamente usando a câmera do sistema. São utilizadas quantidades e imagens com características semelhantes às do primeiro banco de dados.

Para testar o sistema foram considerados três fatores de avaliação que são a iluminação, a distância e o ângulo de rotação da face. A Figura 414 mostra um exemplo de captura da imagem do usuário com a câmera utilizada no sistema.

A partir dos resultados obtidos, o sistema desenvolvido foi capaz de reconhecer cerca de 78% das diferentes posições de orientação do rosto do sujeito a 40 cm da câmera. Entretanto, o sistema não conseguiu realizar o reconhecimento de face quando a distância da face do sujeito é inferior a 10 cm. Isto se deve ao fato de que a imagem da face está muito próxima da câmera e a região da face capturada não está completa. Para a distância do objeto acima de 50 cm da câmera, a taxa de desempenho do sistema de reconhecimento caiu de cerca de 56% para 12%.

A maior taxa de reconhecimento do sistema é obtida com a distância da câmera e do objeto entre 40 cm a 60 cm. Nesta faixa, a taxa de reconhecimento é razoável, mas o ângulo de rotação da cabeça do sujeito deve estar dentro da faixa de -20 a +20 graus apenas. Quando a variação do ângulo de rotação é superior a -30 ou +30 graus, o sistema falhou.

- **8 - A Low-Cost Embedded Facial Recognition System for Door Access Control using Deep Learning**

No trabalho de (ORNA; BENÍTEZ; PÉREZ, 2020) é proposto um modelo de baixo custo para um sistema de fechadura inteligente que utiliza o reconhecimento facial baseado em técnicas de aprendizagem profunda (*Deep Learning*) para conceder acesso às portas.

O sistema usa uma plataforma baseada em ARM (Raspberry Pi) para realizar parte do processamento do sistema e um hardware dedicado de aceleração de inferência de rede neural (um dispositivo complementar) conhecido como processador *Intel Neural Compute Stick-2* (NCS2) para executar as tarefas de classificação baseadas no aprendizado profundo. O NCS2 é compatível com o kit de ferramentas *Intel Distribution of OpenVINO*, que é um kit de ferramentas gratuito que permite o desenvolvimento de aplicações de visão

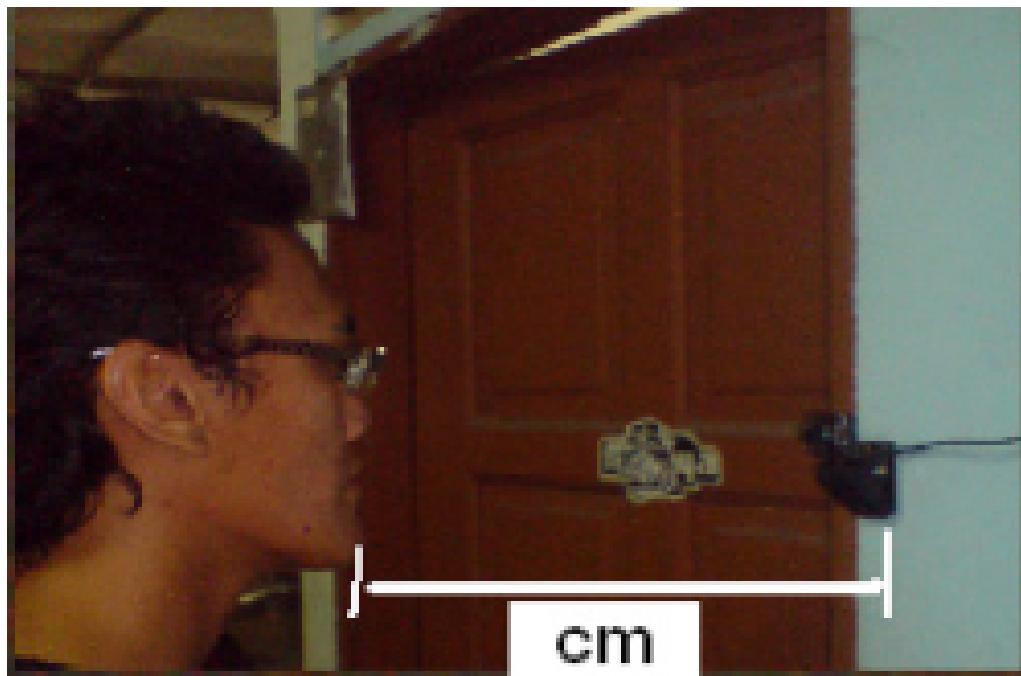


Figura 414 – Câmera do sistema capturando a imagem do usuário. Fonte: extraído de (IBRAHIM; ZIN, 2011)

artificial baseadas em *Convolutional Neural Network* (CNN) e que permite um aprendizado profundo. A Figura 415 mostra a detecção sendo realizada pelo sistema.

Foi utilizada uma rede neural artificial profunda (MobileNet) para detectar objetos aleatórios. Depois, um detector de rostos que compara imagens de vídeo em tempo real com rostos codificados armazenados em um banco de dados. O detector usou o algoritmo de cascata Haar para identificar as faces.

Para testar a eficácia do sistema implementado, foram realizadas experiências um pequeno conjunto de dados de galeria construído com imagens da face frontal e lateral de 15 indivíduos que utilizaram o sistema. O teste de identificação do sistema foi realizado cinco vezes por pessoa para determinar quantas vezes o sistema foi capaz de reconhecer a pessoa na frente efetivamente e quantas vezes ela foi confundida com outra pessoa do banco de dados.

Dos resultados uma precisão média de 88,75% foi alcançada pelo sistema. Em alguns casos, a pessoa foi corretamente identificada nas cinco tentativas, enquanto no pior dos casos, o sujeito foi identificado apenas em três das cinco tentativas.

• 09 - A Privacy Enhanced Facial Recognition Access Control System Using Biometric Encryption

No trabalho de (COLE; EL-KHATIB, 2017) é apresentada uma técnica prática de criptografia biométrica integrada com reconhecimento facial. A metodologia utiliza geração de chaves biométricas e autenticação de faces para garantir a privacidade do usuário.

O esquema de criptografia consiste de duas fases: a fase de inscrição e a fase de autenticação. Na fase de inscrição são seguidos sete estágios para criptografar modelos biométricos a partir das imagens de inscrição.

No estágio de pré-processamento, as imagens capturadas são convertidas em escala de cinza e redimensionadas para um tamanho padrão. O próximo estágio é o de quantização, onde as imagens são quantizadas para reduzir pequenas variações e reduzir a imagem em uma escala adequada para os cálculos de criptografia biométrica.

No estágio de geração de chaves biométricas aleatórias uma chave biométrica aleatória é gerada usando um gerador de números aleatórios que produz um vetor biométrico do mesmo tamanho da imagem de entrada vetorial. Na fase de extração de característica o PCA é aplicada nas imagens quantificadas.

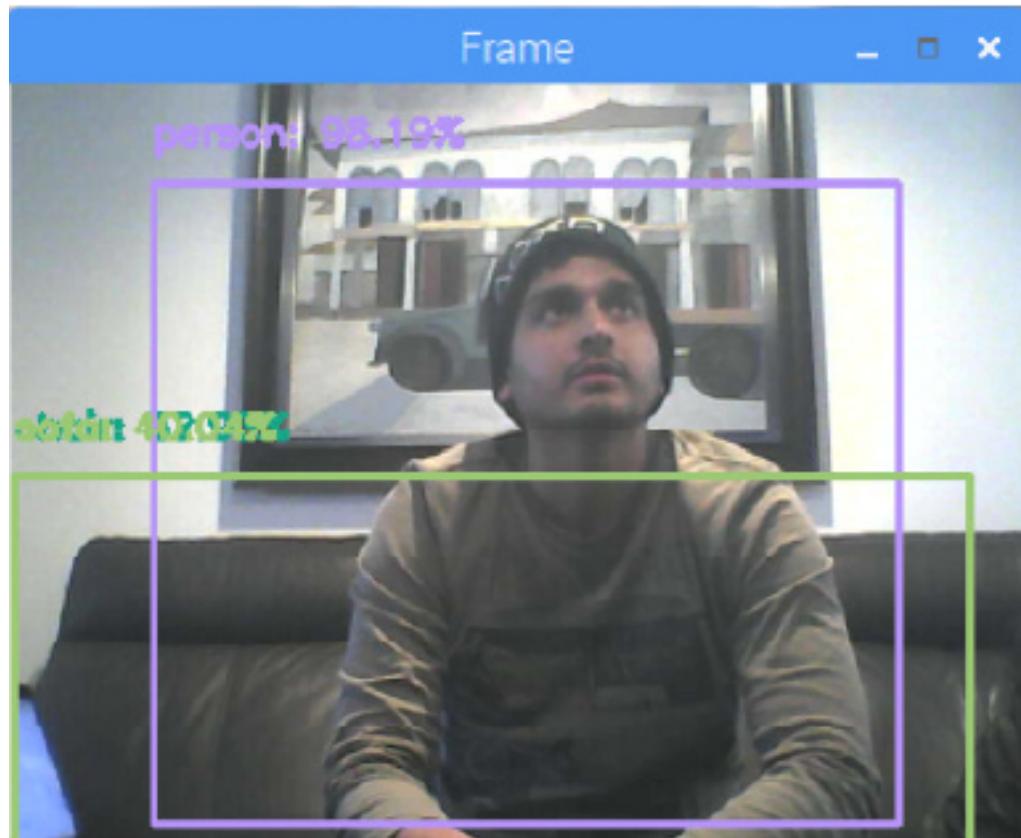


Figura 415 – Face sendo detectada pelo NCS2. Fonte: extraído de (ORNA; BENÍTEZ; PÉREZ, 2020)

No estágio de extração média e de peso o resultado do processamento do PCA produz as médias das imagens de entrada registradas, estes valores são fundidos com a chave biométrica para ocultar seus valores originais antes do armazenamento, pois os próprios valores podem vazar informações que poderiam ser usadas para deduzir a biometria original.

Já no estágio de encadernação de chaves, a chave gerada aleatoriamente é fundida com a imagem de entrada quantificada para produzir o modelo biométrico; este modelo é armazenado como dados auxiliares e é usado para recuperar a chave biométrica desde que um novo modelo biométrico estiver suficientemente próximo do que foi inicialmente registrado.

Na fase de armazenamento, os dados que serão usados para recuperar a chave biométrica são armazenados, uma vez que uma nova imagem de entrada seja fornecida e tenha muito pouca variação em comparação com o que foi registrado. Isto garante que a chave só será desbloqueada se for fornecida a nova biometria do usuário inscrito, pois um impostor não deve ser capaz de reproduzir uma característica biométrica muito parecida.

Na fase de autenticação são repetidos alguns dos estágios da primeira fase, mas é acrescentado um módulo de recuperação da chave biométrica e um outro módulo para o cálculo de distância euclidiana.

Para a tarefa de reconhecimento facial foram utilizados smartphones para pré-processamento dos modelos faciais. Após o pré-processamento o modelo facial biométrico é enviado para um serviço de nuvem da *Biometrics Trusted Authority* (BTA). O algoritmo utilizado foi o PCA.

Segundo (COLE; EL-KHATIB, 2017) os resultados obtidos do protótipo desenvolvido sugeriram que o esquema proposto pode ser uma abordagem adequada para melhorar a exigência de privacidade dos esquemas biométricos tradicionais e seus modelos biométricos armazenados.

- 10 - *Development of a Secure Access Control System Based on Two-Factor Authentication Using Face Recognition and OTP SMS-Token*

No trabalho de (PRAMANA; LESTYEA; AMIRUDDIN, 2020) foi projetado um sistema de controle de acesso seguro baseado na autenticação de dois fatores usando reconhecimento facial e *One Time Password* (OTP) por SMS para evitar ataques de falsificação. Para superar os ataques de falsificação mais comuns é aplicado o código OTP como uma segunda autenticação. OTP é uma senha de uso único que não pode ser usada uma segunda vez. Ao ser registrado um número móvel no sistema, sempre que o usuário fizer acesso, o sistema enviará um código de acesso único via SMS que garante que um usuário legítimo tenha solicitado acesso ao serviço.

No trabalho não é informado qual das metodologias foi adotada no sistema proposto, apenas é dito que o primeiro passo do sistema proposto é escanear ou identificar o rosto do usuário usando um módulo de câmera chamado ESP32-CAM. Se o processo de escaneamento for bem sucedido, significa que o sistema reconhece o rosto do usuário, então o administrador enviará um código para o usuário.

Infelizmente não foram dadas maiores informações a respeito dos métodos de detecção e reconhecimento e nem sobre a acurácia dos testes realizados.

- **11 - Door Access Control based on Illumination Invariant Face Recognition in Embedded System**

No trabalho de (DEWANTARA; BACHTIAR; LANTANG, 2020) é proposto um sistema de fechadura baseado na biometria e no reconhecimento automático da face. É feito um reconhecimento facial com a capacidade de reconhecer pessoas mesmo em condições de iluminação difíceis. A entrada do sistema consiste em um rosto humano que é capturado por meio de uma webcam. A webcam é conectada a um mini PC embutido, chamado Lattepanda que roda um sistema de Windows 10.

Após isso é feita a captura da imagem, porém não é informado exatamente qual algoritmo realiza a detecção da imagem. Após esta etapa, é feito o reconhecimento utilizando o método invariante de iluminação desenvolvido para o trabalho proposto. O método ajusta o contraste do rosto detectado em condições difíceis de iluminação. A Figura 416 mostra as condições de iluminação testadas no sistema proposto.

É utilizado um algoritmo genético no processamento. A tarefa de reconhecimento é realizada pelo algoritmo Coeficiente de Correlação do Momento do Produto Pearson (PPMCC - *Pearson Product Moment Correlation Coefficient*). Para os testes foram convidadas 20 pessoas para armazenar suas imagens faciais no banco de dados. Os participantes consistiam de 17 homens e três mulheres. Para avaliar o desempenho do sistema foram incluídas apenas 10 pessoas. Os testes foram feitos sob várias condições de iluminação, alcançando uma acurácia de 91% quando a iluminação é invariável. Porém quando em baixa iluminação, o sistema teve um desempenho inferior alcançando 66,7% em condições de baixa iluminação, 44,4% em condições de iluminação direta e 55,5% em nenhuma condição de iluminação.

- **12 - Face recognition system for access control using principal component analysis**

No trabalho de (BAKSHI; PRABHU, 2017) foi desenvolvida uma aplicação de um sistema de trava de segurança através de um método de reconhecimento facial. A aplicação foi desenvolvida no Matlab e utiliza uma placa Arduíno UNO para transmissão de dados. A entrada de vídeo é feita por um módulo de detecção de movimento através de uma câmera integrada. Após ser feita a detecção de movimento, a etapa seguinte é a detecção facial. O método Eigenfaces que é baseado no PCA é utilizado para treinamento de imagens no banco de dados. O PCA é utilizado para extrair as características da face mas não realiza a classificação, a classificação dos dados é feito utilizando a distância Euclidiana. A placa de Arduíno Uno é utilizada para controlar a trava de segurança dependendo do resultado do processamento.

Não foi informado quantas imagens foram utilizadas nos testes, apenas que o módulo de detecção atingiu uma alta taxa de reconhecimento correto. O sistema fornece resultados precisos quando é operado sob parâmetros controlados e não descontrolados. O sistema apresentou problemas de comunicação entre a placa Arduíno e o Matlab e também o sistema sofreu com relação ao baixo desempenho da placa Arduíno Uno, sendo sugerido pelos autores utilizar placas mais potentes como "Arduino Duemilanove" e "Arduino Mega", além de aumentar as imagens do banco de dados.

- **13 - Implementation of Two Factor Authentication based on RFID and Face Recognition using LBP Algorithm on Access Control System**

Condições	Aparências originais	Iluminação com base em aparências invariantes
Sem iluminação (O rosto parece fraco)		
Baixa iluminação		
Normal		
Retroiluminação		
Iluminação direta		

Figura 416 – Condições de iluminação testadas. Fonte: extraído de ([DEWANTARA; BACHTIAR; LANTANG, 2020](#))

No trabalho de ([WAHYUDONO; OGI, 2020](#)) foi implementado um sistema de controle de acesso utilizando dois fatores de autenticação, RFID e reconhecimento facial. O sistema utiliza o algoritmo LBP para a criação de um método de segurança contra ataques *spoofing*.

Na autenticação por dois fatores, primeiramente o usuário ativa o módulo de entrada de dados do sistema, após isso o usuário autentica a etiqueta RFID no leitor. O módulo USB-RFID EM4100 é utilizado para realizar a função de autenticação RFID, caso a etiqueta seja identificada, o sistema envia um log com a mensagem de autenticado. Após isso é ativado o módulo de reconhecimento facial. Para a captura das imagens é utilizada uma câmera do processador Raspberry Pi, e o reconhecimento facial é feito utilizando o algoritmo LBP para superar a vulnerabilidade de ataques de falsificação facial. As Figuras 417 e 418 mostram a interface de interação com o usuário e a autenticação sendo realizada, respectivamente.

- **14 - Personal access control system using moving object detection and face recognition**

No trabalho de ([ZELJKOVIĆ et al., 2014](#)) é proposto um sistema de controle de acesso pessoal em tempo real baseado na detecção de objetos em movimento e reconhecimento facial. Ele consiste em três etapas



Figura 417 – Notificação de Autenticação RFID bem sucedida . Fonte: extraído de ([WAHYUDONO; OGI, 2020](#))



Figura 418 – Exibição de autenticação de face bem sucedida. Fonte: extraído de ([WAHYUDONO; OGI, 2020](#))

principais: a detecção de objetos em movimento, localização e extração de rosto e reconhecimento de rosto em seqüências reais de imagem.

Inicialmente o vídeo de entrada é capturado por uma câmera colocada no teto da área a ser vigiada de frente para a entrada. A saída da câmera é conectada à CPU que contém o sistema proposto e processa o vídeo gravado.

Os autores propuseram um algoritmo que realiza a detecção de objetos em movimento que é resistente a pequenas mudanças de iluminação. A extração de características é feita por um sistema de detecção facial que utiliza um algoritmo chamado *The Successive Mean Quantization Transform* (SMQT) e uma rede dispersa do classificador Winnows. A detecção facial é feita utilizando o PCA. As Figuras 419 e 420 mostram a detecção de objetos em movimento e a extração das faces, respectivamente.

O sistema foi testado através de quatro sequências de vídeos gravados tanto em ambiente interno quanto externo com ligeiras mudanças de iluminação. O sistema proposto teve 100% de aproveitamento na fase de

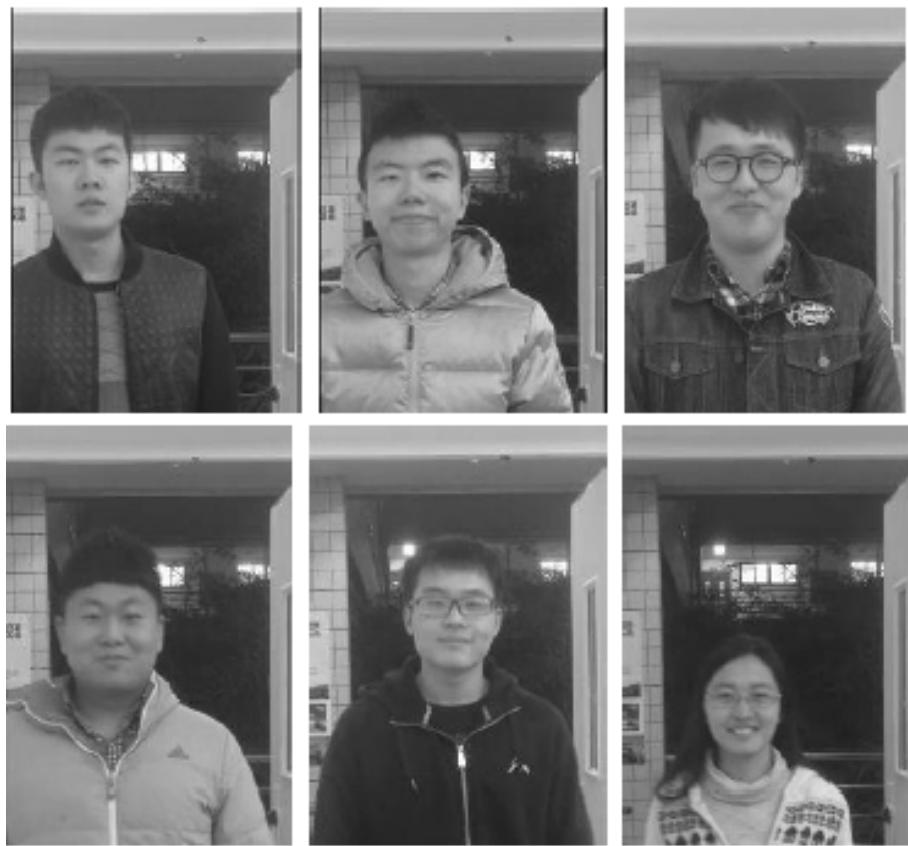


Figura 419 – Detecção de objetos em movimento. Fonte: extraído de ([ZELJKOVIĆ et al., 2014](#))

deteção de objetos em movimento, 96,67% na fase de classificação de extração da face e 100% na fase de reconhecimento.

- ***15 - The implementation of the visitor access control system for the senior citizen based on the LBP face recognition***

No trabalho de ([ZELJKOVIĆ et al., 2014](#)) foi implementado um sistema de controle de acesso voltado para o público da terceira idade. Foi utilizado o algoritmo LBP para a extração de característica das imagens integrado com uma plataforma Raspberry Pi. A placa Raspberry é utilizada para pré-processamento e análise de imagens. Para captura das imagens foi conectada uma *webcam USB Logitech C310*, instalado um componente de comunicação sem fio para transferir dados para um smartphone, feita uma implementação de uma fechadura elétrica pelo Arduíno Uno e conectado um motor MG995 para controlar a admissão dos usuários.

A câmera utilizada fica em modo de espera aguardando o usuário para detectar o rosto. Quando o usuário chega, ela captura uma foto e armazena o rosto do visitante. A classificação da face é feita por uma biblioteca do OpenCV que realiza classificação em cascata. A imagem da face capturada em escala RGB é transformada em uma imagem em escala de cinza. A imagem em escala de cinza é alimentada no algoritmo LBP para extrair os vetores de característica. Então, o algoritmo KNN (*k-Nearest Neighbors*) é usado para determinar qual rosto do conjunto de dados é o mais próximo do rosto testado. Os resultados da identificação são enviados então para o celular do usuário. A Figura 421 mostra as imagens após o processo de transformação em escala cinza.

Inicialmente é feito um treinamento com as imagens no banco de dados, são utilizadas 12 imagens com diferentes ângulos de captura de cada indivíduo, totalizando 120 imagens. Os resultados experimentais mostram que o sistema implementado pode atingir 90% da taxa de reconhecimento facial. Foi relatado que



Figura 420 – Extração da face das imagens. Fonte: extraído de ([ZELJKOVIć et al., 2014](#))

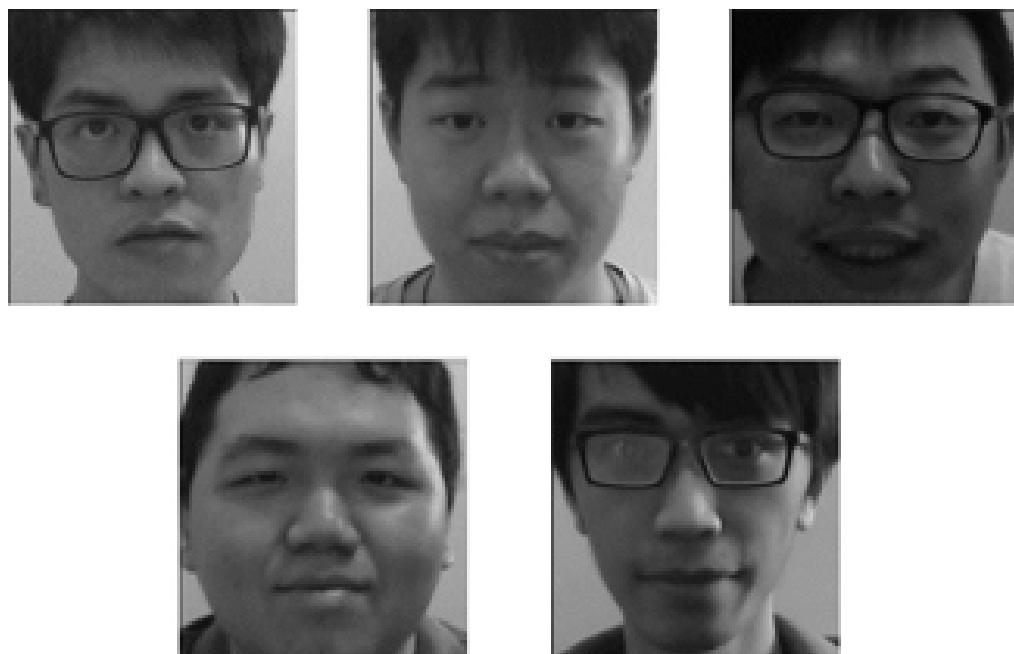


Figura 421 – Fotos após o processo de transformação da imagem em escala cinza. Fonte: extraído de ([LIN; HUANG, 2017](#))

o sistema teve uma perda de desempenho na taxa de reconhecimento quando este foi feito em imagens escuras e com menos iluminação.

- 16 - *Web-based online embedded door access control and home security system based on face recognition*

No trabalho de (SAHANI et al., 2015) é descrita a implementação de um sistema de segurança doméstico baseado na tecnologia de reconhecimento do rosto humano e na tecnologia de monitoramento remoto, para confirmar a identidade do visitante e controlar a acessibilidade das portas. Foi implementado um módulo de controle sem fio via Web para que o sistema possa ser controlado remotamente. O sistema é de baixo custo e a implementação foi pensada para um ambiente doméstico de maneira que possa ser operado utilizando um *smartphone*.

No sistema proposto são utilizados dois componentes para comunicação, uma unidade de controle sem fio (WCU - Wireless Control Units) e uma unidade de informação sem fio (WIU - Wireless Information Unit) conectada por um transceptor de rádio que permite a transferência de informações de controle que utiliza a tecnologia de rede ZigBee. Também foi utilizada uma placa Raspberry Pi como a unidade de processamento da WIU e um servidor Web para enviar e receber informações do sistema embarcado.

Inicialmente, a unidade *Raspberry Pi* e a câmera são instaladas em uma casa através do ZigBee. Se algum visitante chegar, o *Raspberry Pi* envia um SMS e/ou e-mail incluindo os detalhes para um servidor baseado na Internet utilizando o *e-mail-id* do usuário. O servidor então envia o nome e a foto dos visitantes ao proprietário da casa. O proprietário pode logar no servidor e interagir com o sistema embarcado em tempo real através da página web sem a necessidade de manter um servidor adicional.

A técnica aplicada ao reconhecimento é o PCA que foi escolhida pela simplicidade e redução de dimensionalidade. A Figura 422 mostra o resultado do reconhecimento facial após o processamento.

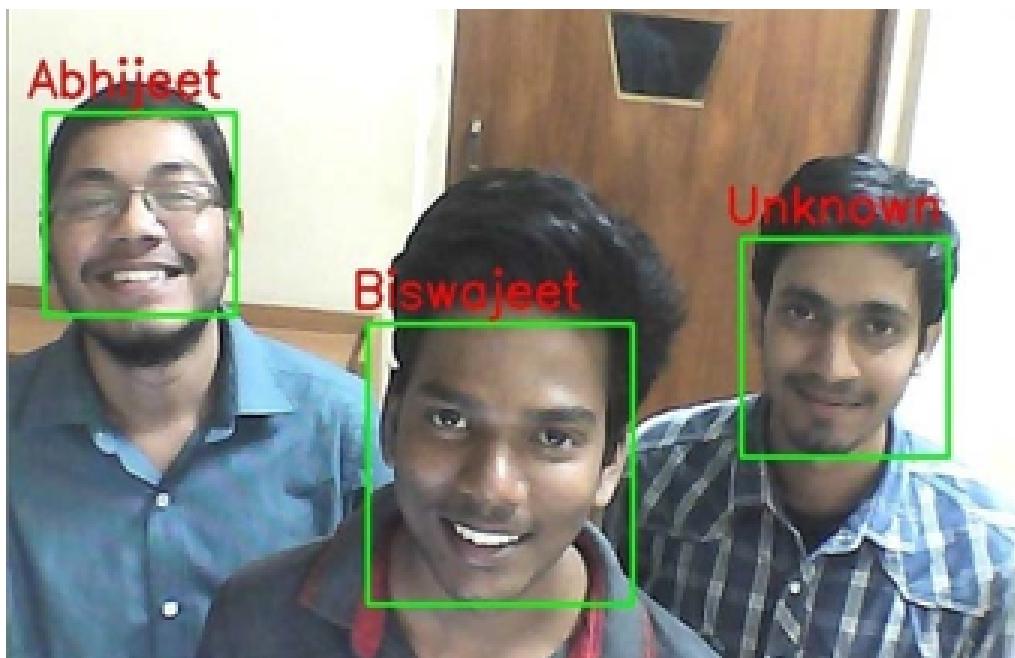


Figura 422 – Resultado do Reconhecimento Facial. Fonte: extraído de (SAHANI et al., 2015)

O SQLite foi utilizado como banco de dados embutido para armazenamento local do sistema. Não foi informado a respeito da porcentagem de acerto no processo de reconhecimento através dos testes realizados. Mas a conclusão é a de que o sistema apresentou bom desempenho quanto à estabilidade de um sistema embarcado utilizando a tecnologia de internet das coisas com monitoramento remoto. Com ele é possível realizar um gerenciamento eficaz, possibilitando ao usuário controlar a fechadura de portas através da web.

- **17 - Automated access control system using face recognition**

No trabalho de (RAMESHWARI et al., 2021) foi desenvolvido um sistema de segurança utilizando reconhecimento facial para um campus universitário. Inicialmente para detectar o rosto da pessoa, a imagem capturada pela câmera é alinhada corretamente para identificar o rosto. É utilizado um método de codificação

facial para diferenciar e encontrar as distâncias entre os olhos, cor, bordas e fundo. Para extração destas características é usado o modelo *Histogram Object Gradient* (HOG) que é um descritor de características utilizado na localização de objetos. A Figura 423 apresenta uma amostra da face após a extração de características realizada pelo HOG.

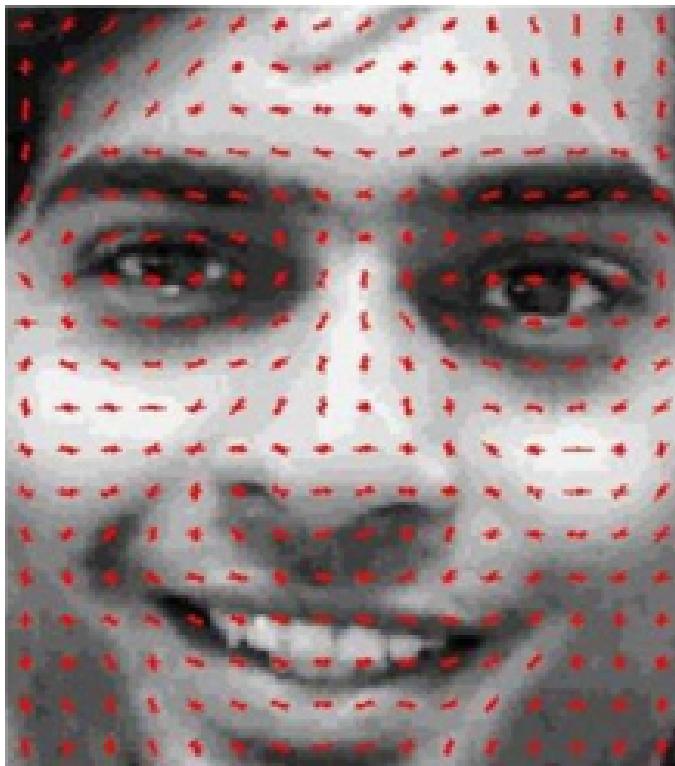


Figura 423 – Amostra de face pelo descritor HOG. Fonte: extraído de ([RAMESWARI et al., 2021](#))

Para a detecção facial é aplicado o algoritmo *FaceNet* que se utiliza de redes convolucionais para alcançar uma maior precisão na localização dos rostos. A tarefa de classificação é feita pela máquina de vetores de suporte (SVM) que é um algoritmo de aprendizagem de máquina supervisionado. A Figura 424 mostra um conjunto de imagens utilizadas no treinamento da SVM.



Figura 424 – Faces para o treinamento da SVM. Fonte: extraído de ([RAMESWARI et al., 2021](#))

Para os testes o algoritmo *FaceNet* foi comparado com os algoritmos *Eigenfaces*, *Fisherface* e Histograma do Padrão Binário Local (*The Local Binary Pattern Histogram*) através da implementação do mesmo conjunto de dados contendo a mesma quantidade de imagens e a mesma quantidade de treinamentos para cada algoritmo. O *FaceNet* obteve o melhor resultado com uma precisão de 97% com um banco de dados de 40 imagens de treinamento enquanto os outros algoritmos ficaram abaixo de 50% de precisão.

- 18 - *Research on embedded access control security system and face recognition system*

No trabalho de (AI; CHENG, 2018) foi descrita uma arquitetura de um sistema completo de controle de acesso. O sistema conta com módulos de detecção de vibração, detector de infravermelho, detector de fumaça, detector de gás combustível. Além disso, ele estabelece um servidor Web que conta com uma página de registro de usuário, interface de login de usuário e uma página de monitoramento de segurança a fim de realizar o acesso remoto ao servidor de segurança além do reconhecimento facial.

Para o reconhecimento facial, inicialmente as imagens capturadas pela câmera são convertidas para o formato RGB. Após a conversão é iniciado o processo de detecção de imagem que é feito pelo algoritmo AdaBoost. No processo seguinte é feita a identificação facial pelo algoritmo PCA. A Figura 425 mostra a identificação facial sendo realizada.



Figura 425 – Identificação facial. Fonte: extraído de (AI; CHENG, 2018)

Para os testes foi utilizada uma biblioteca de identificação facial da ORL que contém imagens de 40 pessoas com 10 imagens de cada uma delas totalizando 400 imagens. As imagens foram tiradas com diferentes poses e expressões faciais.

Para o experimento foram selecionadas aleatoriamente 5 imagens de cada pessoa, totalizando 200 imagens, as outras duzentas são usadas como amostras de treinamento do PCA. Os resultados mostraram que a taxa de acerto do algoritmo se altera dependendo da quantidade de informações processadas. No experimento com o melhor resultado, o algoritmo conseguiu uma acurácia de 88% sob um determinado número de informações de entrada e amostras de treinamento de 5 fotos.

- 19 - *Electronic Door Access Control using MyAccess Two-Factor Authentication Scheme featuring Near-Field Communication and Eigenface-based Face Recognition using Principal Component Analysis*

No trabalho de (YIN et al., 2016) é implementado um sistema de controle de acesso utilizando autenticação em dois fatores através da tecnologia de Comunicação Próxima ao Campo (*Near Field Communication* (NFC) e reconhecimento facial.

O NFC é uma tecnologia de comunicação sem fio que funciona através da aproximação de dois dispositivos. O NFC se comunica com um leitor de cartões com a tecnologia RFID. No sistema proposto é utilizada uma etiqueta NFC digitalizada como primeiro fator de validação.

Posteriormente, para a segunda autenticação, uma câmera detectará e reconhecerá o rosto do usuário comparando a entrada com a impressão facial do banco de dados. A impressão facial capturada é utilizada como um segundo fator de validação. Se a imagem de entrada for conhecida, o sistema prosseguirá para o próximo processo comparando as duas entradas, a etiqueta NFC e a imagem para verificar se a combinação é válida e o usuário deve ser autorizado ou não.

No sistema o usuário é solicitado a inicialmente inserir imagens com diferentes poses e expressões, que são o conjunto de imagens utilizados no reconhecimento facial. A Figura 426 mostra o conjunto de imagens que serão utilizadas no treinamento de reconhecimento facial.



Figura 426 – Conjunto de imagens de treinamento (YIN et al., 2016)

O algoritmo utilizado no reconhecimento é o PCA. Foram utilizados três tipos diferentes de iluminação para testar a precisão do sistema na detecção e reconhecimento facial. O banco de dados é constituído de imagens de 50 pessoas. Nos testes o sistema foi capaz de reconhecer o usuário dependendo da configuração da intensidade da iluminação, pois isso tem um grande efeito nos resultados dos testes. Na conclusão os autores reconheceram que o sistema pode ser melhorado ainda mais através da automação do sistema para que haja menos necessidade de interferência do usuário e através da utilização de uma câmera de maior precisão.

Na seção seguinte são feitas algumas discussões e análises a respeito dos trabalhos coletados.

36.4.2 Discussões

Através da leitura dos trabalhos coletados, foi observado que nem todos informaram completamente quais algoritmos foram utilizados em todas as etapas. Alguns trabalhos informaram apenas qual técnica foi utilizada na etapa de extração, outros informaram qual técnica aplicaram na classificação e etc. Alguns trabalhos desenvolveram metodologias próprias para o reconhecimento, sem o uso de técnicas clássicas.

36.4.3 Detecção de faces

Após o procedimento de análise dos trabalhos e a extração os dados referentes às técnicas de reconhecimento facial mais utilizadas em aplicações de controle de acesso, foi observado que na fase de detecção o algoritmo mais utilizado foi o Viola-Jones, utilizado em (SHAVETOV; SIVTSOV, 2020), (TALEB; OUIS; MAMMAR, 2014) e (AI; CHENG, 2018).

Os trabalhos de (PAN, 2012), (DEWANTARA; BACHTIAR; LANTANG, 2020) e (ZELJKOVIĆ et al., 2014) foram os que utilizaram metodologias próprias para a fase de detecção. O classificador para detecção

de objetos *Haar Cascade* foi utilizado nos trabalhos de (NAG; NIKHILENDRA; KALMATH, 2018) e (ORNA; BENÍTEZ; PÉREZ, 2020) para a detecção da região da face. Outras técnicas que foram menos utilizadas foram MobileNet em (ORNA; BENÍTEZ; PÉREZ, 2020), FaceNet em (RAMESWARI et al., 2021) e Robust Real-Time Face Detection no trabalho de (CHANG; LIAO, 2015).

36.4.4 Extração de Características

O PCA foi a técnica de extração de características mais utilizada para uma ou mais etapas de reconhecimento facial nos trabalhos, sendo utilizada em (TALEB; OUIS; MAMMAR, 2014), (IBRAHIM; ZIN, 2011), (COLE; EL-KHATIB, 2017), (BAKSHI; PRABHU, 2017), (ZELJKOVIĆ et al., 2014), (SAHANI et al., 2015), (AI; CHENG, 2018) e (YIN et al., 2016). Essa técnica foi utilizada para a redução da dimensionalidade das imagens, em seguida, o vetor reduzido resultante era utilizado nas fases seguintes. O uso do PCA foi motivada devido a sua simplicidade e pela capacidade de redução de dimensionalidade nas imagens. A técnica LBP também foi utilizada para extração de características nos trabalhos de (CHANG; LIAO, 2015), (WAHYUDONO; OGI, 2020) e (LIN; HUANG, 2017). Outras técnicas menos utilizadas foram SMQT utilizada em (ZELJKOVIĆ et al., 2014) e HOG utilizada em (RAMESWARI et al., 2021).

36.4.5 Classificação

O classificador mais utilizado nos trabalhos foi o SVM, utilizado em (MANTECÓN et al., 2015) e (RAMESWARI et al., 2021). Outros métodos baseados em redes neurais foram amplamente utilizados. Os trabalhos de (PAN, 2012), (IBRAHIM; ZIN, 2011), (ORNA; BENÍTEZ; PÉREZ, 2020) e (SHAVETOV; SIVTSOV, 2020) utilizaram diversas metodologias baseadas em redes neurais. Outros classificadores utilizados incluíram *Wavelet Structural Similarly* usado em (CHANG; LIAO, 2015) e Coeficiente de Correlação do Momento do Produto Pearson (PPMCC - Pearson Product Moment Correlation Coefficient) usado em (DEWANTARA; BACHTIAR; LANTANG, 2020).

36.4.6 Módulo Anti Spoofing

Um módulo de detecção de ataques que detecta tentativas de fraudes, como por exemplo, utilizar uma foto de alguém na frente da câmera, ou incluir arquivos maliciosos nas imagens, foi pouco utilizado nos trabalhos, sendo utilizado apenas em (SHAVETOV; SIVTSOV, 2020) e (WAHYUDONO; OGI, 2020).

Alguns trabalhos também utilizaram a autenticação em dois fatores para aumentar a segurança. Nos artigos de (CHANG; LIAO, 2015), (PAN, 2012), (COLE; EL-KHATIB, 2017), (PRAMANA; LESTYEÀ; AMIRUD-DIN, 2020), (DEWANTARA; BACHTIAR; LANTANG, 2020), (WAHYUDONO; OGI, 2020) e (YIN et al., 2016) foi implementada algum fator extra de autenticação. O RFID foi o mais utilizado, tendo sua adesão nos trabalhos de (CHANG; LIAO, 2015), (PAN, 2012) e (WAHYUDONO; OGI, 2020),

36.4.7 Processamento de Dados

Para processamento dos algoritmos a placa *Raspberry Pi* foi a mais utilizada tendo adesão em vários trabalhos, incluindo (NAG; NIKHILENDRA; KALMATH, 2018), (ORNA; BENÍTEZ; PÉREZ, 2020), (WAHYUDONO; OGI, 2020), (LIN; HUANG, 2017) e (SAHANI et al., 2015). O fato se deve que a grande maioria dos trabalhos foram feitos para serem utilizados em casas para controlar portas ou portões, o que inclui a necessidade de utilizar algoritmos mais simples e menos robustos que possam ser executados em processadores de baixo custo, como é o caso da *Raspberry*.

36.4.8 Problemas Encontrados

Pela leitura dos trabalhos, foi observado que os principais problemas enfrentados pelos algoritmos eram relacionados com a iluminação do ambiente e das imagens. Dependendo da iluminação o algoritmo tinha um desempenho melhor ou pior. Outro fator tem relação com a angulação das imagens e expressões, alguns algoritmos não conseguiram performar bem quando a imagem tinha uma determinada angulação de rosto.

O mesmo vale para a distância, a variação desse parâmetro também afetava o desempenho obtido pelos métodos.

36.5 Conclusão

Neste trabalho foi realizado um mapeamento sistemático das técnicas de reconhecimento de faces aplicadas à área de segurança. Foi feita uma análise de cada artigo para selecionar as técnicas utilizadas nas etapas de reconhecimento facial para entender quais têm sido aplicadas e quais as tendências para o futuro envolvendo a área de segurança e o reconhecimento facial.

Foi identificado que o setor de segurança envolvendo técnicas de reconhecimento facial tem um campo bem amplo para desenvolvimento e que cada vez mais será necessário, devido que ao passar do tempo aumenta-se mais ainda a importância de se proteger dados e informações com técnicas cada vez mais eficazes. Além disso, também existem outros fatores que contribuem para isso, como fatores mundiais relacionados ao controle de saúde e segurança, como por exemplo, a pandemia causada pelo Coronavírus, onde o contato com superfícies biométricas se tornou inviável, fazendo com que seja muito arriscado utilizar sistemas de biometria digitais.

Diversas técnicas têm sido implementadas no desenvolvimento do reconhecimento facial como PCA, LBP, SVM e técnicas baseadas em redes neurais que foram amplamente utilizadas em aplicações de segurança que envolvem o reconhecimento facial. A tendência é que cada vez mais sejam integrados algoritmos utilizados para detecção, extração e classificação com técnicas de inteligência artificial e internet das coisas.

Para o futuro, o reconhecimento de faces envolve desafios relacionados à melhorar a qualidade de detecção em condições de baixa iluminação, com diferentes expressões de rosto, detecção em várias distâncias da imagem para a câmera, detecção de objetos em movimento e detectar imagens e vídeos em diferentes ângulos.

O desenvolvimento de módulos anti-*spoofing* representa outro desafio, uma vez que tentativas de ataques e fraldes fazem parte da área de segurança de acesso. Poucos trabalhos analisados fizeram implementações neste sentido envolvendo detecção de vivacidade e detecção de *malwares* em imagens, abrindo espaço para o desenvolvimento de mais pesquisas no assunto.

Como trabalhos futuros, pretende-se realizar o mapeamento sistemático das técnicas de reconhecimento facial aplicadas na área de investigação de crimes que envolvem detectar suspeitos através de vídeos e imagens sob diversas condições. Este tipo de trabalho também representa um desafio para o futuro conforme as técnicas de reconhecimento facial avancem.

Referências

- ABBAS, A.; ABDELSAMEA, M. M.; GABER, M. M. Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network. *arXiv preprint arXiv:2003.13815*, 2020. Citado 3 vezes nas páginas 191, 192 e 201.
- ABBAS, N. et al. Mobile edge computing: A survey. *IEEE Internet of Things Journal*, IEEE, v. 5, n. 1, p. 450–465, 2017. Citado na página 493.
- ABDEL-ZAHER, A. M.; ELDEIB, A. M. Breast cancer classification using deep belief networks. *Expert Systems with Applications*, Elsevier, v. 46, p. 139–144, 2016. Citado na página 162.
- ABDOLI, S.; CARDINAL, P.; KOERICH, A. L. End-to-end environmental sound classification using a 1d convolutional neural network. *Expert Systems with Applications*, Elsevier, v. 136, p. 252–263, 2019. Citado 4 vezes nas páginas 590, 595, 597 e 603.
- ABDULLA, W. *Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow*. [S.l.]: Github, 2017. https://github.com/matterport/Mask_RCNN. Citado na página 295.
- ACHARYA, U. R. et al. Automated diagnosis of glaucoma using texture and higher order spectra features. *IEEE Transactions on information technology in biomedicine*, IEEE, v. 15, n. 3, p. 449–455, 2011. Citado na página 59.
- ADEGUN, A. A.; VIRIRI, S. Deep learning-based system for automatic melanoma detection. *IEEE Access*, IEEE, v. 8, p. 7160–7172, 2019. Citado 2 vezes nas páginas 224 e 230.
- ADHIANTO, L. et al. Hptoolkit: Tools for performance analysis of optimized parallel programs. *Concurrency and Computation: Practice and Experience*, Wiley Online Library, v. 22, n. 6, p. 685–701, 2010. Citado na página 341.
- AFSHAR, P. et al. Covid-caps: A capsule network-based framework for identification of covid-19 cases from x-ray images. *arXiv preprint arXiv:2004.02696*, 2020. Citado na página 172.
- AFSHAR, P.; MOHAMMADI, A.; PLATANIOTIS, K. N. Brain tumor type classification via capsule networks. In: IEEE. *2018 25th IEEE International Conference on Image Processing (ICIP)*. [S.l.], 2018. p. 3129–3133. Citado na página 162.
- AFSHAR, P. et al. 3d-mcn: A 3d multi-scale capsule network for lung nodule malignancy prediction. *Scientific Reports*, Nature Publishing Group, v. 10, n. 1, p. 1–11, 2020. Citado 2 vezes nas páginas 32 e 35.
- AFSHAR, P.; PLATANIOTIS, K. N.; MOHAMMADI, A. Capsule networks for brain tumor classification based on mri images and coarse tumor boundaries. In: IEEE. *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2019. p. 1368–1372. Citado na página 172.
- AGAFONOVA, A.; MYASNIKOVA, V. Efficiency comparison of the routing algorithms used in centralized traffic management systems. *ScienceDirect*, 2017. Citado na página 554.
- AHA, D.; KIBLER, D. Instance-based learning algorithms. *Machine Learning*, v. 6, p. 37–66, 1991. Citado na página 73.

- AHMED, A.; AHMED, E. A survey on mobile edge computing. In: . [S.l.: s.n.], 2016. Citado na página 310.
- AHMED, E.; REHMANIB, M. H. Mobile edge computing: Opportunities, solutions, and challenges. *ScienceDirect*, ScienceDirect, v. 70, p. 59–63, 2017. Citado na página 401.
- AHN, J. M. et al. A deep learning model for the detection of both advanced and early glaucoma using fundus photography. *PloS one*, Public Library of Science, v. 13, n. 11, 2018. Citado 2 vezes nas páginas 9 e 10.
- AI, H.; CHENG, X. Research on embedded access control security system and face recognition system. *Measurement*, v. 123, p. 309–322, 2018. ISSN 0263-2241. Disponível em: <https://www.sciencedirect.com/science/article/pii/S026322411830280X>. Citado 4 vezes nas páginas 691, 708, 709 e 710.
- AL-BANDER, B. et al. Automated glaucoma diagnosis using deep learning approach. In: IEEE. 2017 14th International Multi-Conference on Systems, Signals & Devices (SSD). [S.l.], 2017. p. 207–210. Citado 2 vezes nas páginas 9 e 10.
- AL-NASHERI, A. et al. Voice pathology detection and classification using auto-correlation and entropy features in different frequency regions. *IEEE Access*, IEEE, v. 6, p. 6961–6974, 2017. Citado 2 vezes nas páginas 573 e 574.
- AL-SHABI, M. et al. Lung nodule classification using deep local–global networks. *International journal of computer assisted radiology and surgery*, Springer, v. 14, n. 10, p. 1815–1819, 2019. Citado 2 vezes nas páginas 31 e 35.
- ALENCAR, F. E. S. Desenvolvimento de um sistema para classificação automática de imagens dermatoscópicas para dispositivos móveis. *Universidade do Estado do Rio Grande do Norte, Biblioteca Central Orlando Teixeira, BR-RN*, 2015. Citado 2 vezes nas páginas 142 e 146.
- ALHEETI, K. M. A. et al. Prediction of dos attacks in external communication for self-driving vehicles using a fuzzy petri net model. In: IEEE. 2016 IEEE Int. Conf. on Consumer Electronics (ICCE). [S.l.], 2016. p. 502–503. Citado 2 vezes nas páginas 468 e 486.
- ALI, F. A. et al. Mobile device power models for energy efficient dynamic offloading at runtime. *Journal of Systems and Software*, Elsevier, v. 113, p. 173–187, 2016. Citado na página 336.
- ALMAZROA, A. et al. Optic disc and optic cup segmentation methodologies for glaucoma image detection: A survey. *Journal of Ophthalmology*, 2015. Citado na página 299.
- ALMEIDA, G. F. *Fatores que interferem no comportamento enxameatório de abelhas africanizadas*. Tese (Doutorado) — Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, Ribeirão Preto, 2008. Citado na página 383.
- ALMOUSLI, H. *Recognition of Facial Expressions with Autoencoders and Convolutional-Nets*. Tese (Doutorado) — Université de Montréal, 2014. Disponível em: <http://hdl.handle.net/1866/10688>. Citado na página 39.
- ALTSCHUL, S. F. et al. Basic local alignment search tool. *Journal of Molecular Biology*, v. 215, n. 3, p. 403 – 410, 1990. ISSN 0022-2836. Citado na página 614.
- ALVES, J. et al. Urban sound event detection and classification. *i-ETC: ISEL Academic Journal of Electronics Telecommunications and Computers*, v. 6, n. 1, p. 2, 2020. Citado 3 vezes nas páginas 577, 578 e 585.
- ALVES, R. d. S. et al. Redes veiculares: Princípios, aplicações e desafios. *Minicursos do Simpósio Brasileiro de Redes de Computadores, SBRC*, p. 17–24, 2009. Citado na página 471.
- ALVES, S. M. A bioinformática e sua importância para a biologia molecular. *Revista Brasileira de Educação e Saúde*, v. 3, n. 4, p. 18–25, 2013. Citado na página 607.
- AMENTO, B. et al. *Systems and Methods For Allocating and Managing Resources in an Internet of Things Environment Using Location Based Focus of Attention*. [S.l.]: Google Patents, 2018. US Patent App. 15/432,042. Citado na página 400.

AMERICAN CANCER SOCIETY. *Early Detection, Diagnosis, And Staging*. 2019. Disponível em: <<https://www.cancer.org/cancer/melanoma-skin-cancer/detection-diagnosis-staging/how-diagnosed.html>>. Acesso em: 10 set. 2020. Citado na página 217.

AMMENDOLA, R. et al. A multi-port 10gbe pcie nic featuring udp offload and gpudirect capabilities. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2015. v. 664, n. 9, p. 092002. Citado na página 369.

AMORIM, M. A. P.; SIQUEIRA, K. Z. Relação entre vivência de fatores estressantes e surgimento de câncer de mama. *Psicologia Argumento*, v. 32, n. 79, 2017. Citado na página 119.

ANBALAGAN, B.; VALLIYAMMAI, C. Chennafloods - leveraging human and machine learning for crisis mapping during disasters using social media. In: IEEE. *2016 IEEE 23rd International Conference on High Performance Computing Workshops (HiPCW)*. [S.l.], 2016. p. 50–59. Citado na página 661.

ANBALAGAN, B.; VALLIYAMMAI, C. #chennafloods: Leveraging human and machine learning for crisis mapping during disasters using social media. In: *2016 IEEE 23rd International Conference on High Performance Computing Workshops (HiPCW)*. [S.l.: s.n.], 2016. p. 50–59. Citado 2 vezes nas páginas 668 e 670.

ANDERSEN, J. B. et al. *Systems and methods for implementing a sample rate converter using hardware and software to maximize speed and flexibility*. [S.l.]: Google Patents, 2007. US Patent 7,167,112. Citado na página 639.

ANDERSON, M. J.; WHITCOMB, P. J. *Design of experiments*. [S.l.]: Wiley Online Library, 2000. Citado na página 384.

ANDERSON, T. W. *The statistical analysis of time series*. [S.l.]: John Wiley & Sons, 2011. v. 19. Citado na página 447.

ANDRADE, E.; NOGUEIRA, B. Dependability evaluation of a disaster recovery solution for iot infrastructures. *The Journal of Supercomputing*, Springer, p. 1–22, 2018. Citado 2 vezes nas páginas 431 e 438.

ANTONI, J. The spectral kurtosis: a useful tool for characterising non-stationary signals. *Mechanical Systems and Signal Processing*, Elsevier, v. 20, n. 2, p. 282–307, 2006. Citado na página 655.

ANTONY, J. Taguchi or classical design of experiments: a perspective from a practitioner. *Sensor Review*, Emerald Group Publishing Limited, v. 26, n. 3, p. 227–230, 2006. Citado 3 vezes nas páginas 338, 366 e 402.

ARAI, K.; HERDIYENI, Y.; OKUMURA, H. Comparison of 2d and 3d local binary pattern in lung cancer diagnosis. *International Journal of Advanced Computer Science and Applications*, v. 3, 2012. Citado na página 26.

ARAUJO, E. et al. Dependability evaluation of an iot system: A hierarchical modelling approach. In: IEEE. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. [S.l.], 2019. p. 2121–2126. Citado 4 vezes nas páginas 328, 329, 438 e 534.

ARAUJO, F. *Descoberta de conhecimento em base de dados para o aprendizado da regulação médica/odontológica em operadora de plano de saúde*. Tese (Doutorado) — Dissertação de Mestrado, 2014. Citado na página 288.

ARAÚJO, F. H. et al. Redes neurais convolucionais com tensorflow: Teoria e prática. *SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. III Escola Regional de Informática do Piauí. Livro Anais-Artigos e Minicursos*, Sociedade Brasileira de Computação, v. 1, p. 382–406, 2017. Citado 5 vezes nas páginas 100, 237, 253, 282 e 284.

ARAÚJO, F. H. et al. Redes neurais convolucionais com tensorflow: Teoria e prática. *SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. III Escola Regional de Informática do Piauí. Livro Anais-Artigos e Minicursos*, Sociedade Brasileira de Computação, v. 1, p. 382–406, 2017. Citado 2 vezes nas páginas 165 e 593.

ARAÚJO, F. H.; SANTANA, A. M.; NETO, P. d. A. S. Using machine learning to support healthcare professionals in making preauthorisation decisions. *International journal of medical informatics*, Elsevier, v. 94, p. 1–7, 2016. Citado na página 595.

ARAUJO, F. H. et al. Reverse image search for scientific data within and beyond the visible spectrum. *Expert Systems with Applications*, v. 109, p. 35 – 48, 2018. ISSN 0957-4174. Citado 3 vezes nas páginas 590, 596 e 603.

- ARAUJO, F. H. et al. Deep learning for cell image segmentation and ranking. *Computerized Medical Imaging and Graphics*, v. 72, p. 13 – 21, 2019. ISSN 0895-6111. Citado na página 600.
- ARAÚJO, G. S. de. *Filogenia de Proteomas*. Tese (Doutorado) — Universidade Federal de Mato Grosso do Sul, 2003. Citado 2 vezes nas páginas 55 e 56.
- ARAUJO, J. Diagnóstico de glaucoma a partir de imagens de fundo de olho utilizando índices de diversidade. *Pós-graduação em ciência da computação, Universidade Federal do Maranhão, São Luís*, 2018. Citado 2 vezes nas páginas 9 e 10.
- ARAUJO, J. et al. Experimental evaluation of software aging effects on the eucalyptus cloud computing infrastructure. In: *Proceedings of the Middleware 2011 Industry Track Workshop*. [S.l.: s.n.], 2011. p. 1–7. Citado 2 vezes nas páginas 450 e 451.
- ARAUJO, J. et al. Software rejuvenation in eucalyptus cloud computing infrastructure: A method based on time series forecasting and multiple thresholds. In: IEEE. *2011 IEEE Third International Workshop on Software Aging and Rejuvenation*. [S.l.], 2011. p. 38–43. Citado 3 vezes nas páginas 444, 450 e 451.
- ARAUJO, J. et al. Dependability evaluation of a mhealth system using a mobile cloud infrastructure. In: IEEE. *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.], 2014. p. 1348–1353. Citado 2 vezes nas páginas 312 e 313.
- ARAUJO, J. et al. Dependability evaluation of a mhealth system using a mobile cloud infrastructure. *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, p. 1348–1353, 2014. Citado 2 vezes nas páginas 521 e 522.
- ARAÚJO, J. D. L. et al. Diagnóstico de glaucoma em imagens de fundo de olho utilizando os índices de diversidade de shannon e mcintosh. In: SBC. *17º Workshop de Informática Médica (WIM 2017)*. [S.l.], 2017. v. 17, n. 1/2017. Citado 3 vezes nas páginas 103, 125 e 221.
- ARAÚJO, J. D. L. et al. Glaucoma diagnosis in fundus eye images using diversity indexes. *Multimedia Tools and Applications*, v. 78, n. 10, p. 12987–13004, May 2019. ISSN 1573-7721. Disponível em: [⟨https://doi.org/10.1007/s11042-018-6429-z⟩](https://doi.org/10.1007/s11042-018-6429-z). Citado na página 163.
- ARAÚJO, T. et al. Classification of breast cancer histology images using convolutional neural networks. *PLoS one*, Public Library of Science, v. 12, n. 6, p. e0177544, 2017. Citado 5 vezes nas páginas 104, 105, 112, 263 e 274.
- ARESTA, G. et al. Bach: Grand challenge on breast cancer histology images. *Medical Image Analysis*, Elsevier, 2019. Citado 2 vezes nas páginas 104 e 105.
- ARESTA, G. et al. iw-net: an automatic and minimalistic interactive lung nodule segmentation deep network. *Scientific reports*, Nature Publishing Group, v. 9, n. 1, p. 1–9, 2019. Citado 3 vezes nas páginas 238, 239 e 244.
- AREVALO, J. et al. Representation learning for mammography mass lesion classification with convolutional neural networks. *Computer Methods and Programs in Biomedicine*, v. 127, p. 248 – 257, 2016. ISSN 0169-2607. Disponível em: [⟨http://www.sciencedirect.com/science/article/pii/S0169260715300110⟩](http://www.sciencedirect.com/science/article/pii/S0169260715300110). Citado 2 vezes nas páginas 81 e 93.
- ARISHANAPALLY, S. C. Building VGG19 with Keras. 2019. [⟨https://medium.com/@saicharanars/building-vgg19-with-keras-f516101c24cf⟩](https://medium.com/@saicharanars/building-vgg19-with-keras-f516101c24cf). [Online; accessed April-2019]. Citado na página 13.
- ARTHUR, D.; VASSILVITSKII, S. k-means++: The advantages of careful seeding. In: SOCIETY FOR INDUSTRIAL AND APPLIED MATHEMATICS. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. [S.l.], 2007. p. 1027–1035. Citado na página 236.
- ARYA, S.; PRATAP, N.; BHATIA, K. Future of face recognition: A review. *Procedia Computer Science*, v. 58, p. 578–585, 2015. ISSN 1877-0509. Second International Symposium on Computer Vision and the Internet (VisionNet'15). Disponível em: [⟨https://www.sciencedirect.com/science/article/pii/S1877050915021870⟩](https://www.sciencedirect.com/science/article/pii/S1877050915021870). Citado 2 vezes nas páginas 684 e 685.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Computer networks*, Elsevier, v. 54, n. 15, p. 2787–2805, 2010. Citado na página 516.

AUDU, G. A. et al. Reliability and quality of service of an off-grid wind powered roadside unit in a motorway vehicular environment. *Vehicular Communications*, Elsevier, v. 9, p. 176–187, 2017. Citado 2 vezes nas páginas 431 e 438.

AVIZIENIS, A. et al. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, IEEE, v. 1, n. 1, p. 11–33, 2004. Citado na página 446.

AVVENUTI, M. et al. Ears (earthquake alert and report system): a real time decision support system for earthquake crisis management. In: ACM. *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*. [S.I.], 2014. p. 1749–1758. Citado 2 vezes nas páginas 663 e 664.

AZEVEDO, E.; CONCI, A. *Computação gráfica: teoria e prática*. [S.I.]: Elsevier, 2008. Citado na página 88.

AZEVEDO, E.; CONCI, A.; VASCONCELOS, C. *Computação gráfica: Teoria e prática: geração de imagens*. [S.I.]: Elsevier Brasil, 2018. v. 1. Citado 2 vezes nas páginas 55 e 184.

AZEVEDO-MARQUES, P. M. d. Diagnóstico auxiliado por computador na radiologia. *Radiologia Brasileira*, SciELO Brasil, v. 34, n. 5, p. 285–293, 2001. Citado na página 217.

AZEVEDO-MARQUES, P. M. de. Diagnóstico auxiliado por computador na radiologia. *Radiologia Brasileira*, SciELO Brasil, v. 34, n. 5, p. 285–293, 2001. Citado 2 vezes nas páginas 120 e 136.

B., B. S. et al. Photocoagulation versus ranibizumab for proliferative diabetic retinopathy should baseline characteristics affect choice of treatment? In: *for the Diabetic Retinopathy Clinical Research Network*. [S.I.]: Publishing Press, 2019. p. 1646–1654. Citado na página 37.

BACAXIXI, P. et al. importância da apicultura no brasil. *Revista Científica Eletrônica de Agronomia*, v. 10, n. 20, 2011. Citado na página 383.

BACHHAWAT, A. K. Comparative genomics a powerful new tool in biology. *Resonance*, v. 11, n. 8, p. 22–40, 2006. Citado na página 610.

BADRI, H. et al. Multi-stage stochastic programming for service placement in edge computing systems: poster. In: ACM. *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*. [S.I.], 2017. p. 28. Citado 3 vezes nas páginas 313, 405 e 495.

BAELDUNG. *Dijkstra Shortest Path Algorithm in Java*. 2019. Disponível em: <https://www.baeldung.com/java-dijkstra>). Citado na página 543.

BAI, J. et al. Analyzing software rejuvenation techniques in a virtualized system: Service provider and user views. *IEEE Access*, IEEE, v. 8, p. 6448–6459, 2020. Citado na página 444.

BAI, J. et al. Music emotions recognition by cognitive classification methodologies. In: IEEE. *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2017 IEEE 16th International Conference on*. [S.I.], 2017. p. 121–129. Citado na página 646.

BAI, J. et al. Dimensional music emotion recognition by valence-arousal regression. In: IEEE. *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2016 IEEE 15th International Conference on*. [S.I.], 2016. p. 42–49. Citado 2 vezes nas páginas 646 e 650.

BAJWA, M. N. et al. Two-stage framework for optic disc localization and glaucoma classification in retinal fundus images using deep learning. *BMC medical informatics and decision making*, BioMed Central, v. 19, n. 1, p. 136, 2019. Citado 4 vezes nas páginas 169, 170, 177 e 178.

BAKILLAH, M. et al. Multi-agent transport simulation for regional evacuation processes. 2012. Citado na página 552.

BAKOS, L. Melanoma cutâneo: estudos de base populacional no brasil cutaneous melanoma: population-based studies in brazil. *An bras dermatol*, SciELO Brasil, v. 81, n. 5, p. 402, 2006. Citado na página 136.

- BAKSHI, N.; PRABHU, V. Face recognition system for access control using principal component analysis. In: *2017 International Conference on Intelligent Communication and Computational Techniques (ICCT)*. [S.l.: s.n.], 2017. p. 145–150. Citado 3 vezes nas páginas [691](#), [701](#) e [710](#).
- BANERJEE, S. et al. Melanoma diagnosis using deep learning and fuzzy logic. *Diagnostics*, Multidisciplinary Digital Publishing Institute, v. 10, n. 8, p. 577, 2020. Citado 2 vezes nas páginas [224](#) e [230](#).
- BARAKAT, C. et al. A flow-based model for internet backbone traffic. In: ACM. *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*. [S.l.], 2002. p. 35–47. Citado na página [362](#).
- BARATLOO, A. et al. Evidence based emergency medicine; part 1: Simple definition and calculation of accuracy, sensitivity and specificity. *Emergency*, v. 3, p. 48–49, 05 2015. Citado na página [595](#).
- BARATLOO, A. et al. Part 1: simple definition and calculation of accuracy, sensitivity and specificity. ARCHIVES OF ACADEMIC EMERGENCY MEDICINE (EMERGENCY), 2015. Citado na página [576](#).
- BARBETTE, T.; SOLDANI, C.; MATHY, L. Fast userspace packet processing. In: IEEE. *Architectures for Networking and Communications Systems (ANCS), 2015 ACM/IEEE Symposium on*. [S.l.], 2015. p. 5–16. Citado 3 vezes nas páginas [335](#), [369](#) e [370](#).
- BARIK, R. K. et al. Performance analysis of virtual machines and containers in cloud computing. In: IEEE. *Computing, Communication and Automation (ICCCA), 2016 International Conference on*. [S.l.], 2016. p. 1204–1210. Citado 2 vezes nas páginas [412](#) e [413](#).
- BARKER, S. A. *Comparison of Ring-Buffer-Based Packet Capture Solutions*. [S.l.], 2015. Citado 2 vezes nas páginas [369](#) e [370](#).
- BARKER, T.; VIRTANEN, T. Blind separation of audio mixtures through nonnegative tensor factorization of modulation spectrograms. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, IEEE, v. 24, n. 12, p. 2377–2389, 2016. Citado na página [573](#).
- BARKER, W. C. et al. The protein information resource (pir). *Nucleic Acids Research*, v. 28, n. 1, p. 41–44, 2000. Citado na página [611](#).
- BARSTUGAN, M.; OZKAYA, U.; OZTURK, S. Coronavirus (covid-19) classification using ct images by machine learning methods. *arXiv preprint arXiv:2003.09424*, 2020. Citado 3 vezes nas páginas [191](#), [192](#) e [201](#).
- BARTHET, M.; FAZEKAS, G.; SANDLER, M. Multidisciplinary perspectives on music emotion recognition: Implications for content and context-based models. *Proc. CMMR*, p. 492–507, 2012. Citado na página [634](#).
- BARTSCH, M. A.; WAKEFIELD, G. H. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on multimedia*, IEEE, v. 7, n. 1, p. 96–104, 2005. Citado na página [658](#).
- BASTOS, T. R. Especialista dá dicas para manter produção de mel em climas extremos. In: . [s.n.], 2016. Disponível em: <https://goo.gl/DEuRqd>. Acesso em: 24.03.2018. Citado na página [382](#).
- BATESOLE, K. et al. Yougenmap: a web platform for dynamic multi-comparative mapping and visualization of genetic maps. *Frontiers in Genetics*, v. 5, p. 183, 2014. ISSN 1664-8021. Citado na página [609](#).
- BATISTA, M. d. L. S. et al. Processamento digital de imagens para a detecção e classificação de nódulos em mamografias. 2010. Citado na página [116](#).
- BAXEVANIS, A. D.; OUELLETTE, B. F. *Bioinformatics: a practical guide to the analysis of genes and proteins*. [S.l.]: John Wiley & Sons, 2004. v. 43. Citado 2 vezes nas páginas [184](#) e [185](#).
- BAYSER, M. de; CERQUEIRA, R. Integrating mpi with docker for hpc. In: IEEE. *2017 IEEE International Conference on Cloud Engineering (IC2E)*. [S.l.], 2017. p. 259–265. Citado na página [446](#).
- BECK, M. T. et al. Mobile edge computing: A taxonomy. In: CITESEER. *Proc. of the Sixth International Conference on Advances in Future Internet*. [S.l.], 2014. p. 48–55. Citado 2 vezes nas páginas [404](#) e [489](#).

BECKEN, S. et al. Monitoring the environment and human sentiment on the great barrier reef: assessing the potential of collective sensing. *Journal of environmental management*, Elsevier, v. 203, p. 87–97, 2017. Citado 2 vezes nas páginas 669 e 670.

BEJNORDI, B. E. et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Jama*, American Medical Association, v. 318, n. 22, p. 2199–2210, 2017. Citado na página 248.

BELL, J. *Machine learning: hands-on for developers and technical professionals*. [S.l.]: John Wiley & Sons, 2020. Citado na página 449.

BELLO, J. P.; MYDLARZ, C.; SALAMON, J. Sound analysis in smart cities. In: *Computational Analysis of Sound Scenes and Events*. [S.l.]: Springer, 2018. p. 373–397. Citado 2 vezes nas páginas 570 e 590.

BENATIA, M. A. et al. A markov chain based model for congestion control in vanets. In: IEEE. 2013 27th International Conference on Advanced Information Networking and Applications Workshops. [S.l.], 2013. p. 1021–1026. Citado na página 427.

BENCHEKROUN, S.; MOURAOUAH, A. E. Tsunami evacuation plan for the city of tangier-morocco based on gis tools. *Journal of Tsunami Society International*, 2018. Citado 3 vezes nas páginas 549, 550 e 552.

BENETOS, E. Sound description toolbox. *MUSCLE Network of Excellence, Information & Software Engineering Group, Institute of Softwate Technology and Interactive Systems*, 2006. Citado na página 650.

BENSON, D. A. et al. Genbank. *Nucleic Acids Research*, v. 41, n. D1, p. D36–D42, 2013. Citado 2 vezes nas páginas 608 e 611.

BENSON, T.; AKELLA, A.; MALTZ, D. A. Network traffic characteristics of data centers in the wild. In: ACM. *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. [S.l.], 2010. p. 267–280. Citado na página 362.

BERMAN, H. M. et al. The protein data bank. *Nucleic Acids Research*, v. 28, n. 1, p. 235–242, 2000. Citado na página 611.

BERNARDO, V. et al. Green wireless-energy efficiency in wireless networks. *Large-scale Distributed Systems and Energy Efficiency: A Holistic View*, John Wiley & Sons Inc, v. 94, p. 81, 2015. Citado na página 334.

BERNHEIM, A. et al. Chest ct findings in coronavirus disease-19 (covid-19): Relationship to duration of infection. *Radiology*, v. 0, n. 0, p. 200463, 2020. PMID: 32077789. Citado na página 150.

BERNHEIM, A. et al. Chest ct findings in coronavirus disease-19 (covid-19): relationship to duration of infection. *Radiology*, Radiological Society of North America, p. 200463, 2020. Citado na página 182.

BERNSTEIN, D. Containers and cloud: From lxc to docker to kubernetes. *IEEE Cloud Computing*, IEEE, v. 1, n. 3, p. 81–84, 2014. Citado na página 447.

BHATTACHARYYA, S.; DIOT, C.; JETCHEVA, J. Pop-level and access-link-level traffic dynamics in a tier-1 pop. In: ACM. *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*. [S.l.], 2001. p. 39–53. Citado na página 362.

BHATTACHARYYA, S. et al. Data mining for credit card fraud: A comparative study. *Decision Support Systems*, Elsevier, v. 50, n. 3, p. 602–613, 2011. Citado na página 31.

BISNETO, T. R. V.; FILHO, A. O. de C.; MAGALHÃES, D. M. V. Generative adversarial network and texture features applied to automatic glaucoma detection. *Applied Soft Computing*, Elsevier, v. 90, p. 106165, 2020. Citado 5 vezes nas páginas 164, 169, 170, 177 e 178.

BISNETO, T. R. V.; OSEAS, A.; MAGALHÃES, D. M. V. Generative adversarial network and texture features applied to automatic glaucoma detection. *Applied Soft Computing*, v. 90, p. 106165, 2020. ISSN 1568-4946. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1568494620301058>. Citado 2 vezes nas páginas 9 e 10.

- BITAM, S.; MELLOUK, A. Markov-history based modeling for realistic mobility of vehicles in vanets. In: IEEE. 2013 IEEE 77th Vehicular Technology Conference (VTC Spring). [S.l.], 2013. p. 1–5. Citado na página 426.
- BLACKING, J. *How musical is man?* [S.l.]: University of Washington Press, 1974. Citado na página 636.
- BOCK, R. et al. Classifying glaucoma with image-based features from fundus photographs. In: SPRINGER. *Joint Pattern Recognition Symposium*. [S.l.], 2007. p. 355–364. Citado na página 59.
- BOGDANOV, D. et al. Essentia: An audio analysis library for music information retrieval. In: INTERNATIONAL SOCIETY FOR MUSIC INFORMATION RETRIEVAL (ISMIR). Britto A, Gouyon F, Dixon S, editors. *14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.* [place unknown]: ISMIR; 2013. p. 493-8. [S.l.], 2013. Citado 2 vezes nas páginas 650 e 657.
- BONAFINI, B. L.; PERLIN, H. A. Um estudo sobre reconhecimento de padrões aplicado a detecção de câncer do tipo melanoma maligno. *Revista Científica Interdisciplinar INTERLOGOS*, v. 3, n. 1, p. 137–154, 2018. Citado 2 vezes nas páginas 142 e 146.
- BORGES, J. et al. Feature engineering for crime hotspot detection. In: IEEE. 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation. [S.l.], 2017. p. 1–8. Citado na página 570.
- BORGIA, E. et al. Mobile edge clouds for information-centric iot services. In: IEEE. 2016 IEEE symposium on computers and communication (ISCC). [S.l.], 2016. p. 422–428. Citado 2 vezes nas páginas 310 e 490.
- Boroczky, L.; Zhao, L.; Lee, K. P. Feature subset selection for improving the performance of false positive reduction in lung nodule cad. In: *18th IEEE Symposium on Computer-Based Medical Systems (CBMS'05)*. [S.l.: s.n.], 2005. p. 85–90. ISSN 1063-7125. Citado 3 vezes nas páginas 66, 67 e 77.
- BOSWORTH, S.; KABAY, M. E.; WHYNE, E. *Computer Security Handbook*. [S.l.]: Wiley, 2014. v. 1. Citado na página 683.
- BOUGUILA, J.; KHOCHTALI, H. Facial plastic surgery and face recognition algorithms: Interaction and challenges. a scoping review and future directions. *Journal of Stomatology, Oral and Maxillofacial Surgery*, v. 121, n. 6, p. 696–703, 2020. ISSN 2468-7855. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2468785520301531>. Citado na página 682.
- BOUTREMANS, C.; IANNACCONE, G.; DIOT, C. Impact of link failures on voip performance. In: ACM. *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*. [S.l.], 2002. p. 63–71. Citado na página 362.
- BRADNER, S.; MCQUAID, J. Rfc 2544. *Benchmarking methodology for network interconnect devices*, 1999. Citado na página 372.
- BRAGA, A. Curvas roc: aspectos funcionais e aplicações. 2001. Citado 6 vezes nas páginas 58, 90, 141, 255, 271 e 289.
- BRAGA, L. F. Z. *Sistemas de Reconhecimento Facial*. 2013. Citado 3 vezes nas páginas 684, 687 e 688.
- BRAHMI, I. H.; DJAHEL, S.; GHAMRI-DOUDANE, Y. A hidden markov model based scheme for efficient and fast dissemination of safety messages in vanets. In: IEEE. 2012 IEEE Global Communications Conference (GLOBECOM). [S.l.], 2012. p. 177–182. Citado 2 vezes nas páginas 426 e 427.
- BRAY, F. et al. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, v. 68, n. 6, p. 394–424, 2018. Citado 2 vezes nas páginas 65 e 68.
- BRAY, F. et al. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, Wiley Online Library, v. 68, n. 6, p. 394–424, 2018. Citado na página 233.
- BRAZIL, S. M. I. Melanoma cutâneo no brasil. *Arquivos Catarinenses de Medicina*, v. 38, n. Suplemento 01, p. 14, 2009. Citado na página 137.

BREIMAN, L. Random forests. *Machine Learning*, Kluwer Academic Publishers, Hingham, MA, USA, v. 45, n. 1, p. 5–32, 2001. ISSN 0885-6125. Citado na página 594.

BREIMAN, L. et al. *Classification and regression trees*. [S.l.]: CRC press, 1984. Citado 2 vezes nas páginas 14 e 16.

BREITBACH, T.; SANDERS, P.; SCHULTES, D. Optimizing energy consumption and user experience in a mobile video streaming scenario. In: IEEE. *Consumer Communications & Networking Conference (CCNC), 2018 15th IEEE Annual*. [S.l.], 2018. p. 1–9. Citado 2 vezes nas páginas 341 e 342.

BRIGADE, D. S. A diferença entre inteligência artificial, machine learning e deep learning. URL: <https://medium.com/data-science-brigade/a-diferen%C3%A7a-entre-intelig%C3%A1ncia-artificial-machine-learning-e-deep-learning-930b5cc2aa42/> (visited on 14/05/2019), 2016. Citado 3 vezes nas páginas 84, 250 e 264.

BRINKER, B. D.; DINOTHER, R. V.; SKOWRONEK, J. Expressed music mood classification compared with valence and arousal ratings. *EURASIP Journal on Audio, Speech, and Music Processing*, Springer, v. 2012, n. 1, p. 24, 2012. Citado 4 vezes nas páginas 645, 652, 654 e 656.

BRIŠ, R. Evaluation of the production availability of an offshore installation by stochastic petri nets modeling. In: IEEE. *The International Conference on Digital Technologies 2013*. [S.l.], 2013. p. 147–155. Citado 3 vezes nas páginas 325, 432 e 434.

BRITANAK, V.; RAO, K. An efficient implementation of the forward and inverse mdct in mpeg audio coding. *IEEE Signal Processing Letters*, IEEE, v. 8, n. 2, p. 48–51, 2001. Citado na página 655.

BRITO, C. Internet no brasil: o problema está na infraestrutura. In: . Abranet - Associação Brasileira de Internet, 2016. Disponível em: <http://www.abranet.org.br/Artigos/Internet-no-Brasil-o-problema-esta-na-infraestrutura-1107.html>. Acesso em: 31.10.2018. Citado na página 360.

BRITO, S. H. B. *IPv6-O novo protocolo da Internet*. [S.l.]: Novatec Editora, 2018. Citado na página 360.

BROOKE, J. Sus: a “quick and dirty’usability. *Usability evaluation in industry*, CRC press, p. 189, 1996. Citado na página 614.

BROWN, J. C. Calculation of a constant q spectral transform. *The Journal of the Acoustical Society of America*, Acoustical Society of America, v. 89, n. 1, p. 425–434, 1991. Citado na página 574.

BROWNLEE, J. How to improve performance with transfer learning for deep learning neural networks. URL: <https://machinelearningmastery.com/how-to-improve-performance-with-transfer-learning-for-deep-learning-neural-networks/> (visited on 14/05/2019), 2019. Citado 2 vezes nas páginas 253 e 268.

BRUNO, B. C. R. et al. Câncer de mama: É possível prevenir? *REVISTA UNINGÁ REVIEW*, v. 28, n. 1, 2018. Citado na página 119.

BUITRAGO, F.; UEMURA, G.; SENA, M. C. F. Fatores prognósticos em câncer de mama. *Comun. ciênc. saúde*, v. 22, n. Sup. Espec. 1, p. 69–81, 2011. Citado na página 97.

BURRED, J. J.; LERCH, A. A hierarchical approach to automatic musical genre classification. In: CITESEER. *Proceedings of the 6th international conference on digital audio effects*. [S.l.], 2003. p. 8–11. Citado na página 657.

BUSSEL, I.; WOLLSTEN, G.; SCHUMAN, J. Oct for glaucoma diagnosis, screening and detection of glaucoma progression. *The British Journal of Ophthalmology*, 2013. Citado 2 vezes nas páginas 293 e 294.

BUTLER, B. *What is edge computing and how it's changing the network*. 2017. Disponível em: <https://www.networkworld.com/article/3224893/what-is-edge-computing-and-how-it-s-changing-the-network.html>. Acesso em: 11 abr. 2019. Citado na página 514.

BUTUN, E. et al. 1d-cadcapsnet: One dimensional deep capsule networks for coronary artery disease detection using ecg signals. *Physica Medica*, Elsevier, v. 70, p. 39–48, 2020. Citado na página 172.

- BUYYA, R.; YEO, C. S.; VENUGOPAL, S. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In: IEEE. 2008 10th IEEE international conference on high performance computing and communications. [S.l.], 2008. p. 5–13. Citado na página 516.
- CABREDO, R. et al. An emotion model for music using brain waves. In: ISMIR. [S.l.: s.n.], 2012. p. 265–270. Citado 3 vezes nas páginas 644, 650 e 652.
- CABRERA, D. et al. PsySound: A computer program for psychoacoustical analysis. In: Proceedings of the Australian Acoustical Society Conference. [S.l.: s.n.], 1999. v. 24, p. 47–54. Citado 2 vezes nas páginas 650 e 655.
- CÂMARA, G. et al. Spring: Integrating remote sensing and gis by object-oriented data modelling. *Computers & graphics*, Elsevier, v. 20, n. 3, p. 395–403, 1996. Citado na página 98.
- CAMARGO, J. Must dominance increase with the number of subordinate species in competitive interactions? *J Theor Biol*, v. 161, p. 537–542, 1993. Citado 2 vezes nas páginas 103 e 125.
- CAMPOLONGO, F.; TARANTOLA, S.; SALTELLI, A. Tackling quantitatively large dimensionality problems. *Computer Physics Communication*, Institute Jbr Systems, Informatics and Safety. Joint Research Centre of the European Commission, v. 117, n. 1, p. 75–85, 1999. Citado 3 vezes nas páginas 312, 425 e 520.
- CAMPOLONGO, F. et al. *Sensitivity analysis in practice: a guide to assessing scientific models*. [S.l.]: John Wiley and Sons, 2004. Citado 3 vezes nas páginas 312, 425 e 520.
- CAPELA, M. V.; CAPELA, J. M. Elaboração de gráficos box-plot em planilhas de cálculo. In: CONGRESSO DE MATEMÁTICA APLICADA E COMPUTACIONAL DA REGIÃO SUDESTE–CNMAC Sudeste. [S.l.: s.n.], 2011. v. 1. Citado na página 337.
- CARDOSO, J.; VALETTE, R. *Redes de petri*. [S.l.]: Universidade Federal de Santa Catarina, 1997. Citado na página 471.
- CAREY, C. L. et al. Additive deleterious effects of methamphetamine dependence and immunosuppression on neuropsychological functioning in hiv infection. *AIDS and Behavior*, Springer, v. 10, n. 2, p. 185, 2006. Citado na página 184.
- CARNEIRO, P. C. et al. Clahe parameters effects on the quantitative and visual assessment of dense breast mammograms. *IEEE Latin America Transactions*, IEEE, v. 17, n. 05, p. 851–857, 2019. Citado na página 219.
- CARNEIRO, T. et al. Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, IEEE, v. 6, p. 61677–61685, 2018. Citado 2 vezes nas páginas 47 e 110.
- CARROLL, A.; HEISER, G. An analysis of power consumption in a smartphone. 2010. Citado 3 vezes nas páginas 334, 341 e 342.
- CARROZZA, G. et al. Memory leak analysis of mission-critical middleware. *Journal of Systems and Software*, Elsevier, v. 83, n. 9, p. 1556–1567, 2010. Citado na página 444.
- CARUANA, R.; FREITAG, D. Greedy attribute selection. In: *Machine Learning Proceedings 1994*. [S.l.]: Elsevier, 1994. p. 28–36. Citado na página 91.
- CARVALHO, D. S. de et al. Aspectos gerais epidemiológicos da mortalidade por câncer de mama feminino no brasil e no mundo. *Anais do Simpósio de Enfermagem*, v. 1, n. 1, 2019. Citado na página 116.
- CARVALHO, E. D. et al. Breast cancer diagnosis from histopathological images using textural features and cbir. *Artificial Intelligence in Medicine*, v. 105, p. 101845, 2020. ISSN 0933-3657. Citado na página 595.
- CARVALHO, E. D.; FILHO, A. O. de C.; DALILIA, A. Classificação de tecidos da mama em maligno e benigno baseado em mamografias digitais usando descritores de textura. 2017. Citado 3 vezes nas páginas 104, 105 e 112.
- CARVALHO, E. D. et al. Method of differentiation of benign and malignant masses in digital mammograms using texture analysis based on phylogenetic diversity. *Computers & Electrical Engineering*, Elsevier, v. 67, p. 210–222, 2018. Citado 5 vezes nas páginas 56, 118, 119, 132 e 139.

- CARVALHO, E. D. et al. Method of differentiation of benign and malignant masses in digital mammograms using texture analysis based on phylogenetic diversity. *Computers & Electrical Engineering*, v. 67, p. 210 – 222, 2018. ISSN 0045-7906. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045790617334663>. Citado na página 150.
- CARVALHO, E. D. et al. Breast cancer diagnosis from histopathological images using textural features and cbir. *Artificial Intelligence in Medicine*, Elsevier, v. 105, p. 101845, 2020. Citado na página 190.
- CARVALHO-JÚNIOR, A. S. V. et al. Desenvolvimento de métodos para detecção automática do glaucoma. In: SBC. *Anais do XVII Workshop de Informática Médica*. [S.l.], 2017. Citado 2 vezes nas páginas 9 e 10.
- CARVALHO, L. M. T. de; JÚNIOR, F. W. A. Classificação digital do uso do solo comparando os métodos “pixel a pixel” e orientada ao objeto em imagem quickbird henriqueta veloso ferreira bernardi maíra dzedzej. 2007. Citado na página 99.
- CARVALHO, V. M. de. História e tradição da música militar. 2007. Citado na página 633.
- CASTELLANO, M. A.; BHARUCHA, J. J.; KRUMHANSL, C. L. Tonal hierarchies in the music of north india. *Journal of Experimental Psychology: General*, American Psychological Association, v. 113, n. 3, p. 394, 1984. Citado na página 636.
- CASTELLI, V. et al. Proactive management of software aging. *IBM Journal of Research and Development*, IBM, v. 45, n. 2, p. 311–332, 2001. Citado na página 446.
- CASTRO, A. A. M. D.; PRADO, P. P. L. D. Algoritmos para reconhecimento de padrões. *Revista Ciências Exatas*, v. 8, n. 2002, 2002. Citado na página 140.
- CASTRO, A. A. M. de; PRADO, P. P. L. do. Algoritmos para reconhecimento de padrões. *Revista Ciências Exatas*, v. 8, n. 2002, 2002. Citado na página 221.
- CAU, E. et al. Efficient exploitation of mobile edge computing for virtualized 5g in epc architectures. In: IEEE. *2016 4th IEEE international conference on mobile cloud computing, services, and engineering (MobileCloud)*. [S.l.], 2016. p. 100–109. Citado 2 vezes nas páginas 404 e 490.
- ČEPIN, M. Reliability block diagram. In: *Assessment of Power System Reliability*. [S.l.]: Springer, 2011. p. 119–123. Citado na página 422.
- CETIC.BR. Os provedores de acesso à internet no brasil: Como atuam e qual sua importância para o desenvolvimento da internet brasileira. In: . CGI.BR/NIC.BR, 2016. v. 8, n. 2. Disponível em: https://nic.br/media/docs/publicacoes/6/Panorama_Setorial_11_2.pdf. Acesso em: 31.10.2018. Citado na página 360.
- CHAE, J. et al. Public behavior response analysis in disaster events utilizing visual analytics of microblog data. *Computers & Graphics*, Elsevier, v. 38, p. 51–60, 2014. Citado 5 vezes nas páginas 662, 663, 664, 668 e 670.
- CHANEL, G.; ANSARI-ASL, K.; PUN, T. Valence-arousal evaluation using physiological signals in an emotion recall paradigm. In: IEEE. *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*. [S.l.], 2007. p. 2662–2667. Citado na página 637.
- CHANG, C.-Y.; LIAO, J.-J. Combination of rfid and face recognition for access control system. In: *2015 IEEE International Conference on Consumer Electronics - Taiwan*. [S.l.: s.n.], 2015. p. 470–471. Citado 4 vezes nas páginas 691, 692, 693 e 710.
- CHANG, H. Skin cancer reorganization and classification with deep neural network. *arXiv preprint arXiv:1703.00534*, 2017. Citado 3 vezes nas páginas 279, 280 e 292.
- CHAUDHARY, B. *Pele: Real-time Object Detection System on Mobile Devices*. 2019. <https://medium.com/@bibekchaudhary/pelee-real-time-object-detection-system-on-mobile-devices-f565947c04c4>. [Online; accessed August-2019]. Citado na página 13.
- CHEH, C.-C. et al. Affective understanding of online songs and speeches. In: IEEE. *Circuits and Systems (MWSCAS), 2010 53rd IEEE International Midwest Symposium on*. [S.l.], 2010. p. 363–366. Citado 2 vezes nas páginas 645 e 652.

- CHEN, L.; HA, W. Reliability prediction and qos selection for web service composition. *International Journal of Computational Science and Engineering*, v. 16, n. 2, p. 202–211, 2018. Citado na página 311.
- CHEN, M. et al. Edge cognitive computing based smart healthcare system. *Future Generation Computer Systems*, Elsevier, v. 86, p. 403–411, 2018. Citado 2 vezes nas páginas 521 e 522.
- CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 785–794. ISBN 9781450342322. Disponível em: (<https://doi.org/10.1145/2939672.2939785>). Citado na página 153.
- CHEN, T. et al. Xgboost: extreme gradient boosting. *R package version 0.4-2*, p. 1–4, 2015. Citado 2 vezes nas páginas 89 e 190.
- CHEN, X. et al. How is energy consumed in smartphone display applications? In: ACM. *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications*. [S.l.], 2013. p. 3. Citado 2 vezes nas páginas 341 e 342.
- CHEN, X. et al. Efficient multi-user computation offloading for mobile-edge cloud computing. *IEEE/ACM Transactions on Networking*, IEEE, v. 24, n. 5, p. 2795–2808, 2016. Citado 3 vezes nas páginas 405, 493 e 495.
- CHEN, X.-E. et al. A threshold autoregressive model for software aging. In: IEEE. *2006 Second IEEE International Symposium on Service-Oriented System Engineering (SOSE'06)*. [S.l.], 2006. p. 34–40. Citado na página 444.
- CHEN, Z. et al. Resnet and model fusion for automatic spoofing detection. In: *INTERSPEECH*. [S.l.: s.n.], 2017. p. 102–106. Citado 3 vezes nas páginas 86, 252 e 267.
- CHENG, H.-T. et al. Automatic chord recognition for music classification and retrieval. In: IEEE. *2008 IEEE International Conference on Multimedia and Expo (ICME)*. [S.l.], 2008. p. 1505–1508. Citado 3 vezes nas páginas 645, 654 e 658.
- CHENG, J.; DONG, L.; LAPATA, M. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*, 2016. Citado na página 449.
- CHENNA, R. et al. Multiple sequence alignment with the clustal series of programs. *Nucleic Acids Research*, v. 31, n. 13, p. 3497–3500, 2003. Citado na página 614.
- CHO, Y.-H. et al. Music emotion recognition using chord progressions. In: IEEE. *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*. [S.l.], 2016. p. 002588–002593. Citado 2 vezes nas páginas 646 e 650.
- CHOI, K.; SOMA, R.; PEDRAM, M. Dynamic voltage and frequency scaling based on workload decomposition. In: ACM. *Proceedings of the 2004 international symposium on Low power electronics and design*. [S.l.], 2004. p. 174–179. Citado na página 335.
- CHOI, M. Power and performance analysis of smart devices. *Int. J. Smart Home*, v. 7, n. 3, p. 57–66, 2013. Citado 3 vezes nas páginas 334, 341 e 342.
- CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016. Disponível em: (<http://arxiv.org/abs/1610.02357>). Citado na página 212.
- CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2017. p. 1251–1258. Citado 3 vezes nas páginas 88, 267 e 285.
- CHOLLET, F. *Deep Learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek*. [S.l.]: MITP-Verlags GmbH & Co. KG, 2018. Citado na página 281.
- CHOLLET, F. et al. Keras: The python deep learning library. *Astrophysics Source Code Library*, 2018. Citado na página 47.

CHONG, W. K. et al. Understanding infrastructure resiliency in chennai, india using twitter's geotags and texts: A preliminary study. *Engineering*, Elsevier, v. 4, n. 2, p. 218–223, 2018. Citado 2 vezes nas páginas 669 e 670.

CHOUDHARY, P. et al. A literature review on vehicular adhoc network for intelligent transport. In: IEEE. 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACoM). [S.l.], 2015. p. 2209–2213. Citado na página 421.

CHU, H.-h. et al. Roam, a seamless application framework. *Journal of Systems and Software*, Elsevier, v. 69, n. 3, p. 209–226, 2004. Citado 2 vezes nas páginas 335 e 336.

CHUA, L. O.; YANG, L. Cellular neural networks: Theory. *IEEE Transactions on circuits and systems*, IEEE, v. 35, n. 10, p. 1257–1272, 1988. Citado na página 117.

CHUNG, M. T. et al. Using docker in high performance computing applications. In: IEEE. 2016 IEEE Sixth International Conference on Communications and Electronics (ICCE). [S.l.], 2016. p. 52–57. Citado na página 446.

CIANCIARUSO, M. V.; SILVA, I. A.; BATALHA, M. A. Phylogenetic and functional diversities: new approaches to community ecology. *Biota Neotropica*, SciELO Brasil, v. 9, n. 3, p. 93–103, 2009. Citado na página 184.

CIANCIARUSO, M. V.; SILVA, I. A. A.; BATALHA, M. A. A. Diversidades filogenética e funcional: novas abordagens para a Ecologia de comunidades. In: . [S.l.]: scielo, 2009. p. 93 – 103. Citado na página 72.

ÇIÇEK, Ö. et al. 3d u-net: learning dense volumetric segmentation from sparse annotation. In: SPRINGER. International conference on medical image computing and computer-assisted intervention. [S.l.], 2016. p. 424–432. Citado na página 237.

CISCO, V. *Cisco Visual Networking Index: Forecast and Methodology 2016–2021.(2017)*. 2017. Citado na página 360.

CLARK, A. *Pillow (PIL Fork) Documentation*. readthedocs, 2015. Disponível em: <https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf>. Citado na página 172.

CLARKE, K.; WARWICK, R. A taxonomic distinctness index and its statistical properties. *Journal of applied ecology*, Wiley Online Library, v. 35, n. 4, p. 523–531, 1998. Citado 2 vezes nas páginas 55 e 56.

CLARKE, K.; WARWICK, R. A taxonomic distinctness index and its statistical properties. *Journal of Applied Ecology*, v. 35, p. 523 – 531, 08 1998. Citado 3 vezes nas páginas 184, 185 e 186.

CLARO, M. et al. An hybrid feature space from texture information and transfer learning for glaucoma classification. *Journal of Visual Communication and Image Representation*, v. 64, p. 102597, 2019. ISSN 1047-3203. Citado na página 595.

CODELLA, N. et al. Deep learning, sparse coding, and svm for melanoma recognition in dermoscopy images. In: SPRINGER. International Workshop on Machine Learning in Medical Imaging. [S.l.], 2015. p. 118–126. Citado na página 281.

COHEN, J. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, Sage Publications Sage CA: Thousand Oaks, CA, v. 20, n. 1, p. 37–46, 1960. Citado 5 vezes nas páginas 109, 576, 586, 595 e 603.

COHEN, J. Weighted kappa - nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological bulletin*, v. 70, p. 213–20, 11 1968. Citado na página 70.

COHEN, J. Weighted kappa - nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological bulletin*, v. 70, p. 213–20, 11 1968. Citado na página 167.

COHEN, J. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, American Psychological Association, v. 70, n. 4, p. 213, 1968. Citado na página 58.

- COLE, O.; EL-KHATIB, K. A privacy enhanced facial recognition access control system using biometric encryption. In: *2017 13th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. [S.l.: s.n.], 2017. p. 199–206. Citado 4 vezes nas páginas [691](#), [699](#), [700](#) e [710](#).
- COLUMBUS, L. *Roundup of Internet of Things Forecasts and Market Estimates, 2016*. 2016. Citado na página [384](#).
- COMPIETA, P. et al. Exploratory spatio-temporal data mining and visualization. *Journal of Visual Languages & Computing*, v. 18, n. 3, p. 255 – 279, 2007. ISSN 1045-926X. Visual Languages and Techniques for Human-GIS Interaction. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1045926X07000134>. Citado na página [662](#).
- CONCI, A.; AZEVEDO, E.; LETA, F. *Computação Gráfica - Teoria e Prática*. [S.l.: s.n.], 2008. ISBN 10: 85-352-2329-0. Citado na página [69](#).
- CONSORTIUM, T. F. The flybase database of the drosophila genome projects and community literature. *Nucleic Acids Research*, v. 31, n. 1, p. 172–175, 2003. Citado na página [611](#).
- CORCORAN, P.; DATTA, S. K. Mobile-edge computing and the internet of things for consumers: Extending cloud computing and services to the edge of the network. *IEEE Consumer Electronics Magazine*, IEEE, v. 5, n. 4, p. 73–74, 2016. Citado na página [493](#).
- CORTES, C.; VAPNIK, V. Support-vector networks. *Machine Learning*, v. 20, n. 3, p. 273–297, Sep 1995. ISSN 1573-0565. Disponível em: <https://doi.org/10.1007/BF00994018>. Citado na página [73](#).
- CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995. Citado 2 vezes nas páginas [576](#) e [595](#).
- COSTA, E. S. et al. Análise da mortalidade e distribuição de neoplasia cutânea em sergipe no período de 2010 a 2015. In: *Congresso Nacional de Enfermagem-CONENF*. [S.l.: s.n.], 2019. v. 1, n. 1. Citado na página [137](#).
- COSTA, I. et al. Availability evaluation and sensitivity analysis of a mobile backend-as-a-service platform. *Quality and Reliability Engineering International*, Wiley Online Library, v. 32, n. 7, p. 2191–2205, 2016. Citado 2 vezes nas páginas [312](#) e [313](#).
- COTRONEO, D. et al. Software aging analysis of the linux operating system. In: *IEEE. 2010 IEEE 21st International Symposium on Software Reliability Engineering*. [S.l.], 2010. p. 71–80. Citado na página [444](#).
- COTRONEO, D. et al. A survey of software aging and rejuvenation studies. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, Acm New York, NY, USA, v. 10, n. 1, p. 1–34, 2014. Citado na página [454](#).
- COUTINHO, E.; CANGELOSI, A. Musical emotions: predicting second-by-second subjective feelings of emotion from low-level psychoacoustic features and physiological measurements. *Emotion*, American Psychological Association, v. 11, n. 4, p. 921, 2011. Citado na página [633](#).
- COUTINHO-RODRIGUES, L. T. J.; ALÇADA-ALMEIDA, L. Solving a location-routing problem with a multiobjective approach: the design of urban evacuation plans. *Journal of Transport Geography*, 2012. Citado 2 vezes nas páginas [548](#) e [550](#).
- CRAMER, J. et al. Look, listen, and learn more: Design choices for deep audio embeddings. In: *IEEE. ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2019. p. 3852–3856. Citado 4 vezes nas páginas [590](#), [596](#), [597](#) e [603](#).
- CRYER, J. D.; KELLET, N. *Time series analysis*. [S.l.]: Springer, 1991. Citado na página [447](#).
- CUERVO, E. et al. Maui: making smartphones last longer with code offload. In: *ACM. Proceedings of the 8th international conference on Mobile systems, applications, and services*. [S.l.], 2010. p. 49–62. Citado 3 vezes nas páginas [335](#), [336](#) e [517](#).
- DABYDEEN, A. *Transfer Learning Using ResNet50 and CIFAR-10*. 2019. <https://mc.ai/transfer-learning-using-resnet50-and-cifar-10/>. [Online; accessed May-2019]. Citado na página [14](#).

- DAI, W. et al. Very deep convolutional neural networks for raw waveforms. In: IEEE. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2017. p. 421–425. Citado na página 595.
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: . [S.l.: s.n.], 2005. Citado na página 12.
- DANIEL, G. *Principles of artificial neural networks*. [S.l.]: World Scientific, 2013. v. 7. Citado 2 vezes nas páginas 83 e 281.
- DANTAS, J. et al. Models for dependability analysis of cloud computing architectures for eucalyptus platform. *International Transactions on Systems Science and Applications*, v. 8, n. 5, p. 13–25, 2012. Citado 2 vezes nas páginas 312 e 313.
- DAVIDSON, J. A. et al. How the diabetic eye loses vision. *Endocrine*, Springer, v. 32, n. 1, p. 107–116, 2007. Citado na página 206.
- DAVIES, M. E.; PLUMBLEY, M. D. Context-dependent beat tracking of musical audio. *IEEE Transactions on Audio, Speech, and Language Processing*, IEEE, v. 15, n. 3, p. 1009–1020, 2007. Citado na página 657.
- de Carvalho Filho, A. O. et al. Classification of patterns of benignity and malignancy based on ct using topology-based phylogenetic diversity index and convolutional neural network. *Pattern Recognition*, v. 81, p. 200 – 212, 2018. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320318301237>. Citado na página 150.
- de Carvalho Junior, A. S. V. et al. Automatic methods for diagnosis of glaucoma using texture descriptors based on phylogenetic diversity. *Computers & Electrical Engineering*, v. 71, p. 102 – 114, 2018. ISSN 0045-7906. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045790617338570>. Citado na página 150.
- de Oliveira Torres, W. et al. Texture analysis of lung nodules in computerized tomography images using functional diversity. *Computers & Electrical Engineering*, v. 84, p. 106618, 2020. ISSN 0045-7906. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0045790620304730>. Citado na página 150.
- DECENCIÈRE, E. et al. Feedback on a publicly distributed database: the messidor database. *Image Analysis & Stereology*, v. 33, n. 3, p. 231–234, ago. 2014. ISSN 1854-5165. Disponível em: <http://www.ias-iss.org/ojs/IAS/article/view/1155>. Citado na página 207.
- DEMIR, F.; ABDULLAH, D. A.; SENGUR, A. A new deep cnn model for environmental sound classification. *IEEE Access*, IEEE, v. 8, p. 66529–66537, 2020. Citado 4 vezes nas páginas 590, 596, 597 e 603.
- DENG, J. et al. ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09. [S.l.: s.n.], 2009. Citado na página 16.
- DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. *2009 IEEE conference on computer vision and pattern recognition*. [S.l.], 2009. p. 248–255. Citado 3 vezes nas páginas 90, 101 e 107.
- DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. *2009 IEEE conference on computer vision and pattern recognition*. [S.l.], 2009. p. 248–255. Citado na página 177.
- DERI, L. et al. Improving passive packet capture: Beyond device polling. In: AMSTERDAM, NETHERLANDS. *Proceedings of SANE*. [S.l.], 2004. v. 2004, p. 85–93. Citado na página 366.
- DERI, L.; RING, M. P. Url http://www.ntop.org/PF_RING.html, 2011. Citado na página 361.
- DESROCHERS, A.; AL-JAAR, R.; SOCIETY, I. C. S. *Applications of petri nets in manufacturing systems: modeling, control, and performance analysis*. [S.l.]: IEEE Press, 1995. ISBN 9780879422950. Citado na página 490.
- DEUTSCH, W. Introduction to electronic access control. 2018. Citado na página 683.
- DEWANTARA, B. S. B.; BACHTIAR, M. M.; LANTANG, S. E. Door access control based on illumination invariant face recognition in embedded system. In: 2020 10th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS). [S.l.: s.n.], 2020. p. 105–110. Citado 5 vezes nas páginas 691, 701, 702, 709 e 710.

- DEY, R.; LU, Z.; HONG, Y. Diagnostic classification of lung nodules using 3d neural networks. In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. [S.l.: s.n.], 2018. p. 774–778. ISSN 1945-8452. Citado 3 vezes nas páginas [31](#), [35](#) e [71](#).
- DICE, L. Measures of the amount of ecologic association between species. *Ecology*, p. 297–302, 1945. Citado na página [302](#).
- DIMITROV, D. V. Medical internet of things and big data in healthcare. *Healthcare informatics research*, v. 22, n. 3, p. 156–163, 2016. Citado na página [384](#).
- DINH, H. T. et al. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, Wiley Online Library, v. 13, n. 18, p. 1587–1611, 2013. Citado 3 vezes nas páginas [310](#), [335](#) e [490](#).
- DJILALI, Y. D. et al. Performances evaluation study of vanet communication technologies for smart and autonomous vehicles. In: *IEEE. 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*. [S.l.], 2018. p. 79–84. Citado 2 vezes nas páginas [482](#) e [483](#).
- DOOLITTLE, R. F. *Molecular evolution: computer analysis of protein and nucleic acid sequences*. [S.l.: s.n.], 1990. Citado na página [610](#).
- DOWNIE, J. S. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, Acoustical Society of Japan, v. 29, n. 4, p. 247–255, 2008. Citado na página [639](#).
- DRUCKER, H. et al. Support vector regression machines. In: *Advances in neural information processing systems*. [S.l.: s.n.], 1997. p. 155–161. Citado na página [108](#).
- DU, B. et al. Stacked convolutional denoising auto-encoders for feature representation. *IEEE transactions on cybernetics*, IEEE, v. 47, n. 4, p. 1017–1027, 2017. Citado na página [60](#).
- DU, X. et al. Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection. In: *IEEE. 2017 IEEE winter conference on applications of computer vision (WACV)*. [S.l.], 2017. p. 953–961. Citado 3 vezes nas páginas [85](#), [253](#) e [268](#).
- DUA, R.; RAJA, A. R.; KAKADIA, D. Virtualization vs containerization to support paas. In: *IEEE. Cloud Engineering (IC2E), 2014 IEEE International Conference on*. [S.l.], 2014. p. 610–614. Citado na página [400](#).
- DUBNOV, S. Generalization of spectral flatness measure for non-gaussian linear processes. *IEEE Signal Processing Letters*, IEEE, v. 11, n. 8, p. 698–701, 2004. Citado 2 vezes nas páginas [573](#) e [574](#).
- DUDA, R. O.; HART, P. E. *Pattern Classification and Scene Analysis*. New York: Wiley-Interscience Publication, 1973. Citado na página [70](#).
- DUDA, R. O.; HART, P. E. et al. *Pattern classification and scene analysis*. [S.l.]: Wiley New York, 1973. v. 3. Citado na página [595](#).
- DUTRA, T. F. S. *Beehiveior-Sistema de monitoramento e controle de colmeias de produção apícola*. Dissertação (Mestrado) — Brasil, 2016. Citado na página [388](#).
- EDWARDS, J. Signal processing opens the internet of things to a new world of possibilities: Research leads to new internet of things technologies and applications [special reports]. *IEEE Signal Processing Magazine*, IEEE, v. 35, n. 5, p. 9–12, 2018. Citado 2 vezes nas páginas [570](#) e [578](#).
- EEROLA, T. Modeling listeners' emotional response to music. *Topics in cognitive science*, Wiley Online Library, v. 4, n. 4, p. 607–624, 2012. Citado 4 vezes nas páginas [645](#), [652](#), [654](#) e [659](#).
- EEROLA, T.; LARTILLOT, O.; TOIVIAINEN, P. Prediction of multidimensional emotional ratings in music from audio using multivariate regression models. In: *ISMIR*. [S.l.: s.n.], 2009. p. 621–626. Citado na página [648](#).

El-Regaily, S. A. et al. Lung nodule segmentation and detection in computed tomography. In: *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*. [S.l.: s.n.], 2017. p. 72–78. Citado 3 vezes nas páginas [66](#), [67](#) e [77](#).

ELLIS, D. Chroma feature analysis and synthesis. *Resources of Laboratory for the Recognition and Organization of Speech and Audio-LabROSA*, 2007. Citado 2 vezes nas páginas [573](#) e [574](#).

ELMORE, J. G. et al. Diagnostic concordance among pathologists interpreting breast biopsy specimens. *Jama*, American Medical Association, v. 313, n. 11, p. 1122–1132, 2015. Citado na página [96](#).

ELSHEIKH, S. H.; MUSTAFA, Z. A.; IBRAHEEM, B. A. Computer aided diagnosis system of skin cancer using artificial neural network. 2018. Citado 2 vezes nas páginas [80](#) e [278](#).

EMMERICH, P. et al. Optimizing latency and cpu load in packet processing systems. In: IEEE. *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2015 International Symposium on*. [S.l.], 2015. p. 1–8. Citado na página [360](#).

EPPOLITO, A. *Multi-channel sound panner*. [S.l.]: Google Patents, 2008. US Patent App. 11/786,863. Citado na página [572](#).

ERCAL, F. et al. Neural network diagnosis of malignant melanoma from color images. *IEEE Transactions on biomedical engineering*, IEEE, v. 41, n. 9, p. 837–845, 1994. Citado na página [278](#).

ERKEL, A. R. V.; PETER, M. Receiver operating characteristic (roc) analysis: basic principles and applications in radiology. *European Journal of radiology*, Elsevier, v. 27, n. 2, p. 88–94, 1998. Citado na página [58](#).

ERKEL, A. R. van; PATTYNAMA, P. M. Receiver operating characteristic (roc) analysis: Basic principles and applications in radiology. *European Journal of Radiology*, v. 27, n. 2, p. 88 – 94, 1998. ISSN 0720-048X. Citado 2 vezes nas páginas [167](#) e [191](#).

ESKOV, E.; TOBOEV, V. Seasonal dynamics of thermal processes in aggregations of wintering honey bees (*apis mellifera*, hymenoptera, apidae). *Entomological review*, Springer, v. 91, n. 3, p. 354–359, 2011. Citado 2 vezes nas páginas [382](#) e [383](#).

ESMAIL, M. Y.; AHMED, D. H.; ELTAYEB, M. Classification system for heart sounds based on random forests. *Journal of Clinical Engineering*, LWW, v. 44, n. 2, p. 76–80, 2019. Citado na página [576](#).

ESPEJO, M. R. *Design of experiments for engineers and scientists*. [S.l.]: Taylor & Francis, 2006. Citado 3 vezes nas páginas [338](#), [366](#) e [402](#).

ESQUEF, I. A.; ALBUQUERQUE, M. P. d.; ALBUQUERQUE, M. P. d. Processamento digital de imagens. CENTRO BRASILEIRO DE PESQUISAS FÍSICAS–CBPF, 2003. Citado na página [98](#).

ESTEVA, A. et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, Nature Publishing Group, v. 542, n. 7639, p. 115, 2017. Citado 3 vezes nas páginas [279](#), [280](#) e [292](#).

FAITH, D. P. Conservation evaluation and phylogenetic diversity. *Biological Conservation*, v. 61, n. 1, p. 1 – 10, 1992. ISSN 0006-3207. Disponível em: [\(<http://www.sciencedirect.com/science/article/pii/0006320792912013>\)](http://www.sciencedirect.com/science/article/pii/0006320792912013). Citado 2 vezes nas páginas [184](#) e [185](#).

FANG, Z. et al. A space–time efficiency model for optimizing intra-intersection vehicle–pedestrian evacuation movements. *Transportation Research Part C*, 2013. Citado 2 vezes nas páginas [549](#) e [550](#).

FARIAS, A. M. L. de. O boxplot. *Universidade Federal Fluminense-Instituto de Matemática*, 2013. Citado na página [337](#).

FARID, A. A.; SELIM, G.; KHATER, H. A novel approach of ct images feature analysis and prediction to screen for corona virus disease (covid-19). *International Journal of Scientific and Engineering Research*, v. 11, p. 1141, 03 2020. Citado 2 vezes nas páginas [155](#) e [160](#).

- FAROOQ, M. U.; PASHA, M.; KHAN, K. U. R. A data dissemination model for cloud enabled vanets using in-vehicular resources. In: IEEE. 2014 International Conference on Computing for Sustainable Global Development (INDIACom). [S.l.], 2014. p. 458–462. Citado na página 422.
- FEIJOO, J. et al. DRIONS-DB: digital retinal images for optic nerve segmentation database. [S.l.]: Academic Press, 2014. Citado na página 45.
- FELD, S. *Music grooves: Essays and dialogues*. [S.l.]: University of Chicago Press, 1994. Citado na página 636.
- FERNANDES, B. J. T. Redes neurais com extração implícita de características para reconhecimento de padrões visuais. Universidade Federal de Pernambuco, 2013. Citado na página 39.
- FERNANDES, C. P. et al. Um sistema de reputação descentralizado para avaliar a confiança dos nós em redes veiculares. *Nao informado*, Universidade do Vale do Itajaí, 2013. Citado na página 468.
- FERREIRA, A. B. d. H. Novo aurélio: o dicionário da língua portuguesa séc. XXI. Rio de Janeiro: Editora Nova Fronteira, 1999. Citado na página 664.
- FERREIRA, C. A. et al. Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 763–770. Citado 2 vezes nas páginas 104 e 105.
- FERREIRA, C. B. R.; BORGES, D. L. Analysis of mammogram classification using a wavelet transform decomposition. *Pattern Recognition Letters*, Elsevier, v. 24, n. 7, p. 973–982, 2003. Citado na página 79.
- FERREIRA, L. et al. Optimizing resource availability in composable data center infrastructures. In: IEEE. 2019 9th Latin-American Symposium on Dependable Computing (LADC). [S.l.], 2019. p. 1–10. Citado na página 310.
- FERREIRA, M. et al. Processamento digital de imagens mÃ©dicas com python e opencv. In: _____. [S.l.: s.n.], 2017. ISBN 978-85-8320-201-1 331. Citado na página 121.
- FILHO, A. C. et al. Automatic detection of solitary lung nodules using quality threshold clustering, genetic algorithm and diversity index. *Artificial intelligence in medicine*, Elsevier, v. 60, n. 3, p. 165–177, 2014. Citado 4 vezes nas páginas 25, 32, 34 e 182.
- FILHO, A. O. d. C. et al. Métodos para sistemas cad e cadx de nódulo pulmonar baseada em tomografia computadorizada usando análise de forma e textura. Universidade Federal do Maranhão, 2016. Citado 6 vezes nas páginas 55, 98, 99, 116, 126 e 594.
- FILHO, A. O. d. C. et al. Métodos para sistemas cad e cadx de nódulo pulmonar baseada em tomografia computadorizada usando análise de forma e textura. Universidade Federal do Maranhão, 2016. Citado na página 164.
- FILHO, A. O. d. C. et al. 3d shape analysis to reduce false positives for lung nodule detection systems. *Medical & Biological Engineering & Computing*, v. 55, n. 8, p. 1199–1213, Aug 2017. Disponível em: <https://doi.org/10.1007/s11517-016-1582-x>. Citado 3 vezes nas páginas 67, 71 e 77.
- FILHO, A. O. de C. et al. Automatic detection of solitary lung nodules using quality threshold clustering, genetic algorithm and diversity index. *Artificial Intelligence in Medicine*, v. 60, n. 3, p. 165 – 177, 2014. ISSN 0933-3657. Citado 2 vezes nas páginas 70 e 71.
- FILHO, A. O. de C. et al. Automatic detection of solitary lung nodules using quality threshold clustering, genetic algorithm and diversity index. *Artificial intelligence in medicine*, Elsevier, v. 60, n. 3, p. 165–177, 2014. Citado na página 193.
- FILHO, A. O. de C. et al. Automatic detection of solitary lung nodules using quality threshold clustering, genetic algorithm and diversity index. *Artificial Intelligence in Medicine*, v. 60, n. 3, p. 165 – 177, 2014. ISSN 0933-3657. Citado na página 204.
- FILHO, A. O. de C. et al. Computer-aided diagnosis of lung nodules in computed tomography by using phylogenetic diversity, genetic algorithm, and svm. *Journal of Digital Imaging*, Springer, p. 1–11, 2017. Citado 4 vezes nas páginas 52, 238, 239 e 244.

FILHO, A. O. de C. et al. Lung-nodule classification based on computed tomography using taxonomic diversity indexes and an svm. *Journal of Signal Processing Systems*, Springer, v. 87, n. 2, p. 179–196, 2017. Citado 2 vezes nas páginas [56](#) e [61](#).

FILHO, A. O. de C. et al. Lung-nodule classification based on computed tomography using taxonomic diversity indexes and an svm. *Journal of Signal Processing Systems*, v. 87, n. 2, p. 179–196, May 2017. ISSN 1939-8115. Disponível em: <https://doi.org/10.1007/s11265-016-1134-5>. Citado na página [71](#).

FILHO, A. O. de C. et al. Classification of patterns of benignity and malignancy based on ct using topology-based phylogenetic diversity index and convolutional neural network. *Pattern Recognition*, Elsevier, v. 81, p. 200–212, 2018. Citado na página [182](#).

FILHO, J. E. M. *Análise de Tráfego em Redes TCP/IP: Utilize tcpdump na análise de tráfegos em qualquer sistema operacional*. [S.l.]: Novatec Editora, 2013. Citado na página [362](#).

FILHO, O. M.; NETO, H. V. *Processamento digital de imagens*. [S.l.]: Brasport, 1999. Citado 3 vezes nas páginas [54](#), [121](#) e [206](#).

FINK, J. Docker: a software as a service, operating system-level virtualization framework. *Code4Lib Journal*, v. 25, p. 29, 2014. Citado na página [400](#).

FLANAGAN, D. *JavaScript: the definitive guide*. [S.l.]: "O'Reilly Media, Inc.", 2006. Citado na página [387](#).

FONG, D. S. et al. Retinopathy in diabetes. *Diabetes Care*, American Diabetes Association, v. 27, n. suppl 1, p. s84–s87, 2004. ISSN 0149-5992. Disponível em: https://care.diabetesjournals.org/content/27/suppl_1/s84. Citado na página [204](#).

FONG, D. S. et al. Retinopathy in diabetes. 2016. Citado na página [39](#).

FOOTE, J.; UCHIHASHI, S. The beat spectrum: A new approach to rhythm analysis. In: IEEE. *null*. [S.l.], 2001. p. 224. Citado na página [657](#).

FOOTE, J. T.; COOPER, M. L. Media segmentation using self-similarity decomposition. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Storage and Retrieval for Media Databases 2003*. [S.l.], 2003. v. 5021, p. 167–176. Citado na página [655](#).

FOSTER, P. J. et al. The definition and classification of glaucoma in prevalence surveys. *British journal of ophthalmology*, BMJ Publishing Group Ltd, v. 86, n. 2, p. 238–242, 2002. Citado 2 vezes nas páginas [53](#) e [54](#).

FOX40. *List of Missing in Camp Fire Down to 1*. 2019. Disponível em: <http://bit.ly/2Oqrka1>. Citado na página [551](#).

FR, B. of Volcanology Editor-in-Chief andrew.harris@ uca. Eruption crisis at kilauea caldera (big island of hawaii, usa). *Bulletin of Volcanology*, Springer, v. 80, p. 1–2, 2018. Citado na página [670](#).

FRALEIGH, C. et al. Packet-level traffic measurements from the sprint ip backbone. *IEEE network*, IEEE, v. 17, n. 6, p. 6–16, 2003. Citado na página [362](#).

FRANCÊS, C. R. L. Introdução às redes de petri. *Laboratório de Computação Aplicada, Universidade Federal do Pará*, 2003. Citado na página [472](#).

FRANCO, G. *Como utilizar o teste t de student*. 2015. Disponível em: <http://manipulandodados.com.br/como-utilizar-o-teste-t-de-student/>. Acesso em: 18 oct. 2017. Citado na página [336](#).

FREUND, Y.; SCHAPIRE, R. E. A desicion-theoretic generalization of on-line learning and an application to boosting. In: SPRINGER. *European conference on computational learning theory*. [S.l.], 1995. p. 23–37. Citado 2 vezes nas páginas [15](#) e [16](#).

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, v. 28, p. 337–407, 04 2000. Citado na página [89](#).

FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, Institute of Mathematical Statistics, v. 29, n. 5, p. 1189–1232, 2001. ISSN 00905364. Citado na página [89](#).

- Fujihara, A.; Miwa, H. Real-time disaster evacuation guidance using opportunistic communications. In: *2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet*. [S.l.: s.n.], 2012. p. 326–331. Citado 2 vezes nas páginas [676](#) e [678](#).
- FUMERO, F. et al. Rim-one: An open retinal image database for optic nerve evaluation. In: . [S.l.: s.n.], 2011. p. 1 – 6. Citado na página [44](#).
- FUMERO, F. et al. Rim-one: An open retinal image database for optic nerve evaluation. In: . [S.l.: s.n.], 2011. p. 1 – 6. Citado 3 vezes nas páginas [165](#), [169](#) e [298](#).
- GADDIPATI, D. J. et al. Glaucoma assessment from oct images using capsule network. In: *IEEE. 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. [S.l.], 2019. p. 5581–5584. Citado 6 vezes nas páginas [162](#), [168](#), [169](#), [170](#), [172](#) e [178](#).
- GAFFURIUS, F. *Theorica musice*. [S.l.]: Edizioni del Galluzzo per la Fondazione Ezio Franceschini, 2005. v. 2. Citado na página [638](#).
- GALLENMÜLLER, S. Comparison of memory mapping techniques for high-speed packet processing. *Technical University of Munich*, 2014. Citado na página [369](#).
- GALLENMÜLLER, S. et al. Comparison of frameworks for high-performance packet io. In: *IEEE COMPUTER SOCIETY. Proceedings of the Eleventh ACM/IEEE Symposium on Architectures for networking and communications systems*. [S.l.], 2015. p. 29–38. Citado 2 vezes nas páginas [369](#) e [370](#).
- GAUR, N.; JOSHI, P.; SRIVASTAVA, R. Modelling database server sizing for concurrent users using coloured petri-nets. In: *IEEE. 2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*. [S.l.], 2017. p. 90–94. Citado na página [414](#).
- GERSHO, A. Principles of quantization. *IEEE Transactions on Circuits and Systems*, v. 25, n. 7, p. 427–436, Jul 1978. ISSN 0098-4094. Citado na página [73](#).
- GHANEM, H. de O. S. A. *Glaucoma*. 2020. <https://sadalla.com.br/especialidades/glaucoma/>. Acesso em: 26 de dez. de 2020. Citado na página [164](#).
- GIANNAKOPOULOS, T.; SPYROU, E.; PERANTONIS, S. J. Recognition of urban sound events using deep context-aware feature extractors and handcrafted features. In: SPRINGER. *IFIP International Conference on Artificial Intelligence Applications and Innovations*. [S.l.], 2019. p. 184–195. Citado 5 vezes nas páginas [573](#), [577](#), [578](#), [585](#) e [590](#).
- GIANNINI, T. C. Desafios atuais da modelagem preditiva de distribuição de espécies. *Rodriguésia-Instituto de Pesquisas Jardim Botânico do Rio de Janeiro*, v. 63, n. 3, 2012. Citado 2 vezes nas páginas [255](#) e [271](#).
- GILLAN, C. J. et al. Expediting assessments of database performance for streams of respiratory parameters. *Computers in biology and medicine*, Elsevier, v. 100, p. 186–195, 2018. Citado na página [414](#).
- GILLANI, K.; LEE, J.-H. Comparison of linux virtual machines and containers for a service migration in 5g multi-access edge computing. *ICT Express*, Elsevier, v. 6, n. 1, p. 1–2, 2020. Citado 2 vezes nas páginas [444](#) e [446](#).
- GINSBERG, J. R.; YOUNG, T. P. Measuring association between individuals or groups in behavioural studies. *Animal Behaviour*, Elsevier Science, v. 44, n. 1, p. 377–379, 1992. Citado 2 vezes nas páginas [38](#) e [46](#).
- GIRSHICK, R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Jun 2014. Disponível em: <http://dx.doi.org/10.1109/CVPR.2014.81>. Citado na página [150](#).
- GLASBEY, C. A.; HORGAN, G. W. *Image analysis for the biological sciences*. [S.l.]: Wiley, 1998. Citado na página [236](#).
- GOLATKAR, A.; ANAND, D.; SETHI, A. Classification of breast cancer histology using deep learning. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 837–844. Citado 7 vezes nas páginas [104](#), [105](#), [112](#), [192](#), [262](#), [263](#) e [274](#).

- GOMES. *Cancer de pele*. 2013. ([https://www\[minhavida.com.br/saude/temas/cancer-de-pele/](https://www[minhavida.com.br/saude/temas/cancer-de-pele/)]). [Online; accessed 19-November-2017]. Citado na página 137.
- GOMES, C. N. *Estudo do paradigma: computação em nuvem*. Tese (Doutorado), 2012. Citado na página 516.
- GÓMEZ, E.; Klapuri, A.; MEUDIC, B. Melody description and extraction in the context of music content processing. *Journal of New Music Research*, Taylor & Francis, v. 32, n. 1, p. 23–40, 2003. Citado na página 658.
- GONTZES, D. *SMART HOSPITALS – PART 1: DESIGNING THE FUTURE*. 2017. Disponível em: https://optimityadvisors.com/Smart_Hospitals_Part1). Acesso em: 11 abr. 2019. Citado na página 516.
- GONZALEZ, R.; WOODS, R.; EDDINS, S. Fundamentals of digital image processing. *prentice Hall, upper saddle river, NJ*, 2002. Citado na página 98.
- GONZALEZ, R. C.; WOODS, R. C. *Processamento digital de imagens* . [S.l.]: Pearson Educación, 2009. Citado na página 138.
- GONZALEZ, R. C.; WOODS, R. E. *Processamento de imagens digitais*. [S.l.]: Edgard Blucher, 2000. Citado 2 vezes nas páginas 234 e 235.
- GONZALEZ, R. C.; WOODS, R. E. Digital image processing second edition. *Beijing: Publishing House of Electronics Industry*, v. 455, 2002. Citado 2 vezes nas páginas 54 e 55.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016. Citado 5 vezes nas páginas 86, 100, 254, 269 e 283.
- GORBALENYA, A. E. et al. Severe acute respiratory syndrome-related coronavirus: The species and its viruses: a statement of the coronavirus study group. *bioRxiv*, 2020. Citado 3 vezes nas páginas 150, 151 e 182.
- GORENSTEIN, M. R. *Diversidade de espécies em comunidades arbóreas: aplicação de índices de distinção taxonômica em três formações florestais do Estado de São Paulo*. Tese (Doutorado) — Universidade de São Paulo, 2009. Citado 2 vezes nas páginas 56 e 57.
- GOTOH, O. An improved algorithm for matching biological sequences. *Journal of molecular biology*, Elsevier, v. 162, n. 3, p. 705–708, 1982. Citado na página 613.
- GOTTSCHALK, M. et al. Removing energy code smells with reengineering services. *GI-Jahrestagung*, v. 208, p. 441–455, 2012. Citado na página 335.
- GOUYON, F.; HERRERA, P. Determination of the meter of musical audio signals: Seeking recurrences in beat segment descriptors. In: AUDIO ENGINEERING SOCIETY. *Audio Engineering Society Convention 114*. [S.l.], 2003. Citado na página 655.
- GOZES, O. et al. *Rapid AI Development Cycle for the Coronavirus (COVID-19) Pandemic: Initial Results for Automated Detection & Patient Monitoring using Deep Learning CT Image Analysis*. 2020. Citado na página 150.
- GRECO, L.; RITROVATO, P.; XHAFA, F. An edge-stream computing infrastructure for real-time analysis of wearable sensors data. *Future Generation Computer Systems*, Elsevier, v. 93, p. 515–528, 2019. Citado 2 vezes nas páginas 521 e 522.
- GREENBERG, A. et al. The cost of a cloud: research problems in data center networks. *ACM SIGCOMM computer communication review*, ACM, v. 39, n. 1, p. 68–73, 2008. Citado na página 517.
- GREKOW, J. Audio features dedicated to the detection of arousal and valence in music recordings. In: IEEE. *INnovations in Intelligent SysTems and Applications (INISTA), 2017 IEEE International Conference on*. [S.l.], 2017. p. 40–44. Citado 3 vezes nas páginas 646, 650 e 652.
- GROOT, P. M. de et al. The epidemiology of lung cancer. *Translational lung cancer research*, AME Publications, v. 7, n. 3, p. 220, 2018. Citado na página 233.
- GROSCHE, P.; MÜLLER, M.; KURTH, F. Cyclic tempogram—a mid-level tempo representation for musicsignals. In: IEEE. *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. [S.l.], 2010. p. 5522–5525. Citado na página 573.

- GROTTKE, M. et al. Analysis of software aging in a web server. *IEEE Transactions on reliability*, IEEE, v. 55, n. 3, p. 411–420, 2006. Citado 3 vezes nas páginas [444](#), [450](#) e [451](#).
- GROTTKE, M.; MATIAS, R.; TRIVEDI, K. S. The fundamentals of software aging. In: IEEE. 2008 IEEE International conference on software reliability engineering workshops (ISSRE Wksp). [S.I.], 2008. p. 1–6. Citado 2 vezes nas páginas [444](#) e [446](#).
- GU, L.; ZENG, D.; GUO, S. Vehicular cloud computing: A survey. In: IEEE. 2013 IEEE Globecom Workshops (GC Wkshps). [S.I.], 2013. p. 403–407. Citado na página [422](#).
- GU, Y. et al. Automatic lung nodule detection using a 3d deep convolutional neural network combined with a multi-scale prediction strategy in chest cts. *Computers in biology and medicine*, Elsevier, v. 103, p. 220–231, 2018. Citado 2 vezes nas páginas [238](#) e [244](#).
- GUARAGNELLA, C.; RIZZI, M. Simple and accurate border detection algorithm for melanoma computer aided diagnosis. *Diagnostics*, v. 10, p. 1–7, 2020. Citado na página [216](#).
- GUERRA, R. L. et al. Custo-utilidade de terapias-alvo comparadas à dacarbazina para o tratamento de primeira linha do melanoma avançado não-cirúrgico e metastático no sistema único de saúde do brasil. *Value in Health Regional Issues*, Elsevier, v. 20, p. 103–109, 2019. Citado na página [137](#).
- GUILLÉN, M. A. et al. Performance evaluation of edge-computing platforms for the prediction of low temperatures in agriculture using deep learning. *The Journal of Supercomputing*, Springer, v. 77, n. 1, p. 818–840, 2021. Citado na página [313](#).
- GUIMARÃES, A. P.; MACIEL, P. R.; JR, R. M. An analytical modeling framework to evaluate converged networks through business-oriented metrics. *Reliability Engineering & System Safety*, Elsevier, v. 118, p. 81–92, 2013. Citado na página [436](#).
- GUNASEKARAN, S.; REVATHY, K. Automatic recognition and retrieval of wild animal vocalizations. *International Journal of Computer Theory and Engineering*, IACSIT Press, v. 3, n. 1, p. 136, 2011. Citado na página [659](#).
- GUNST, R. F. *Response surface methodology: process and product optimization using designed experiments*. [S.I.]: Taylor & Francis, 1996. Citado 3 vezes nas páginas [338](#), [366](#) e [402](#).
- GUO, Y. et al. Breast cancer histology image classification based on deep neural networks. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.I.], 2018. p. 827–836. Citado 3 vezes nas páginas [81](#), [93](#) e [263](#).
- GUO, Y.; ZHAO, G.; PIETIKÄINEN, M. Discriminative features for texture description. *Pattern Recognition*, Elsevier, v. 45, n. 10, p. 3834–3843, 2012. Citado na página [118](#).
- GUPTA, L.; JAIN, R.; CHAN, H. A. Mobile edge computing—an important ingredient of 5g networks. *IEEE Software Defined Networks Newsletter*, 2016. Citado na página [494](#).
- GUPTA, P. et al. Iot based smart healthcare kit. In: IEEE. 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT). [S.I.], 2016. p. 237–242. Citado 2 vezes nas páginas [400](#) e [412](#).
- GUTIÉRREZ, E. G. et al. *Tonal description of music audio signals*. [S.I.]: Citeseer, 2006. Citado 2 vezes nas páginas [657](#) e [658](#).
- HAFEMANN, L. G. An analysis of deep neural networks for texture classification. 2014. Citado na página [40](#).
- HAFEMANN, L. G. An analysis of deep neural networks for texture classification. 2014. Citado na página [40](#).
- HAGEN, J. B. The origins of bioinformatics. *Nature Reviews Genetics*, Nature Publishing Group, v. 1, n. 3, p. 231, 2000. Citado na página [607](#).

HAKLAY, M.; WEBER, P. Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 2008. Citado na página 542.

HALL, M. et al. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, ACM New York, NY, USA, v. 11, n. 1, p. 10–18, 2009. Citado 3 vezes nas páginas 62, 581 e 600.

HALL, M. A. Correlation-based feature selection of discrete and numeric class machine learning. University of Waikato, Department of Computer Science, 2000. Citado na página 140.

HAN, B.-J. et al. Music emotion classification and context-based music recommendation. *Multimedia Tools and Applications*, Springer, v. 47, n. 3, p. 433–460, 2010. Citado 5 vezes nas páginas 645, 652, 655, 656 e 657.

HANCOCK, M. C. *Pylidc - An object relational mapping for the LIDC dataset using SQLAlchemy*. 2016. [\(https://pylidc.github.io/\)](https://pylidc.github.io/). Citado na página 193.

HANLEY, J. A.; MCNEIL, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, v. 143, n. 1, p. 29–36, 1982. Citado 2 vezes nas páginas 577 e 595.

HAO, S. et al. Estimating mobile application energy consumption using program analysis. In: IEEE. *Software Engineering (ICSE), 2013 35th International Conference on*. [S.l.], 2013. p. 92–101. Citado 2 vezes nas páginas 334 e 341.

HAO, S. et al. Puma: Programmable ui-automation for large-scale dynamic analysis of mobile apps. In: ACM. *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. [S.l.], 2014. p. 204–217. Citado 2 vezes nas páginas 334 e 341.

HARALICK, R. M. et al. Statistical and structural approaches to texture. *Proceedings of the IEEE*, v. 67, n. 5, p. 786–804, 1979. Citado na página 11.

HARALICK, R. M. et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, v. 3, n. 6, p. 610–621, 1973. Citado na página 55.

HARALICK, R. M.; SHANMUGAM, K.; DINSTEIN, I. H. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, Ieee, n. 6, p. 610–621, 1973. Citado na página 184.

Harangi, B.; Antal, B.; Hajdu, A. Automatic exudate detection with improved nave-bayes classifier. In: 2012 25th IEEE International Symposium on Computer-Based Medical Systems (CBMS). [S.l.: s.n.], 2012. p. 1–4. ISSN 1063-7125. Citado 2 vezes nas páginas 205 e 213.

HARDISON, R. C. Comparative genomics. *PLoS biology*, Public Library of Science, v. 1, n. 2, p. e58, 2003. Citado na página 610.

HARTE, C.; SANDLER, M.; GASSEN, M. Detecting harmonic change in musical audio. In: ACM. *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*. [S.l.], 2006. p. 21–26. Citado 3 vezes nas páginas 573, 575 e 658.

HARVEY, P. H.; RAMBAUT, A. Comparative analyses for adaptive radiations. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, The Royal Society, v. 355, n. 1403, p. 1599–1605, 2000. Citado na página 139.

HATANAKA, Y. et al. Vertical cup-to-disc ratio measurement for diagnosis of glaucoma on fundus images. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Medical Imaging 2010: Computer-Aided Diagnosis*. [S.l.], 2010. v. 7624, p. 76243C. Citado na página 59.

HE, K. et al. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. Citado na página 29.

HE, K. et al. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. Citado na página 176.

HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778. Citado 5 vezes nas páginas 101, 252, 267, 284 e 285.

HE, X. et al. Sample-efficient deep learning for covid-19 diagnosis based on ct scans. *medRxiv*, Cold Spring Harbor Laboratory Press, 2020. Disponível em: <<https://www.medrxiv.org/content/early/2020/04/17/2020.04.13.20063941>>. Citado 3 vezes nas páginas 150, 154 e 160.

HE, X. et al. Sample-efficient deep learning for covid-19 diagnosis based on ct scans. *medRxiv*, Cold Spring Harbor Laboratory Press, 2020. Citado 3 vezes nas páginas 191, 192 e 201.

HEATH, M. et al. The digital database for screening mammography. In: MEDICAL PHYSICS PUBLISHING. *Proceedings of the 5th international workshop on digital mammography*. [S.l.], 2000. p. 212–218. Citado 3 vezes nas páginas 82, 128 e 129.

HELABS. Deep learning é o motor que propulsiona a inteligência artificial. URL: <https://helabs.com/blog/deep-learning-propulsiona-a-inteligencia-artificial/> (visited on 14/05/2019), 2019. Citado 3 vezes nas páginas 84, 250 e 264.

HELMUS, M. et al. Phylogenetic measures of biodiversity. *The American Naturalist*, [The University of Chicago Press, The American Society of Naturalists], v. 169, n. 3, p. E68–E83, 2007. ISSN 00030147, 15375323. Citado 3 vezes nas páginas 184, 185 e 186.

HENDRICK, A. M.; GIBSON, M. V.; KULSHRESHTHA, A. Diabetic retinopathy. *Primary Care: Clinics in Office Practice*, v. 42, n. 3, p. 451 – 464, 2015. ISSN 0095-4543. Primary Care Ophthalmology. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S009545431500038X>>. Citado na página 204.

HENGARTNER, U. et al. Detection and analysis of routing loops in packet traces. In: ACM. *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*. [S.l.], 2002. p. 107–112. Citado na página 362.

HEVNER, K. Experimental studies of the elements of expression in music. *The American Journal of Psychology*, JSTOR, v. 48, n. 2, p. 246–268, 1936. Citado 2 vezes nas páginas 636 e 637.

HOFFMAN, F.; GARDNER, R. *Evaluation of Uncertainties in Environmental Radiological Assessment Models*. [S.l.]: Radiological Assessments, 1983. Citado 3 vezes nas páginas 312, 425 e 520.

HOLTZ, Y.; DAVID, J. L.; RANWEZ, V. The genetic map comparator: a user-friendly application to display and compare genetic maps. *Bioinformatics*, v. 33, n. 9, p. 1387–1388, 2017. Citado na página 609.

HOLUBČÍK, M. et al. A model approach for the formation of synergy effects in the automotive industry with big data solutions: Application for distribution and transport service strategy. In: *Smart Technology Trends in Industrial and Business Management*. [S.l.]: Springer, 2019. p. 467–488. Citado na página 517.

HORNI, K. N. A.; AXHAUSEN, K. W. *The Multi-Agent Transport Simulation MATSim*. [S.l.: s.n.], 2019. 10–22 p. Citado 2 vezes nas páginas 544 e 545.

HOSPITAL DE CÂNCER DE BARRETOS. *Câncer de Pulmão*. 2016. Disponível em: <<https://www.hcancerbarretos.com.br/home-paciente/77-paciente/tipos-de-cancer/171-cancer-de-pulmao>>. Acesso em: 09 out. 2019. Citado na página 234.

HOSSEINZADEH, D.; KRISHNAN, S. Combining vocal source and mfcc features for enhanced speaker recognition performance using gmm's. In: IEEE. *Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on*. [S.l.], 2007. p. 365–368. Citado na página 655.

HSIAO, S.-W.; CHEN, S.-K.; LEE, C.-H. Methodology for stage lighting control based on music emotions. *Information Sciences*, Elsevier, v. 412, p. 14–35, 2017. Citado 2 vezes nas páginas 646 e 652.

HSIEH, Y.-C. et al. Managed edge computing on internet-of-things devices for smart city applications. In: IEEE. *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. [S.l.], 2018. p. 1–2. Citado na página 400.

HU, W. et al. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, Hindawi, v. 2015, 2015. Citado na página 253.

HU, X.; DOWNIE, J. S. Exploring mood metadata: Relationships with genre, artist and usage metadata. In: *ISMIR*. [S.l.: s.n.], 2007. p. 67–72. Citado na página 648.

HU, X.; YANG, Y.-H. Cross-dataset and cross-cultural music mood prediction: A case on western and chinese pop songs. *IEEE Transactions on Affective Computing*, IEEE, v. 8, n. 2, p. 228–240, 2017. Citado 6 vezes nas páginas 646, 650, 652, 654, 657 e 658.

HU, Y. C. et al. Mobile edge computing a key technology towards 5g. ISBH, v. 1, 2015. Citado na página 401.

HU, Y. C. et al. Mobile edge computing—a key technology towards 5g. *ETSI white paper*, v. 11, n. 11, p. 1–16, 2015. Citado na página 492.

HUANG, D. et al. Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, IEEE, v. 41, n. 6, p. 765–781, 2011. Citado na página 270.

HUANG, D. et al. Mobicloud: building secure cloud framework for mobile computing and communication. In: IEEE. *2010 fifth IEEE international symposium on service oriented system engineering*. [S.l.], 2010. p. 27–34. Citado na página 516.

HUANG, G. et al. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2017. p. 4700–4708. Citado na página 30.

HUANG, G. et al. Densely connected convolutional networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2017. p. 4700–4708. Citado na página 176.

HUANG, G.; LIU, Z.; WEINBERGER, K. Q. Densely connected convolutional networks. CoRR, 2016. Citado na página 13.

Huang, P. et al. A classification system of lung nodules in ct images based on fractional brownian motion model. In: *2013 International Conference on System Science and Engineering (ICSSE)*. [S.l.: s.n.], 2013. p. 37–40. ISSN 2325-0925. Citado na página 65.

HUANG, Y. et al. Software rejuvenation: Analysis, module and applications. In: IEEE. *Twenty-fifth international symposium on fault-tolerant computing. Digest of papers*. [S.l.], 1995. p. 381–390. Citado 2 vezes nas páginas 444 e 446.

HUNDT, A. *Keras-tensorflow implementation of Fully Convolutional Networks for Semantic Segmentation*. [S.l.]: Github, 2017. (<https://github.com/aurora95/Keras-FCN>). Citado na página 300.

HUSSAIN, R. et al. Privacy-aware route tracing and revocation games in vanet-based clouds. In: IEEE. *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. [S.l.], 2013. p. 730–735. Citado na página 422.

HUSSAIN, R.; HUSSAIN, F.; ZEADALLY, S. Integration of vanet and 5g security: A review of design and implementation issues. *Future Generation Computer Systems*, Elsevier, v. 101, p. 843–864, 2019. Citado na página 422.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. [S.l.]: OTexts, 2018. Citado na página 448.

IAZZETTA, F. *Tutoriais de áudio e acústica*. [S.l.]: São Paulo: Departamento de Música da ECA-USP. Disponível em <http://www.eca...>, 2010. Citado 2 vezes nas páginas 571 e 572.

IBRAHIM, R.; ZIN, Z. M. Study of automated face recognition system for office door access control application. In: *2011 IEEE 3rd International Conference on Communication Software and Networks*. [S.l.: s.n.], 2011. p. 132–136. Citado 5 vezes nas páginas 690, 697, 698, 699 e 710.

IEEE. Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments. *IEEE Std 802.11p-2010*, p. 1–51, 2010. Citado 2 vezes nas páginas 427 e 486.

IEEE. Ieee international symposium on biomedical imaging. IEEE Pulse, 2018. Citado na página 44.

III, S. A. et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, Wiley Online Library, v. 38, n. 2, p. 915–931, 2011. Citado 2 vezes nas páginas 25 e 240.

III, S. G. A. et al. The lung image database consortium (lidc) and image database resource initiative (idri): A completed reference database of lung nodules on ct scans. *Medical Physics*, v. 38, n. 2, p. 915–931, 2011. Citado 3 vezes nas páginas 67, 70 e 71.

III, S. G. A. et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, Wiley Online Library, v. 38, n. 2, p. 915–931, 2011. Citado na página 193.

INCA. Abordagens basicas para o controle do cancer. 2011. Citado na página 262.

INCA. Instituto Nacional do Câncer Brasil. Estimativa 2018 Incidência do Câncer no Brasil. 2018. Citado 2 vezes nas páginas 80 e 262.

INCA. Instituto Nacional do Câncer. 2018. (<http://www2.inca.gov.br/wps/wcm/connect/inca/portal/home/>). [Online; accessed 19-November-2017]. Citado 3 vezes nas páginas 135, 136 e 137.

INCA. Câncer de Pulmão. 2019. Disponível em: (<https://www.inca.gov.br/tipos-de-cancer/cancer-de-pulmao>). Acesso em: 10 out. 2019. Citado na página 233.

INCA. Câncer de pulmão. 2019. Disponível em: (<https://www.inca.gov.br/tipos-de-cancer/cancer-de-pulmao>). Citado na página 68.

INCA, I. N. d. C. Câncer de pele melanoma. 2020. Disponível em: (<https://www.inca.gov.br/tipos-de-cancer/cancer-de-pele-melanoma>). Acesso em: 22.08.2020. Citado na página 216.

INCA, I. N. do C. Câncer de mama. 2019. Disponível em: (<https://www.inca.gov.br/tipos-de-cancer/cancer-de-mama>). Acessado em: 23 de jun. 2019. Citado 4 vezes nas páginas 96, 97, 116 e 120.

INSTITUTE, N. H. G. R. Comparative Genomics. 2015. Disponível em: (<https://www.genome.gov/11509542/comparative-genomics-fact-sheet/#al-3>). Acesso em: 11 ago. 2020. Citado na página 610.

INTEL, D. Data Plane development kit. 2014. Citado na página 361.

INTROINI, A.; PRESTI, G.; BOCCIGNONE, G. Audio features affected by music expressiveness: Experimental setup and preliminary results on tuba players. In: ACM. *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. [S.l.], 2016. p. 757–760. Citado na página 659.

IOFFE, S.; SZEGEDY, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015. (<https://arxiv.org/abs/1502.03167>). Citado na página 210.

ISIC. COLIVAC. 2016. (<https://challenge.kitware.com/#challenge/560d7856cad3a57cfde481ba>). [Online; accessed 19-November-2018]. Citado na página 143.

ISMAEL, P. M.; IBRAHIM, H. Y.; AL-KHALIL, A. B. A real time parking reservation system based on vehicular cloud computing. In: IEEE. *2020 International Conference on Computer Science and Software Engineering (CSASE)*. [S.l.], 2020. p. 26–31. Citado na página 422.

JAIN, R. *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. [S.l.]: John Wiley & Sons, 1990. Citado 3 vezes nas páginas 316, 317 e 410.

JAISWAL, A. et al. Classification of the covid-19 infected patients using densenet201 based deep transfer learning. *Journal of Biomolecular Structure and Dynamics*, Taylor & Francis, p. 1–8, 2020. Citado 4 vezes nas páginas 182, 191, 192 e 201.

JAISWAL, A. K. et al. Semi-supervised learning for cancer detection of lymph node metastases. *arXiv preprint arXiv:1906.09587*, 2019. Citado na página 249.

- JAISWAL, S. et al. Measurement and classification of out-of-sequence packets in a tier-1 ip backbone. *IEEE/ACM Transactions on Networking (ToN)*, IEEE Press, v. 15, n. 1, p. 54–66, 2007. Citado na página 362.
- JANTSCH, G. E.; FELIPUSSI, S. C. Estudo de técnicas de pseudocoloração aplicadas a imagens radiográficas—uma abordagem aprimorada. *Scientia Plena*, v. 8, n. 11 (b), 2013. Citado na página 219.
- JARARWEH, Y. et al. The future of mobile cloud computing: integrating cloudlets and mobile edge computing. In: IEEE. 2016 23rd International conference on telecommunications (ICT). [S.l.], 2016. p. 1–5. Citado 6 vezes nas páginas 310, 313, 405, 490, 492 e 495.
- JARARWEH, Y. et al. Sdmec: Software defined system for mobile edge computing. In: IEEE. 2016 IEEE International Conference on Cloud Engineering Workshop (IC2EW). [S.l.], 2016. p. 88–93. Citado 3 vezes nas páginas 310, 404 e 490.
- JIAN, S. et al. Secure real-time traffic data aggregation with batch verification for vehicular cloud in vanets. *IEEE Transactions on Vehicular Technology*, IEEE, 2019. Citado na página 422.
- JIANG, D.-N. et al. Music type classification by spectral contrast feature. In: IEEE. *Multimedia and Expo, 2002. ICME'02. Proceedings. 2002 IEEE International Conference on*. [S.l.], 2002. v. 1, p. 113–116. Citado 3 vezes nas páginas 573, 574 e 655.
- JIANG, L. et al. An iot-oriented data storage framework in cloud computing platform. *IEEE Transactions on Industrial Informatics*, IEEE, v. 10, n. 2, p. 1443–1451, 2014. Citado 2 vezes nas páginas 400 e 412.
- JIANG, X. et al. Hyperspectral image classification with capsnet and markov random fields. *IEEE Access*, IEEE, v. 8, p. 191956–191968, 2020. Citado na página 172.
- JIMÉNEZ-SÁNCHEZ, A.; ALBARQOUNI, S.; MATEUS, D. Capsule networks against medical imaging data challenges. In: *Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*. [S.l.]: Springer, 2018. p. 150–160. Citado na página 162.
- JOSHI, J.; JAIN, K.; AGARWAL, Y. Cvms: Cloud based vehicle monitoring system in vanets. In: IEEE. 2015 International Conference on Connected Vehicles and Expo (ICCVE). [S.l.], 2015. p. 106–111. Citado na página 422.
- JOSHI, S.; KARULE, P. A review on exudates detection methods for diabetic retinopathy. *Biomedicine e Pharmacotherapy*, v. 97, p. 1454 – 1460, 2018. ISSN 0753-3322. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0753332217346267>>. Citado na página 204.
- JR, G. B. *Classificação de tecidos da mama a partir de imagens mamográficas em massa e não massa usando Índice de Diversidade de McIntosh e máquinas de vetores de suporte*. Tese (Dissertação de Mestrado. Programa de Pós-graduação em Engenharia de Eletricidade) — Universidade Federal do Maranhão, 2012. Citado na página 69.
- JR, G. B. *Detecção de regiões de massas em mamografias usando índices de diversidade, geoestatística e geometria côncava*. Tese (Tese de Doutorado. Programa de Pós-graduação em Engenharia de Eletricidade) — Universidade Federal do Maranhão, 2014. Citado na página 69.
- JU, J. A playable implementation of Fully Convolutional Networks with Keras. [S.l.]: Github, 2016. <<https://github.com/JihongJu/keras-fcn>>. Citado na página 300.
- JUNG, H. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper*. [S.l.], 2016. Disponível em: <<http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-indexvni/mobile-white-paper-cl1-520862.html>>. Citado 2 vezes nas páginas 404 e 489.
- JUNG, W. et al. Devscope: a nonintrusive and online power analysis tool for smartphone hardware components. In: ACM. *Proceedings of the eighth IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*. [S.l.], 2012. p. 353–362. Citado 3 vezes nas páginas 334, 341 e 342.
- JUNIOR, A. L. et al. Expolynomial modelling for supporting vanet infrastructure planning. In: IEEE. 2017 IEEE 22nd Pacific rim international symposium on dependable computing (PRDC). [S.l.], 2017. p. 86–91. Citado 6 vezes nas páginas 422, 426, 427, 468, 473 e 486.

JUNIOR, A. S. V. de C. et al. Automatic methods for diagnosis of glaucoma using texture descriptors based on phylogenetic diversity. *Computers & Electrical Engineering*, Elsevier, v. 71, p. 102–114, 2018. Citado na página 52.

JUNIOR, A. S. V. de C. et al. Automatic methods for diagnosis of glaucoma using texture descriptors based on phylogenetic diversity. *Computers & Electrical Engineering*, Elsevier, v. 71, p. 102–114, 2018. Citado 2 vezes nas páginas 59 e 63.

JUNIOR, A. S. V. de C. et al. Automatic methods for diagnosis of glaucoma using texture descriptors based on phylogenetic diversity. *Computers & Electrical Engineering*, Elsevier, v. 71, p. 102–114, 2018. Citado na página 162.

JUNIOR, G. de B. V. ESTATÍSTICA: Teste *t student*. 2017. Disponível em: http://www.cpaqv.org/estatistica/teste_t.pdf. Acesso em: 18 de outubro de 2017. Citado na página 336.

JÚNIOR, J. L. O. L.; SOARES, J. B. Desenvolvimento de metodologia para avaliação da adesividade agregado-ligante com o uso de processamento digital de imagem. *Transportes*, v. 27, n. 1, p. 82–95, 2019. Citado na página 218.

JÚNIOR, L.; FERREIRA, A. *Modelos para o planejamento de infraestruturas de comunicações veiculares*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2016. Citado 2 vezes nas páginas 468 e 472.

JUNIOR, N. F. d. A. et al. Ferramentas para comparação genômica. [sn], 2002. Citado na página 608.

JUSLIN, P. N.; SLOBODA, J. A. *Music and emotion: Theory and research*. [S.l.]: Oxford University Press, 2001. Citado na página 634.

JUSTUS, T.; HUTSLER, J. J. Fundamental issues in the evolutionary psychology of music: Assessing innateness and domain specificity. *Music Perception: An Interdisciplinary Journal*, University of California Press Journals, v. 23, n. 1, p. 1–27, 2005. Citado na página 636.

JÚNIOR, R. de S. M. *Identification of Availability and Performance Bottlenecks in Cloud Computing Systems: An Approach Based On Hierarchical Models and Sensitivity Analysis*. Tese (Doutorado) — Federal University of Pernambuco, Center for Informatics, Graduate in Computer Science, Recife, 2016. Citado 3 vezes nas páginas 312, 425 e 520.

KALEKAR, P. S. Time series forecasting using holt-winters exponential smoothing. *Kanwal Rekhi School of Information Technology*, v. 4329008, n. 13, 2004. Citado na página 448.

KALMEGH, S. Analysis of weka data mining algorithm reptree, simple cart and randomtree for classification of indian news. In: . [S.l.: s.n.], 2015. Citado na página 73.

KANADAM, K. P.; CHEREDDY, S. R. Mammogram classification using sparse-roi: A novel representation to arbitrary shaped masses. *Expert Systems with Applications*, Elsevier, v. 57, p. 204–213, 2016. Citado 3 vezes nas páginas 117, 119 e 132.

KANAKARAJ, M.; GUDDETI, R. M. R. Nlp based sentiment analysis on twitter data using ensemble classifiers. In: IEEE. 2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN). [S.l.], 2015. p. 1–5. Citado 2 vezes nas páginas 664 e 665.

KANG, K.; WANG, X. Fully convolutional neural networks for crowd segmentation. *arXiv preprint arXiv:1411.4464*, 2014. Citado na página 39.

KANWAL, N.; BOSTANCI, E. Comparative study of instance based learning and back propagation for classification problems. Lahore, 2016. Citado na página 222.

KANZ, C. et al. The embl nucleotide sequence database. *Nucleic Acids Research*, v. 33, n. suppl1, p. D29–D33, 2005. Citado na página 611.

KARAKOÇ, M.; VAROL, A. Visual and auditory analysis methods for speaker recognition in digital forensic. In: . [S.l.: s.n.], 2017. Citado na página 591.

KARPATHY, A. et al. Cs231n convolutional neural networks for visual recognition. *Neural networks*, v. 1, 2016. Citado na página 283.

KAUR, K. Modified info gain attribute eval feature selection algorithm to increase efficiency of classification algorithms in data mining. *Journal of the Gujarat Research Society*, v. 21, n. 10s, p. 199–209, 2019. Citado 2 vezes nas páginas 575 e 594.

KE, M. Y. Z.; S, Z. Q. L.; L, P. X. L. Energy-efficient offloading for mobile edge computing in 5g heterogeneous networks. v. 4, p. 5896–5907, 2016. Citado 3 vezes nas páginas 313, 405 e 495.

KELLER, J. M.; GRAY, M. R.; GIVENS, J. A. A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, IEEE, n. 4, p. 580–585, 1985. Citado na página 126.

KERTESZ, A.; PFLANZNER, T.; GYIMOTHY, T. A mobile iot device simulator for iot-fog-cloud systems. *Journal of Grid Computing*, Springer, p. 1–23, 2018. Citado na página 310.

KESSLER, E. J.; HANSEN, C.; SHEPARD, R. N. Tonal schemata in the perception of music in bali and in the west. *Music Perception: An Interdisciplinary Journal*, University of California Press Journals, v. 2, n. 2, p. 131–165, 1984. Citado na página 636.

KHAN, A. et al. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, Springer, v. 53, n. 8, p. 5455–5516, 2020. Citado na página 28.

KIDO, S.; HIRANO, Y.; HASHIMOTO, N. Detection and classification of lung abnormalities by use of convolutional neural network (cnn) and regions with cnn features (r-cnn). In: IEEE. 2018 International Workshop on Advanced Image Technology (IWAIT). [S.l.], 2018. p. 1–4. Citado 3 vezes nas páginas 238, 239 e 244.

KIM, H.-G. et al. Speech enhancement of noisy speech using log-spectral amplitude estimator and harmonic tunneling. In: CITESEER. Proc. International Workshop on Acoustic Echo and Noise Control. [S.l.], 2003. p. 119–122. Citado na página 655.

KIM, J.-H.; NA, J.-C. A study on one-way communication using pf_ring zc. In: IEEE. Advanced Communication Technology (ICACT), 2017 19th International Conference on. [S.l.], 2017. p. 301–304. Citado 2 vezes nas páginas 369 e 370.

KIM, S. L. J.; LEE, S. An evacuation route choice model based on multi-agent simulation in order to prepare tsunami disasters. *Transportmetrica B: Transport Dynamics*, 2017. Citado 2 vezes nas páginas 549 e 550.

KIM, Y. E. et al. Music emotion recognition: A state of the art review. In: CITESEER. Proc. ISMIR. [S.l.], 2010. p. 255–266. Citado na página 638.

KING, T. Packet sniffing in a switched environment. *SANS Institute, GESC practical*, v. 1, 2002. Citado na página 362.

KINGMAN, S. Glaucoma is second leading cause of blindness globally. *Bulletin of the World Health Organization*, SciELO Public Health, v. 82, p. 887–888, 2004. Citado 2 vezes nas páginas 51 e 206.

KINGMAN, S. Glaucoma is second leading cause of blindness globally. *Bulletin of the World Health Organization*, SciELO Public Health, v. 82, p. 887–888, 2004. Citado na página 162.

KIRILENKO, A. P.; MOLODTSOVA, T.; STEPCHENKOVA, S. O. People as sensors: Mass media and local temperature influence climate change discussion on twitter. *Global Environmental Change*, v. 30, p. 92 – 100, 2015. ISSN 0959-3780. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0959378014001952>>. Citado 2 vezes nas páginas 668 e 670.

KITANOV, S.; MONTEIRO, E.; JANEVSKI, T. 5g and the fog—survey of related technologies and research directions. In: IEEE. 2016 18th Mediterranean Electrotechnical Conference (MELECON). [S.l.], 2016. p. 1–6. Citado 2 vezes nas páginas 404 e 489.

KITCHENHAM, B. Procedure for undertaking systematic reviews. *Computer Science Department, Keele University (TRISE-0401) and National ICT Australia Ltd (0400011T. 1), Joint Technical Report*, 2004. Citado na página 639.

- KITCHENHAM, B. Procedures for performing systematic reviews. *Keele, UK, Keele Univ.*, v. 33, 08 2004. Citado na página 688.
- KLAPURI, A. P. Multiple fundamental frequency estimation based on harmonicity and spectral smoothness. *IEEE Transactions on Speech and Audio Processing*, IEEE, v. 11, n. 6, p. 804–816, 2003. Citado na página 656.
- KLEIJNEN, J. P. Sensitivity analysis and optimization in simulation: design of experiments and case studies. In: IEEE. *Winter Simulation Conference Proceedings*, 1995. [S.l.], 1995. p. 133–140. Citado na página 426.
- KNOB, A. H. Aplicação do processamento de imagens digitais para análise da anisotropia da massa de grãos. 2011. Citado na página 138.
- KOIZUMI, Y. et al. Toyadmos: A dataset of miniature-machine operating sounds for anomalous sound detection. In: IEEE. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. [S.l.], 2019. p. 313–317. Citado na página 572.
- KOSTA, S. et al. Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading. In: IEEE. *Infocom, 2012 Proceedings IEEE*. [S.l.], 2012. p. 945–953. Citado na página 335.
- KOSTEK, B.; PLEWA, M. Rough sets applied to mood of music recognition. In: IEEE. *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on*. [S.l.], 2016. p. 71–78. Citado na página 646.
- KOVALESKI, P. de A. *Implementação de Redes Neurais Profundas para Reconhecimento de ações em Vídeo*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2018. Citado 3 vezes nas páginas 84, 251 e 265.
- KOWSALYA, N. et al. An approach to extract optic-disc from retinal image using k-means clustering. In: *4th International Conference on Biosignals, Images and Instrumentation (ICBSII)*. [S.l.]: Publishing Press, 2018. Citado 2 vezes nas páginas 43 e 49.
- KRIDI, D. S.; CARVALHO, C. G. N. d.; GOMES, D. G. A predictive algorithm for mitigate swarming bees through proactive monitoring via wireless sensor networks. In: ACM. *Proceedings of the 11th ACM symposium on Performance evaluation of wireless ad hoc, sensor, & ubiquitous networks*. [S.l.], 2014. p. 41–47. Citado 2 vezes nas páginas 383 e 384.
- KRIDI, D. S.; CARVALHO, C. G. N. de; GOMES, D. G. Application of wireless sensor networks for beehive monitoring and in-hive thermal patterns detection. *Computers and Electronics in Agriculture*, Elsevier, v. 127, p. 221–235, 2016. Citado na página 388.
- KRISHNAMURTHY, S.; NARASIMHAN, G.; RENGASAMY, U. Three-dimensional lung nodule segmentation and shape variance analysis to detect lung cancer with reduced false positives. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, SAGE Publications Sage UK: London, England, v. 230, n. 1, p. 58–70, 2016. Citado na página 24.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2012. p. 1097–1105. Citado 4 vezes nas páginas 86, 252, 267 e 284.
- KROHLING, L. L.; PAULA, K. M. P. de; BEHLAU, M. Roc curve of the pediatric voice related quality-of-life survey (p-vrqol). *CEP, SciELO Brasil*, v. 29107, p. 372, 2016. Citado na página 223.
- KRUSE, R. et al. *Computational intelligence: a methodological introduction*. [S.l.]: Springer, 2016. Citado na página 57.
- KRUTSCH, R.; TENORIO, D. Histogram equalization: Technical report. *Microcontroller Solutions Group, Guadalajara*, 2011. Citado 2 vezes nas páginas 164 e 170.
- KRYVASHEYEU, Y. et al. Rapid assessment of disaster damage using social media activity. *Science advances*, American Association for the Advancement of Science, v. 2, n. 3, p. e1500779, 2016. Citado 5 vezes nas páginas 662, 663, 665, 668 e 670.
- KUHN, D. R.; REILLY, M. J. An investigation of the applicability of design of experiments to software testing. In: IEEE. *Software Engineering Workshop, 2002. Proceedings. 27th Annual NASA Goddard/IEEE*. [S.l.], 2002. p. 91–95. Citado 3 vezes nas páginas 338, 366 e 402.

- KUHNEN, A. Meio ambiente e vulnerabilidade a percepção ambiental de risco e o comportamento humano. *Geografia (Londrina)*, v. 18, n. 2, p. 37–52, 2009. Citado 2 vezes nas páginas 663 e 665.
- KULAKARNI, M. A.; SHARMILA, M. Efficient and dynamic routing protocol in vanet: A survey. In: . [S.l.: s.n.], 2015. Citado na página 470.
- KUMAR, A. Ml metrics – sensitivity vs specificity difference. URL: <https://vitalflux.com/ml-metrics-sensitivity-vs-specificity-difference/> (visited on 14/05/2019), 2018. Citado 3 vezes nas páginas 254, 270 e 271.
- KUMAR, K. et al. A survey of computation offloading for mobile systems. *Mobile Networks and Applications*, Springer, v. 18, n. 1, p. 129–140, 2013. Citado na página 335.
- KUMAR, M. et al. A de-ann inspired skin cancer detection approach using fuzzy c-means clustering. Springer, 2020. Citado 2 vezes nas páginas 224 e 230.
- KUMAR, P.; GOYAL, D. Performance analysis for audio streaming in cloud. *IOSR-JCE*, v. 16, p. 98–104, 2014. Citado na página 638.
- KUMAR, S. CNN vs CAPSULE NETWORKS. 2019. (<https://www.linkedin.com/pulse/cnn-vs-capsule-network-sonu-kumar/>). Acessado em 11-01-2021. Citado na página 166.
- KUO, W.; ZUO, M. J. *Optimal Reliability Modeling - Principles and Applications*. [S.l.]: Wiley, 2003. ISBN 047139761X. Citado na página 423.
- KUROSE, J. F.; ROSS, K. W. *Computer networking: a top-down approach: international edition*. [S.l.]: Pearson Higher Ed, 2013. Citado na página 360.
- KUSUMO, A.; RECKIEN, D.; VERPLANKE, J. Utilising volunteered geographic information to assess resident's flood evacuation shelters. case study: Jakarta. *Applied Geography*, v. 88, p. 174 – 185, 2017. ISSN 0143-6228. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0143622816305811>. Citado 2 vezes nas páginas 669 e 670.
- KVIESIS, A. et al. Application of wireless sensor networks in precision apiculture. In: *Proceedings of the 14th International Scientific Conference Engineering for Rural Development (ERDev), Jelgava, Latvia*. [S.l.: s.n.], 2015. Citado na página 388.
- LAFFITTE, P. et al. Assessing the performances of different neural network architectures for the detection of screams and shouts in public transportation. *Expert systems with applications*, Elsevier, v. 117, p. 29–41, 2019. Citado na página 589.
- LAMERE, P. Social tagging and music information retrieval. *Journal of new music research*, Taylor & Francis, v. 37, n. 2, p. 101–114, 2008. Citado na página 634.
- LANDIS, J. R.; KOCH, G. G. An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers. *Biometrics*, JSTOR, p. 363–374, 1977. Citado na página 288.
- LANDIS, J. R.; KOCH, G. G. The measurement of observer agreement for categorical data. *biometrics*, JSTOR, p. 159–174, 1977. Citado 4 vezes nas páginas 34, 90, 255 e 271.
- LANDIS, J. R.; KOCH, G. G. The measurement of observer agreement for categorical data. *Biometrics*, [Wiley, International Biometric Society], v. 33, n. 1, p. 159–174, 1977. ISSN 0006341X, 15410420. Disponível em: <http://www.jstor.org/stable/2529310>. Citado 6 vezes nas páginas 70, 168, 173, 174, 177 e 179.
- LANE, R. D.; CHUA, P. M.; DOLAN, R. J. Common effects of emotional valence, arousal and attention on neural activation during visual processing of pictures. *Neuropsychologia*, Elsevier, v. 37, n. 9, p. 989–997, 1999. Citado na página 638.
- LARSEN, E.; AARTS, R. M. *Audio bandwidth extension: application of psychoacoustics, signal processing and loudspeaker design*. [S.l.]: John Wiley & Sons, 2005. Citado na página 656.
- LARSEN, S.; LEE, B. Survey on system i/o hardware transactions and impact on latency, throughput, and other factors. In: *Advances in Computers*. [S.l.]: Elsevier, 2014. v. 92, p. 67–104. Citado na página 369.

- LARTILLOT, O. Mirtoolbox user's manual. *Finnish Centre of Excellence in Interdisciplinary Music Research*, 2011. Citado 4 vezes nas páginas 655, 656, 657 e 658.
- LARTILLOT, O. et al. Multi-feature modeling of pulse clarity: Design, validation and optimization. In: CITESEER. ISMIR. [S.l.], 2008. p. 521–526. Citado na página 657.
- LARTILLOT, O.; TOIVIAINEN, P. A matlab toolbox for musical feature extraction from audio. In: BORDEAUX. *International conference on digital audio effects*. [S.l.], 2007. p. 237–244. Citado 2 vezes nas páginas 650 e 658.
- LARTILLOT, O. M. 1.5—*Users Manual*. 2014. Citado na página 659.
- LAURETTI, C. R.; FILHO, A. L. Glaucomas. *Medicina (Ribeirão Preto. Online)*, 1997. Citado na página 294.
- LAZARUS, R. S. Progress on a cognitive-motivational-relational theory of emotion. *American psychologist*, American Psychological Association, v. 46, n. 8, p. 819, 1991. Citado na página 636.
- LAZIC, Z. R. *Design of experiments in chemical engineering: a practical guide*. [S.l.]: John Wiley & Sons, 2006. Citado 3 vezes nas páginas 338, 366 e 402.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436–444, 2015. ISSN 0028-0836. Citado 6 vezes nas páginas 39, 250, 264, 265, 296 e 593.
- LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, IEEE, v. 86, n. 11, p. 2278–2324, 1998. Citado 7 vezes nas páginas 26, 29, 99, 281, 593, 594 e 599.
- LECUN, Y. et al. Convolutional networks and applications in vision. In: ISCAS. [S.l.: s.n.], 2010. p. 253–256. Citado na página 40.
- LEDERLIN, M. et al. Management strategy of pulmonary nodule in 2013. *Diagnostic and interventional imaging*, Elsevier, v. 94, n. 11, p. 1081–1094, 2013. Citado na página 52.
- LEDERLIN, M. et al. Management strategy of pulmonary nodule in 2013. *Diagnostic and interventional imaging*, Elsevier, v. 94, n. 11, p. 1081–1094, 2013. Citado na página 96.
- LEE, J.; JOE, H.; KIM, H. Smart phone power model generation using use pattern analysis. In: IEEE. *Consumer Electronics (ICCE), 2012 IEEE International Conference on*. [S.l.], 2012. p. 412–413. Citado na página 334.
- LEE, S.; KOUZANI, A.; HU, E. Random forest based lung nodule classification aided by clustering. *Computerized Medical Imaging and Graphics*, v. 34, n. 7, p. 535 – 542, 2010. ISSN 0895-6111. Citado na página 66.
- LEE, S.; LEVANTI, K.; KIM, H. S. Network monitoring: Present and future. *Computer Networks*, Elsevier, v. 65, p. 84–98, 2014. Citado na página 362.
- LEITE, E. F.; ROSA, R. Análise do uso, ocupação e cobertura da terra na bacia hidrográfica do rio formiga, tocantins. *Observatorium: Revista Eletrônica de Geografia*, v. 4, n. 12, 2012. Citado na página 219.
- LEMAITRE, G.; GRIMAUT, N.; SUIED, C. Acoustics and psychoacoustics of sound scenes and events. In: _____. [S.l.: s.n.], 2018. p. 41–67. ISBN 978-3-319-63449-4. Citado na página 591.
- LENSCOPE. *Glaucoma: o que é, tipos, sintomas e tratamentos*. 2018. Disponível em: [\(https://lenscope.com.br/blog/glaucoma-o-que-e-tipos-sintomas-e-tratamentos/\)](https://lenscope.com.br/blog/glaucoma-o-que-e-tipos-sintomas-e-tratamentos/). Citado na página 8.
- LEVITIN, D. J. *The world in six songs: How the musical brain created human nature*. [S.l.]: Penguin, 2008. Citado na página 636.
- LEVITIN, G.; XING, L.; HUANG, H.-Z. Optimization of partial software rejuvenation policy. *Reliability Engineering & System Safety*, Elsevier, v. 188, p. 289–296, 2019. Citado na página 446.
- LEW, M. S. et al. Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, ACM, v. 2, n. 1, p. 1–19, 2006. Citado na página 639.

LI, B. et al. Modeling and qos analysis of the ieee 802.11p broadcast scheme in vehicular ad hoc networks. *Journal of Communications and Networks*, IEEE, v. 19, n. 2, p. 169–179, 2017. Citado 3 vezes nas páginas [426](#), [427](#) e [486](#).

LI, J. et al. A comparison of deep learning methods for environmental sound detection. In: IEEE. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2017. p. 126–130. Citado na página [595](#).

LI, J. W. G.; WU, J. S. J. Data processing delay optimization in mobile edge computing. v. 2018, 2018. Citado 5 vezes nas páginas [313](#), [405](#), [406](#), [493](#) e [495](#).

LI, L.; VAIDYANATHAN, K.; TRIVEDI, K. S. An approach for estimation of software aging in a web server. In: IEEE. *Proceedings International Symposium on Empirical Software Engineering*. [S.l.], 2002. p. 91–100. Citado 2 vezes nas páginas [450](#) e [451](#).

LI, L.; ZIC, J. Image matching algorithm based on feature-point and daisy descriptor. *journal of multimedia*, v. 9, n. 6, p. 829–834, 2014. Citado 2 vezes nas páginas [270](#) e [287](#).

LI, Y. et al. Mass classification in mammograms based on two-concentric masks and discriminating texton. *Pattern Recognition*, Elsevier, v. 60, p. 648–656, 2016. Citado 3 vezes nas páginas [118](#), [119](#) e [132](#).

LI, Y.; SHEN, L. Skin lesion analysis towards melanoma detection using deep learning network. *Sensors, Multidisciplinary Digital Publishing Institute*, v. 18, n. 2, p. 556, 2018. Citado 5 vezes nas páginas [142](#), [146](#), [223](#), [224](#) e [230](#).

LI, Z. et al. Efficacy of a deep learning system for detecting glaucomatous optic neuropathy based on color fundus photographs. *Ophthalmology*, Elsevier, v. 125, n. 8, p. 1199–1206, 2018. Citado 2 vezes nas páginas [9](#) e [10](#).

LI, Z.; WANG, C.; XU, R. Computation offloading to save energy on handheld devices: a partition scheme. In: ACM. *Proceedings of the 2001 international conference on Compilers, architecture, and synthesis for embedded systems*. [S.l.], 2001. p. 238–246. Citado na página [335](#).

LIANG, W.-Y.; LAI, P.-T. Design and implementation of a critical speed-based dvfs mechanism for the android operating system. In: IEEE. *Embedded and Multimedia Computing (EMC), 2010 5th International Conference on*. [S.l.], 2010. p. 1–6. Citado 2 vezes nas páginas [335](#) e [336](#).

LIDY, T. et al. Improving genre classification by combination of audio and symbolic descriptors using a transcription systems. In: ISMIR. [S.l.: s.n.], 2007. p. 61–66. Citado na página [657](#).

LIMA, J. A. O. V. *Kit investidor-decisões de investimento e de financiamento*. Dissertação (Mestrado) — FEUC, 2011. Citado na página [384](#).

LIMA, S. M. de et al. Detection and classification of masses in mammographic images in a multi-kernel approach. *Computer methods and programs in biomedicine*, Elsevier, v. 134, p. 11–29, 2016. Citado 3 vezes nas páginas [118](#), [119](#) e [132](#).

Lima, T. J. B. et al. Lung ct screening with 3d convolutional neural network architectures. In: *2020 IEEE 17th International Symposium on Biomedical Imaging Workshops*. [S.l.: s.n.], 2020. p. 1–4. Citado na página [600](#).

LIN, C. et al. Music emotion recognition based on two-level support vector classification. In: IEEE. *Machine Learning and Cybernetics (ICMLC), 2016 International Conference on*. [S.l.], 2016. v. 1, p. 375–389. Citado 6 vezes nas páginas [644](#), [652](#), [654](#), [656](#), [657](#) e [658](#).

LIN, J.-J.; HUANG, S.-C. The implementation of the visitor access control system for the senior citizen based on the lbp face recognition. In: *2017 International Conference on Fuzzy Theory and Its Applications (iFUZZY)*. [S.l.: s.n.], 2017. p. 1–6. Citado 3 vezes nas páginas [691](#), [705](#) e [710](#).

LIN, Y.-D. et al. Time-and-energy-aware computation offloading in handheld devices to coprocessors and clouds. *IEEE Systems Journal*, IEEE, v. 9, n. 2, p. 393–405, 2015. Citado 2 vezes nas páginas [335](#) e [336](#).

LIN, Y.-P.; YANG, Y.-H.; JUNG, T.-P. Fusion of electroencephalographic dynamics and musical contents for estimating emotional responses in music listening. *Frontiers in neuroscience*, Frontiers, v. 8, p. 94, 2014. Citado 5 vezes nas páginas 645, 652, 654, 658 e 659.

Lindsey, K. Guidelines for emergency resource planning for overhead transmission line asset owners. In: 2008 IEEE/PES Transmission and Distribution Conference and Exposition. [S.l.: s.n.], 2008. p. 1–5. ISSN 2160-8555. Citado na página 678.

LINMANS, J. et al. Sample efficient semantic segmentation using rotation equivariant convolutional networks. *arXiv preprint arXiv:1807.00583*, 2018. Citado na página 249.

LIOU, C.-Y. et al. Autoencoder for words. *Neurocomputing*, v. 139, p. 84 – 96, 2014. ISSN 0925-2312. Citado na página 67.

LISBOA, M. F. F. da S. et al. Modeling the availability of an e-health system integrated with edge, fog and cloud infrastructures. In: IEEE. 2018 IEEE Symposium on Computers and Communications (ISCC). [S.l.], 2018. p. 00416–00421. Citado 2 vezes nas páginas 328 e 534.

LITTLE, J. D. A proof for the queuing formula: $L = \lambda w$. *Operations research*, INFORMS, v. 9, n. 3, p. 383–387, 1961. Citado 4 vezes nas páginas 316, 408, 475 e 498.

LITTLE, J. D.; GRAVES, S. C. Little's law. In: *Building intuition*. [S.l.]: Springer, 2008. p. 81–100. Citado na página 525.

LIU, D.; LU, L.; ZHANG, H.-J. Automatic mood detection from acoustic music data. Johns Hopkins University, 2003. Citado na página 657.

LIU, J. et al. Delay-optimal computation task scheduling for mobile-edge computing systems. In: IEEE. 2016 IEEE International Symposium on Information Theory (ISIT). [S.l.], 2016. p. 1451–1455. Citado 2 vezes nas páginas 405 e 495.

LIU, J.; MENG, L. Integrating artificial bee colony algorithm and bp neural network for software aging prediction in iot environment. *IEEE Access*, IEEE, v. 7, p. 32941–32948, 2019. Citado 2 vezes nas páginas 444 e 446.

LIU, J.; TAN, X.; WANG, Y. Cssap: Software aging prediction for cloud services based on arima-lstm hybrid model. In: IEEE. 2019 IEEE International Conference on Web Services (ICWS). [S.l.], 2019. p. 283–290. Citado 2 vezes nas páginas 444 e 446.

LIU, K.; KANG, G. Multiview convolutional neural networks for lung nodule classification. *International Journal of Imaging Systems and Technology*, Wiley Online Library, v. 27, n. 1, p. 12–22, 2017. Citado na página 24.

LIU, M.-K.; TSENG, Y.-H.; TRAN, M.-Q. Tool wear monitoring and prediction based on sound signal. *The International Journal of Advanced Manufacturing Technology*, Springer, v. 103, n. 9-12, p. 3361–3373, 2019. Citado na página 573.

LIU, X. et al. Multi-view multi-scale cnns for lung nodule type classification from ct images. *Pattern Recognition*, v. 77, p. 262 – 275, 2018. ISSN 0031-3203. Disponível em: [\(<http://www.sciencedirect.com/science/article/pii/S0031320317305186>\)](http://www.sciencedirect.com/science/article/pii/S0031320317305186). Citado na página 150.

LIU, Y. Feature extraction and image recognition with convolutional neural networks. *Journal of Physics: Conference Series*, v. 1087, p. 062032, 09 2018. Citado na página 152.

LIU, Y. et al. Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1703.02442*, 2017. Citado 2 vezes nas páginas 248 e 249.

LOEWEN, N. A.; TANNA, A. P. Glaucoma risk factors: Intraocular pressure. In: _____. *Clinical Glaucoma Care: The Essentials*. New York, NY: Springer New York, 2014. p. 1–22. ISBN 978-1-4614-4172-4. Disponível em: [\(\[https://doi.org/10.1007/978-1-4614-4172-4_1\]\(https://doi.org/10.1007/978-1-4614-4172-4_1\)\)](https://doi.org/10.1007/978-1-4614-4172-4_1). Citado na página 163.

LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 3431–3440. Citado na página 41.

Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2015. p. 3431–3440. ISSN 1063-6919. Citado na página 41.

LOONEY, C. G. *Pattern recognition using neural networks: theory and algorithms for engineers and scientists*. [S.l.]: Oxford University Press, Inc., 1997. Citado 2 vezes nas páginas 57 e 594.

LOPEZ, A. R. et al. Skin lesion classification from dermoscopic images using deep learning techniques. In: IEEE. *2017 13th IASTED International Conference on Biomedical Engineering (BioMed)*. [S.l.], 2017. p. 49–54. Citado 4 vezes nas páginas 278, 279, 280 e 292.

LOSANGELESTIME. *California's Camp fire was the costliest global disaster last year, insurance report shows*. 2019. Disponível em: <https://lat.ms/345ddxl>. Citado na página 551.

LUJAK, M.; GIORDANI, S. Centrality measures for evacuation: Finding agile evacuation routes. *Future Generation Computer Systems*, 2018. Citado na página 550.

LYONS, R. G. *Understanding Digital Signal Processing*, 3/E. [S.l.]: Pearson Education India, 2011. Citado na página 655.

LÄMMEL, G.; FLÖTTERÖD, G. Towards system optimum: Finding optimal routing strategies in time dependent networks for large-scale evacuation problems. 2009. Citado 2 vezes nas páginas 548 e 550.

LÄMMEL, M. R. G.; NAGEL, K. Bottlenecks and congestion in evacuation scenarios: A microscopic evacuation simulation for large-scale disasters. 2008. Citado na página 552.

MA, J. et al. De-anonymizing social networks with random forest classifier. *IEEE Access*, IEEE, v. 6, p. 10139–10150, 2017. Citado na página 595.

MA, J. et al. De-anonymizing social networks with random forest classifier. *IEEE Access*, IEEE, v. 6, p. 10139–10150, 2018. Citado na página 58.

MACEACHREN, A. M. et al. Senseplace2: Geotwitter analytics support for situational awareness. In: IEEE. *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*. [S.l.], 2011. p. 181–190. Citado 2 vezes nas páginas 663 e 664.

MACH, P.; BECVAR, Z. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials*, IEEE, v. 19, n. 3, p. 1628–1656, 2017. Citado na página 493.

MACIEL, P. et al. Mercury: Performance and dependability evaluation of systems with exponential, expolynomial, and general distributions. In: IEEE. *2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC)*. [S.l.], 2017. p. 50–57. Citado na página 481.

MAGALHÃES, F. dos S. Utilização do fitoplâncton como instrumento de avaliação em programas de monitoramento nos ecossistemas aquáticos costeiros. estudo de caso: Laguna de araruama/rj. 2012. Citado na página 384.

MAGALHÃES, J. P.; SILVA, L. M. Prediction of performance anomalies in web-applications based-on software aging scenarios. In: IEEE. *2010 IEEE second international workshop on software aging and rejuvenation*. [S.l.], 2010. p. 1–7. Citado 2 vezes nas páginas 450 e 451.

MAGURRAN, A. E. *Measuring biological diversity*. [S.l.]: John Wiley & Sons, 2013. Citado 2 vezes nas páginas 55 e 139.

MAHBOD, A. et al. Skin lesion classification using hybrid deep neural networks. In: IEEE. *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2019. p. 1229–1233. Citado 2 vezes nas páginas 279 e 292.

MAHESHWARI, S.; KANHANGAD, V.; PACHORI, R. B. Cnn-based approach for glaucoma diagnosis using transfer learning and lbp-based data augmentation. *arXiv preprint arXiv:2002.08013*, 2020. Citado 4 vezes nas páginas 169, 170, 177 e 178.

MAHESHWARI, S.; PACHORI, R. B.; ACHARYA, U. R. Automated diagnosis of glaucoma using empirical wavelet transform and correntropy features extracted from fundus images. *IEEE journal of biomedical and health informatics*, IEEE, v. 21, n. 3, p. 803–813, 2017. Citado 2 vezes nas páginas 59 e 63.

MAIA, L. B. et al. Aprendizagem profunda aplicada ao diagnóstico de melanoma. Universidade Federal do Maranhão, 2019. Citado 4 vezes nas páginas 142, 146, 223 e 230.

MAIA, U. *Como a IoT está mudando os hospitais e o mercado de saúde*. 2017. Disponível em: <<https://docmanagement.com.br/03/02/2017/como-iot-esta-mudando-os-hospitais-e-o-mercado-de-saude/>>. Acesso em: 14 abr. 2019. Citado na página 514.

MAJTNER, T.; YILDIRIM-YAYILGAN, S.; HARDEBERG, J. Y. Combining deep learning and hand-crafted features for skin lesion classification. In: IEEE. *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*. [S.l.], 2016. p. 1–6. Citado 3 vezes nas páginas 279, 280 e 292.

MAKKI, J. Diversity of breast carcinoma: histological subtypes and clinical relevance. *Clinical Medicine Insights: Pathology*, SAGE Publications Sage UK: London, England, v. 8, p. CPath-S31563, 2015. Citado na página 262.

MALDONADO, M. et al. System for monitoring natural disasters using natural language processing in the social network twitter. In: IEEE. *2016 IEEE International Carnahan Conference on Security Technology (ICCST)*. [S.l.], 2016. p. 1–6. Citado na página 665.

MALHOTRA, P. et al. Long short term memory networks for anomaly detection in time series. In: PRESSES UNIVERSITAIRES DE LOUVAIN. *Proceedings*. [S.l.], 2015. v. 89. Citado na página 449.

MALIK, A.; PANDEY, B. Security analysis of discrete event based threat driven authentication approach in vanet using petri nets. *IJ Network Security*, v. 20, n. 4, p. 601–608, 2018. Citado 5 vezes nas páginas 422, 426, 427, 468 e 486.

MALLAT, S. G. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Ieee, n. 7, p. 674–693, 1989. Citado na página 11.

MALLICK, S. Histogram of oriented gradients. URL: <https://www.learnopencv.com/histogram-of-oriented-gradients/> (visited on 14/05/2019), 2016. Citado na página 269.

MANTECÓN, T. et al. Access control based on visual face recognition using depth spatiograms of local quantized patterns. In: *2015 IEEE International Conference on Consumer Electronics (ICCE)*. [S.l.: s.n.], 2015. p. 530–531. Citado 3 vezes nas páginas 691, 693 e 710.

MAO, Y.; ZHANG, J.; LETAIEF, K. B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices. *IEEE Journal on Selected Areas in Communications*, IEEE, v. 34, n. 12, p. 3590–3605, 2016. Citado 3 vezes nas páginas 313, 405 e 495.

MARCUZZO, L. d. C.; SANTOS, C. R. dos. Análise e comparação de técnicas e aceleradores de processamento de pacotes. *ERRC 2017*, p. 2, 2017. Citado 3 vezes nas páginas 369, 370 e 371.

MARGALEF, R. Diversity and stability: a practical proposal and a model of interdependence. 1969. Citado 3 vezes nas páginas 102, 124 e 221.

MARKOWITZ, V. M. et al. The integrated microbial genomes (img) system. *Nucleic Acids Research*, v. 34, n. suppl, p. D344–D348, 2006. Citado na página 611.

MAROTTA, M. A. et al. Managing mobile cloud computing considering objective and subjective perspectives. *Computer Networks*, Elsevier, v. 93, p. 531–542, 2015. Citado 2 vezes nas páginas 310 e 490.

MARSAN, A. *Modelling with generalized stochastic Petri nets*. [S.l.]: Wiley, 1995. (Wiley series in parallel computing). ISBN 9780471930594. Citado 4 vezes nas páginas 310, 424, 517 e 519.

MARSAN, M. A. et al. *Modelling with generalized stochastic Petri nets*. [S.l.]: John Wiley & Sons, Inc., 1994. Citado na página 402.

MARTÍN, Y.; LI, Z.; CUTTER, S. L. Leveraging twitter to gauge evacuation compliance: spatiotemporal analysis of hurricane matthew. *PLoS one*, Public Library of Science, v. 12, n. 7, p. e0181701, 2017. Citado 3 vezes nas páginas 663, 664 e 665.

MARTINS, E. S. P. R.; MAGALHÃES, A. R. A seca de 2012-2015 no nordeste e seus impactos. *Parcerias Estratégicas*, 2015. Citado na página 382.

MASCI, J. et al. Stacked convolutional auto-encoders for hierarchical feature extraction. In: HONKELA, T. et al. (Ed.). *Artificial Neural Networks and Machine Learning – ICANN 2011*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 52–59. ISBN 978-3-642-21735-7. Citado na página 152.

MATHEW, A.; AMUDHA, P.; SIVAKUMARI, S. Deep learning techniques: An overview. In: SPRINGER. *International Conference on Advanced Machine Learning Technologies and Applications*. [S.l.], 2020. p. 599–608. Citado na página 593.

MATOS, C. Diagnosis of breast cancer in images mammography through local features and invariants. 2017. Citado na página 70.

MATOS, R. et al. Sensitivity analysis of a hierarchical model of mobile cloud computing. *Simulation Modelling Practice and Theory*, Elsevier, v. 50, p. 151–164, 2015. Citado 2 vezes nas páginas 312 e 313.

MATSIM. 2019. Disponível em: <https://matsim.org/>. Citado na página 540.

MATTAR, M. R. M. Deslocamentos ambientais: O caso dos pequenos países insulares. 2011. Citado na página 540.

MATTOS, A. C. M.; VASCONCELLOS, H. Análise de sensibilidade. *Revista de administração de empresas*, Scielo Brasil, v. 29, n. 1, p. 85–91, 1989. Citado na página 384.

MAZALA, T. T. Perfil da mortalidade por câncer de mama em sete lagoas-mg no período de 2011-2015. *Revista Brasileira de Ciências da Vida*, v. 7, n. Especial, p. 38–42, 2019. Citado na página 116.

MCDERMOTT, J.; HAUSER, M. The origins of music: Innateness, uniqueness, and evolution. *Music Perception: An Interdisciplinary Journal*, University of California Press Journals, v. 23, n. 1, p. 29–59, 2005. Citado na página 636.

MCFEE, B. et al. librosa: Audio and music signal analysis in python. In: *Proceedings of the 14th python in science conference*. [S.l.: s.n.], 2015. v. 8, p. 18–25. Citado 2 vezes nas páginas 580 e 599.

MCHUGH, M. L. Interrater reliability: the kappa statistic. *Biochémia medica: Biochémia medica*, Medicinska naklada, v. 22, n. 3, p. 276–282, 2012. Citado na página 127.

MCINTOSH, R. P. An index of diversity and the relation of certain concepts to diversity. *Ecology*, Wiley Online Library, v. 48, n. 3, p. 392–404, 1967. Citado 3 vezes nas páginas 102, 125 e 221.

MCKAY, C.; FUJINAGA, I. jsymbolic: A feature extractor for midi files. In: ICMC. [S.l.: s.n.], 2006. Citado na página 650.

MCKAY, C.; FUJINAGA, I.; DEPALLE, P. jaudio: A feature extraction library. In: *Proceedings of the International Conference on Music Information Retrieval*. [S.l.: s.n.], 2005. p. 600–3. Citado 4 vezes nas páginas 650, 655, 656 e 657.

MEDHAT, F.; CHESMORE, D.; ROBINSON, J. Masked conditional neural networks for sound classification. *Applied Soft Computing*, Elsevier, v. 90, p. 106073, 2020. Citado 4 vezes nas páginas 590, 596, 597 e 603.

MEDINA, A. et al. Traffic matrix estimation: Existing techniques and new directions. In: ACM. *ACM SIGCOMM Computer Communication Review*. [S.l.], 2002. v. 32, n. 4, p. 161–174. Citado na página 362.

MEDSEG. COVID-19 CT segmentation dataset. 2020. <http://medicalsegmentation.com/covid19/>. Citado na página 193.

MEITALOVS, J. et al. Automatic microclimate controlled beehive observation system. In: *8th International Scientific Conference Engineering for Rural Development*. [S.l.: s.n.], 2009. p. 265–271. Citado 3 vezes nas páginas 382, 387 e 388.

MELL, P.; GRANCE, T. et al. The nist definition of cloud computing. Computer Security Division, Information Technology Laboratory, National ... , 2011. Citado na página 516.

MELLO, A. A. P.; SUSANNA, J. R.; ALMEIDA, H. G. Glaucoma, 4. ed. Rio de Janeiro: Cultura Medica, p. 512, 2016. Citado 2 vezes nas páginas 51 e 53.

MELO, C. et al. Investigation of software aging effects on the openstack cloud computing platform. *JSW*, v. 12, n. 2, p. 125–137, 2017. Citado 2 vezes nas páginas 450 e 451.

MELO, C. et al. Capacity-oriented availability model for resources estimation on private cloud infrastructure. In: IEEE. *2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC)*. [S.l.], 2017. p. 255–260. Citado 3 vezes nas páginas 428, 431 e 438.

MENDONÇA, J. et al. Assessing performance and energy consumption in mobile applications. In: IEEE. *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*. [S.l.], 2015. p. 74–79. Citado 2 vezes nas páginas 341 e 342.

MENDONCA, T. et al. Ph2: A public database for the analysis of dermoscopic images. *Dermoscopy image analysis*, CRC Press, 2015. Citado na página 224.

MENEGOLA, A. et al. Knowledge transfer for melanoma screening with deep learning. In: IEEE. *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. [S.l.], 2017. p. 297–300. Citado 2 vezes nas páginas 279 e 292.

MENG, H. et al. Software aging and rejuvenation in a j2ee application server. *Quality and Reliability Engineering International*, Wiley Online Library, v. 32, n. 1, p. 89–97, 2016. Citado 2 vezes nas páginas 450 e 451.

MILLER, C. et al. Bim-extracted energyplus model calibration for retrofit analysis of a historically listed building in switzerland. *Proceedings of SimBuild*, v. 2014, 2014. Citado 3 vezes nas páginas 338, 366 e 402.

MILLER, M. *Cloud computing: Web-based applications that change the way you work and collaborate online*. [S.l.]: Que publishing, 2008. Citado na página 517.

MILLET, G. P. et al. Altitude and covid-19: Friend or foe? a narrative review. *Physiological Reports*, Wiley Online Library, v. 8, n. 24, p. e14615, 2021. Citado na página 184.

MILLETARI, F.; NAVAB, N.; AHMADI, S.-A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: IEEE. *2016 fourth international conference on 3D vision (3DV)*. [S.l.], 2016. p. 565–571. Citado 2 vezes nas páginas 237 e 238.

MINITAB. Designing an experiment. In: . [s.n.], 2017. Disponível em: <<https://goo.gl/1hrtyz>>. Acesso em: 27.11.2017. Citado na página 385.

MINITAB. O que é ANOVA? 2017. Disponível em: <<https://support.minitab.com/pt-br/minitab/18/help-and-how-to/modeling-statistics/anova/supporting-topics/basics/what-is-anova/>>. Acesso em: 17 oct. 2017. Citado na página 337.

MINITAB. *Tipos de Teste T*. 2017. Disponível em: <<https://support.minitab.com/pt-br/minitab/18/help-and-how-to/statistics/basic-statistics/supporting-topics/tests-of-means/types-of-t-tests/>>. Acesso em: 17 de outubro de 2017. Citado na página 337.

MIRANDA, F. B. d. et al. Quantização vetorial utilizando códigos esféricos. [sn], 2015. Citado na página 220.

MIRANDA, G. H. B. *Método para processamento e análise computacional de imagens histopatológicas visando apoiar o diagnóstico de câncer de colo de útero*. Tese (Doutorado) — Universidade de São Paulo, 2011. Citado na página 262.

- MISHRA, A. Metrics to evaluate your machine learning algorithm. URL: <https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234/> (visited on 14/05/2019), 2018. Citado 2 vezes nas páginas 254 e 270.
- MISRA, H. et al. Spectral entropy based feature for robust asr. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. [S.l.: s.n.], 2004. Citado na página 655.
- MOFFAT, D. et al. An evaluation of audio feature extraction toolboxes. 2015. Citado na página 639.
- MOLLOY, M. K. On the integration of delay and throughput measures in distributed processing models., University of California, Los Angeles, 1981. Citado na página 401.
- MONIKA, P.; VENKATESAN, D. Di-ann clustering algorithm for pruning in mlp neural network. *Indian Journal of Science and Technology*, v. 8, n. 16, 2015. Citado na página 57.
- MONTEIRO, J. G. *História do Glaucoma*. Loures, Portugal: Théa Portugal, 2014. Citado na página 294.
- MORABITO, R. A performance evaluation of container technologies on internet of things devices. In: *IEEE Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*. [S.l.], 2016. p. 999–1000. Citado 2 vezes nas páginas 412 e 413.
- MORABITO, R. Virtualization on internet of things edge devices with container technologies: a performance evaluation. *IEEE Access*, IEEE, v. 5, p. 8835–8850, 2017. Citado na página 400.
- MORAES, L. Â. G. d. Utilização de redes em cápsulas para classificação de regiões promotoras em sequências de dna. 2018. Citado na página 166.
- MORENO, V. et al. Testing the capacity of off-the-shelf systems to store 10gbe traffic. *IEEE Communications Magazine*, IEEE, v. 53, n. 9, p. 118–125, 2015. Citado na página 360.
- MORENO, V. et al. Commodity packet capture engines: Tutorial, cookbook and applicability. *IEEE Communications Surveys & Tutorials*, IEEE, v. 17, n. 3, p. 1364–1390, 2015. Citado na página 362.
- MORI, K.; MATSUGU, M.; SUZUKI, T. Face recognition using svm fed with intermediate output of cnn for face detection. In: . [S.l.: s.n.], 2005. p. 410–413. Citado na página 150.
- MOURA, N. et al. Proposta de um descriptor híbrido para aprimoramento da identificação automática de melanoma. In: SBC. *17º Workshop de Informática Médica (WIM 2017)*. [S.l.], 2017. v. 17, n. 1/2017. Citado 2 vezes nas páginas 142 e 146.
- MTIBAA, A.; HARRAS, K. A.; FAHIM, A. Towards computational offloading in mobile device clouds. In: *IEEE. Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on*. [S.l.], 2013. v. 1, p. 331–338. Citado na página 335.
- MUELLER, L. A. et al. The sgn comparative map viewer. *Bioinformatics*, v. 24, n. 3, p. 422–423, 2008. Citado na página 609.
- MÜLLER, M.; EWERT, S. Chroma toolbox: Matlab implementations for extracting variants of chroma-based audio features. In: CITESEER. *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR), 2011. hal-00727791, version 2-22 Oct 2012*. [S.l.], 2011. Citado 2 vezes nas páginas 573 e 650.
- MULLER, M.; KURTH, F.; CLAUSEN, M. Chroma-based statistical audio features for audio matching. In: *IEEE. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005. [S.l.], 2005. p. 275–278. Citado na página 574.
- MUNIA, T. T. K. et al. Automatic diagnosis of melanoma using linear and nonlinear features from digital image. In: IEEE. *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*. [S.l.], 2017. p. 4281–4284. Citado na página 52.
- MURATA, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, IEEE, v. 77, n. 4, p. 541–580, 1989. Citado na página 401.

MURATA, T. Petri nets: Properties, analysis and applications. *Proc. IEEE*, v. 77, n. 4, p. 541–580, April 1989. Citado 2 vezes nas páginas 424 e 517.

MURPHY, F. E. et al. b+ wsn: Smart beehive with preliminary decision tree analysis for agriculture and honey bee health monitoring. *Computers and Electronics in Agriculture*, Elsevier, v. 124, 2016. Citado na página 388.

MURPHY, F. E. et al. Development of an heterogeneous wireless sensor network for instrumentation and analysis of beehives. In: *IEEE. Instrumentation and Measurement Technology Conference (I2MTC), 2015 IEEE International*. [S.l.], 2015. p. 346–351. Citado na página 382.

MURTHY, D.; GROSS, A. J. Social media processes in disasters: Implications of emergent technology use. *Social science research*, Elsevier, v. 63, p. 356–370, 2017. Citado 3 vezes nas páginas 669, 670 e 674.

MUSHTAQ, Z.; SU, S.-F. Environmental sound classification using a regularized deep convolutional neural network with data augmentation. *Applied Acoustics*, Elsevier, v. 167, p. 107389, 2020. Citado 7 vezes nas páginas 577, 578, 585, 590, 596, 597 e 603.

MUTLAG, A. A. et al. Enabling technologies for fog computing in healthcare iot systems. *Future Generation Computer Systems*, Elsevier, v. 90, p. 62–78, 2019. Citado na página 517.

NADKARNI, P. M.; OHNO-MACHADO, L.; CHAPMAN, W. W. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, v. 18, n. 5, p. 544–551, 09 2011. ISSN 1527-974X. Disponível em: <https://doi.org/10.1136/amiajnl-2011-000464>. Citado na página 665.

NAG, A.; NIKHILENDRA, J. N.; KALMATH, M. Iot based door access control using face recognition. In: *2018 3rd International Conference for Convergence in Technology (I2CT)*. [S.l.: s.n.], 2018. p. 1–3. Citado 4 vezes nas páginas 691, 693, 694 e 710.

NANDHINI, U.; TAMILSELVAN, L. Computational analytics of client awareness for mobile application offloading with cloud migration. *KSII Transactions on Internet & Information Systems*, v. 8, n. 11, 2014. Citado 2 vezes nas páginas 335 e 336.

NAQI, S. M.; SHARIF, M.; JAFFAR, A. Lung nodule detection and classification based on geometric fit in parametric form and deep learning. *Neural Computing and Applications*, Oct 2018. ISSN 1433-3058. Disponível em: <https://doi.org/10.1007/s00521-018-3773-x>. Citado 2 vezes nas páginas 67 e 77.

NARIN, A.; KAYA, C.; PAMUK, Z. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *arXiv preprint arXiv:2003.10849*, 2020. Citado 3 vezes nas páginas 191, 192 e 201.

NASCIMENTO, L. B. et al. Classificação de nódulos pulmonares em maligno e benigno utilizando os índices de diversidade de shannon e de simpson. Universidade Federal do Maranhão, 2012. Citado 4 vezes nas páginas 122, 124, 219 e 220.

NAVARRO, J. M. et al. Sound levels forecasting in an acoustic sensor network using a deep neural network. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 20, n. 3, p. 903, 2020. Citado na página 589.

NAVATHE, S. B.; KOGLNIK, A. M. The challenges of modeling biological information for genome databases. In: *Conceptual Modeling*. [S.l.]: Springer, 1999. p. 168–182. Citado na página 611.

NAVEED, T.; SIDDIQUI, I. S.; AHMED, S. Parallel needleman-wunsch algorithm for grid. In: *Proceedings of the PAK-US International Symposium on High Capacity Optical Networks and Enabling Technologies (HONET 2005), Islamabad, Pakistan*. [S.l.: s.n.], 2005. Citado na página 611.

NAWAZ, W. et al. Classification of breast cancer histology images using alexnet. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 869–876. Citado 4 vezes nas páginas 104, 105, 112 e 192.

NCBI. Welcome to NCBI. 2018. Disponível em: <<https://www.ncbi.nlm.nih.gov/>>. Acesso em: 11 ago. 2020. Citado na página 608.

NEEDLEMAN, S. B.; WUNSCH, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, Elsevier, v. 48, n. 3, p. 443–453, 1970. Citado 2 vezes nas páginas 608 e 612.

NETO, H. V. Processamento digital de imagens. *Brasport*, p. 55, 1999. Citado na página 206.

NETO, J. D. de O.; RICCIO, E. L. Desenvolvimento de um instrumento para mensurar a satisfação do usuário de sistemas de informações. *Revista de Administração da Universidade de São Paulo*, v. 38, n. 3, 2003. Citado na página 614.

NETTO, S. M. B. et al. Segmentação automática de nódulos pulmonares com growing neural gas e máquina de vetores de suporte. Universidade Federal do Maranhão, 2010. Citado na página 57.

NETTO, S. M. B. et al. Automatic segmentation of lung nodules with growing neural gas and support vector machine. *Computers in biology and medicine*, Elsevier, v. 42, n. 11, p. 1110–1121, 2012. Citado na página 182.

NEWSWEEK. *What Fires Are Still Burning In California? Update, Map, Containment Latest*. 2018. Disponível em: <http://bit.ly/33Vxh58>. Citado na página 553.

Ngo, T.; Huynh-The, T.; Kim, D. A novel vanets-based traffic light scheduling scheme for greener planet and safer road intersections. *IEEE Access*, v. 7, p. 22175–22185, 2019. Citado na página 486.

NGU, A. H. et al. Smartwatch-based iot fall detection application. *Open Journal of Internet Of Things (OJIOT)*, RonPub, v. 4, n. 1, p. 87–98, 2018. Citado na página 515.

NGUYEN-MEIDINE, L. T. et al. A comparison of cnn-based face and head detectors for real-time video surveillance applications. In: . [S.l.: s.n.], 2017. Citado na página 286.

NGUYEN, T. A.; KIM, D. S.; PARK, J. S. A Comprehensive Availability Modeling and Analysis of a Virtualized Servers System Using Stochastic Reward Nets. *The Scientific World Journal*, Seoul, Korea, v. 2014, p. 1–18, 2014. ISSN 2356-6140. Disponível em: <http://www.hindawi.com/journals/tswj/2014/165316/> <http://dx.doi.org/10.1155/2014/165316>. Citado na página 444.

NIBALI, A.; HE, Z.; WOLLERSHEIM, D. Pulmonary nodule classification with deep residual networks. *International journal of computer assisted radiology and surgery*, Springer, v. 12, n. 10, p. 1799–1808, 2017. Citado na página 24.

NIU, M. et al. Poster: Docker-based self-organizing iot services architecture for smarthome. In: ACM. *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. [S.l.], 2017. p. 153–153. Citado na página 400.

NIU, X.-X.; SUEN, C. Y. A novel hybrid cnn–svm classifier for recognizing handwritten digits. *Pattern Recognition*, v. 45, n. 4, p. 1318 – 1325, 2012. ISSN 0031-3203. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0031320311004006>. Citado na página 150.

NÓBREGA, R. V. M. da et al. Lung nodule classification via deep transfer learning in ct lung images. In: IEEE. *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*. [S.l.], 2018. p. 244–249. Citado 3 vezes nas páginas 238, 239 e 244.

NUGROHO, H. A. et al. Segmentation of optic disc on retinal fundus images using morphological reconstruction enhancement and active contour. In: IEEE. *Science in Information Technology (ICSTech), 2016 2nd International Conference on*. [S.l.], 2016. p. 362–366. Citado na página 53.

NUNES, F. L. Introdução ao processamento de imagens médicas para auxílio a diagnóstico—uma visão prática. *Livro das Jornadas de Atualizações em Informática*, p. 73–126, 2006. Citado na página 52.

NURIKA, O. et al. Real throughput measurements comparison between unoptimized and optimized network cards. *Science International*, v. 29, p. 87–91, 2016. Citado 2 vezes nas páginas 369 e 370.

OJALA, T.; PIETIKÄINEN, M.; HARWOOD, D. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, Elsevier, v. 29, n. 1, p. 51–59, 1996. Citado na página 25.

OJALA, T.; PIETIKÄINEN, M.; MÄENPÄÄ, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, IEEE, n. 7, p. 971–987, 2002. Citado na página [24](#).

OLIVEIRA, D. et al. Availability and energy consumption analysis of mobile cloud environments. In: IEEE. *2013 IEEE International Conference on Systems, Man, and Cybernetics*. [S.l.], 2013. p. 4086–4091. Citado 2 vezes nas páginas [312](#) e [313](#).

OLIVEIRA, D. et al. Advanced stochastic petri net modeling with the mercury scripting language. In: ACM. *Proceedings of the 11th EAI International Conference on Performance Evaluation Methodologies and Tools*. [S.l.], 2017. p. 192–197. Citado na página [491](#).

OLIVEIRA, F. S. S. *Classificação de tecidos da mama em massa e não-massa usando índice de diversidade taxonômico e máquina de vetores de suporte*. Tese (Dissertação de Mestrado. Curso de Pós-Graduação em Engenharia de Eletricidade) — Universidade Federal do Maranhão, 2013. Citado na página [68](#).

OLIVEIRA, F. S. S. d. et al. Classificação de tecidos da mama em massa e não-massa usando índice de diversidade taxonômico e máquina de vetores de suporte. Universidade Federal do Maranhão, 2013. Citado 3 vezes nas páginas [57](#), [138](#) e [139](#).

OLIVEIRA, F. S. S. de et al. Classification of breast regions as mass and non-mass based on digital mammograms using taxonomic indexes and svm. *Computers in biology and medicine*, Elsevier, v. 57, p. 42–53, 2015. Citado na página [96](#).

OLIVEIRA, F. S. S. de et al. Classification of breast regions as mass and non-mass based on digital mammograms using taxonomic indexes and svm. *Computers in Biology and Medicine*, v. 57, p. 42 – 53, 2015. ISSN 0010-4825. Citado na página [204](#).

OLIVEIRA, J. Z. d. Assimetria funcional dos hemisférios cerebrais na percepção de timbre, intensidade ou altura, em contexto musical. 2002. Citado na página [571](#).

OLIVEIRA, M. d.; CUNHA, J. A. Abelhas africanizadas apis mellifera scutellata lepeletier, 1836 (hymenoptera: Apidae: Apinae) exploram recursos na floresta amazônica. *Acta Amazônica*, SciELO Brasil, v. 35, n. 3, p. 389–394, 2005. Citado na página [383](#).

OLIVEIRA, P. S. *Relação da Composição Química com a Origem Geográfica e Atividade Antioxidante de Méis de Meliponíneos e Apis Mellifera Produzidos no Estado do Pará*. Dissertação (Mestrado) — Universidade Federal do Pará., Belém, 2010. Citado na página [383](#).

OLIVEIRA, R. B. Método de detecção e classificação de lesões de pele em imagens digitais a partir do modelo chan-vese e máquina de vetor de suporte. Universidade Estadual Paulista (UNESP), 2012. Citado na página [136](#).

OLIVEIRA, R. B. Método de detecção e classificação de lesões de pele em imagens digitais a partir do modelo chan-vese e máquina de vetor de suporte. 2012. Citado na página [216](#).

OLIVEIRA RENAN MAGALHÃES MONTENEGRO JUNIOR, S. V. José Egídio Paulo de. Retinopatia diabética. *Sociedade Brasileira de Diabetes: SBD*, 2018. Citado na página [39](#).

OMS. *World Health Organization: What is diabetes?* 2020. Acessado em 11 de agosto de 2020. Disponível em: [\(http://www.who.int/diabetes/en/\)](http://www.who.int/diabetes/en/). Citado na página [203](#).

ONUBR. *Relatório da ONU alerta para aumento dramático das perdas econômicas provocadas por desastres*. 2018. Disponível em: [\(\)](http://bit.ly/2XqCDD5). Citado na página [539](#).

OPENSTREETMAP. 2019. Disponível em: [\(\)](https://www.openstreetmap.org/about). Citado na página [542](#).

ORNA, G.; BENÍTEZ, D. S.; PÉREZ, N. A low-cost embedded facial recognition system for door access control using deep learning. In: *2020 IEEE ANDESCON*. [S.l.: s.n.], 2020. p. 1–6. Citado 4 vezes nas páginas [691](#), [698](#), [700](#) e [710](#).

ORSINI, G.; BADE, D.; LAMERSDORF, W. Computing at the mobile edge: Designing elastic android applications for computation offloading. In: IEEE. 2015 8th IFIP Wireless and Mobile Networking Conference (WMNC). [S.l.], 2015. p. 112–119. Citado 3 vezes nas páginas 310, 404 e 490.

ORTEGA, D. D. Pf_ring: Solución de software libre y código abierto para la captura de paquetes en redes de alta velocidad. *Revista Telem@tica*, v. 14, n. 2, p. 13–22, 2015. Citado 2 vezes nas páginas 361 e 364.

OSAREH, A. et al. Automated identification of diabetic retinal exudates in digital colour images. *British Journal of Ophthalmology*, BMJ Publishing Group Ltd, v. 87, n. 10, p. 1220–1223, 2003. ISSN 0007-1161. Disponível em: <https://bjm.bmjjournals.com/content/87/10/1220>. Citado 2 vezes nas páginas 205 e 213.

OSHIRO, T. M. *Uma abordagem para a construção de uma única árvore a partir de uma Random Forest para classificação de bases de expressão gênica*. Tese (Doutorado) — Universidade de São Paulo, 2013. Citado na página 140.

OSHIRO, T. M. *Uma abordagem para a construção de uma única árvore a partir de uma Random Forest para classificação de bases de expressão gênica*. Monografia (Mestrado) — USP, São Paulo, 2013. Citado na página 221.

OST, D.; FEIN, A. M.; FEINSILVER, S. H. The solitary pulmonary nodule. *New England Journal of Medicine*, v. 348, n. 25, p. 2535–2542, 2003. PMID: 12815140. Disponível em: <https://doi.org/10.1056/NEJMcp012290>. Citado na página 66.

OTHMAN, M. et al. Mobibyte: an application development model for mobile cloud computing. *Journal of Grid Computing*, Springer, v. 13, n. 4, p. 605–628, 2015. Citado na página 310.

OU, S.; YANG, K.; LIOTTA, A. An adaptive multi-constraint partitioning algorithm for offloading in pervasive systems. In: IEEE. *Pervasive Computing and Communications*, 2006. PerCom 2006. *Fourth Annual IEEE International Conference on*. [S.l.], 2006. p. 10–pp. Citado na página 336.

OUEIDA, S. et al. An edge computing based smart healthcare framework for resource management. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 18, n. 12, p. 4307, 2018. Citado 2 vezes nas páginas 520 e 522.

OUYANG, W. et al. DeepID-Net: multi-stage and deformable deep convolutional neural networks for object detection. 2014. Citado na página 150.

OVERMARS, L. et al. Civi: circular genome visualization with unique features to analyze sequence elements. *Bioinformatics*, v. 31, n. 17, p. 2867–2869, 2015. Citado na página 609.

OZKAYA, U.; OZTURK, S.; BARSTUGAN, M. Coronavirus (covid-19) classification using deep features fusion and ranking technique. *arXiv preprint arXiv:2004.03698*, 2020. Citado 2 vezes nas páginas 154 e 160.

OZKAYA, U.; OZTURK, S.; BARSTUGAN, M. Coronavirus (covid-19) classification using deep features fusion and ranking technique. *arXiv preprint arXiv:2004.03698*, 2020. Citado 4 vezes nas páginas 183, 191, 192 e 201.

PAKHOMOV, D. et al. Deep residual learning for instrument segmentation in robotic surgery. *Cornell University Library*, 2017. Citado na página 300.

PAL, S. K.; MITRA, S. Multilayer perceptron, fuzzy sets, and classification. *IEEE Transactions on neural networks*, IEEE, v. 3, n. 5, p. 683–697, 1992. Citado na página 126.

PALMER, M. *Hands-on networking fundamentals*. [S.l.]: Cengage learning, 2012. Citado na página 363.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, IEEE, v. 22, n. 10, p. 1345–1359, 2009. Citado na página 100.

PAN, X. Research and implementation of access control system based on rfid and fnn-face recognition. In: *2012 Second International Conference on Intelligent System Design and Engineering Application*. [S.l.: s.n.], 2012. p. 716–719. Citado 6 vezes nas páginas 690, 694, 695, 696, 709 e 710.

PANCHAL, G. et al. Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers. *International Journal of Computer Theory and Engineering*, v. 3, n. 2, p. 332–337, 2011. Citado na página 140.

PANTELOPOULOS, A.; BOURBAKIS, N. Spn-model based simulation of a wearable health monitoring system. *31st Annual International Conference of the IEEE EMBS*, 2009. Citado na página 521.

PAPADOPOULOS, A. Mathematics and music theory: From pythagoras to rameau. *The Mathematical Intelligencer*, Springer, v. 24, n. 1, p. 65–73, 2002. Citado na página 638.

PARADELA, E. M. P.; LOURENÇO, R. A.; VERAS, R. P. Validação da escala de depressão geriátrica em um ambulatório geral. *Revista de saúde pública*, SciELO Public Health, v. 39, p. 918–923, 2005. Citado na página 110.

PARAMKUSHAM, S. et al. Novel technique for the detection of abnormalities in mammograms using texture and geometric features. In: IEEE. *2015 International Conference on Microwave, Optical and Communication Engineering (ICMOCE)*. [S.l.], 2015. p. 150–153. Citado 3 vezes nas páginas 117, 119 e 132.

Park, G.; Cho, H.; Choi, M. A contrast enhancement method using dynamic range separate histogram equalization. *IEEE Transactions on Consumer Electronics*, v. 54, n. 4, p. 1981–1987, 2008. Citado na página 156.

PARK, H.; YOO, C. D. Cnn-based learnable gammatone filterbank and equal-loudness normalization for environmental sound classification. *IEEE Signal Processing Letters*, IEEE, v. 27, p. 411–415, 2020. Citado 4 vezes nas páginas 590, 596, 597 e 603.

PARK, K.; WILLINGER, W. *Self-similar network traffic and performance evaluation*. [S.l.]: John Wiley & Sons, Inc., 2000. Citado na página 362.

PASTORE, A. P. et al. Effects of the association of experimental neuroendocrine and exocrine obesity on tail blood pressure and glucose metabolism in wistar rats. *Jornal Brasileiro de Nefrologia*, SciELO Brasil, v. 32, n. 2, p. 195–200, 2010. Citado na página 336.

Pathak, Y.; Shukla, P. K.; Arya, K. V. Deep bidirectional classification model for covid-19 disease infected patients. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, p. 1–1, 2020. Citado 2 vezes nas páginas 155 e 160.

PATRICK, M. K. et al. Capsule networks—a survey. *Journal of King Saud University-Computer and Information Sciences*, Elsevier, 2019. Citado na página 168.

PEARCE, D. K. Short-term inflation expectations: Evidence from a monthly survey: note. *Journal of Money, Credit and Banking*, JSTOR, v. 19, n. 3, p. 388–395, 1987. Citado na página 448.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Citado 3 vezes nas páginas 15, 16 e 172.

PEDRINI, H.; SCHWARTZ, W. R. *Análise de imagens digitais: princípios, algoritmos e aplicações*. [S.l.]: Thomson Learning, 2008. Citado 2 vezes nas páginas 8 e 55.

PEDRINI, H.; SCHWARTZ, W. R. *Análise de imagens digitais: princípios, algoritmos e aplicações*. [S.l.]: Thomson Learning, 2008. Citado na página 184.

PEREIRA, A. dos S.; SEIXAS, F. R.; NETO, F. R. de A. Própolis: 100 anos de pesquisa e suas perspectivas futuras. *Quim. Nova*, v. 25, n. 2, p. 321–326, 2002. Citado na página 383.

PEREIRA, G. R. Desenvolvimento e uso de tecnologia utilizando realidade virtual e aumentada como forma de cooperação com o tratamento de problemas psíquicos, somáticos ou psicossomáticos. 2018. Citado na página 218.

PEREIRA, P. et al. Software rejuvenation in computer systems: An automatic forecasting approach based on time series. In: IEEE. *2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC)*. [S.l.], 2018. p. 1–8. Citado 2 vezes nas páginas 450 e 451.

- PÉRIER, R. C. et al. The eukaryotic promoter database (epd). *Nucleic Acids Research*, v. 28, n. 1, p. 302–303, 2000. Citado na página 611.
- PERLOVSKY, L. Musical emotions: Functions, origins, evolution. *Physics of life reviews*, Elsevier, v. 7, n. 1, p. 2–27, 2010. Citado na página 636.
- PERONA, P.; FINK, M. *The Full Images for Natural Knowledge Caltech Office DB*. California, 2013. Citado na página 39.
- PERRY, R. W.; LINDELL, M. K. Preparedness for emergency response: guidelines for the emergency planning process. *Disasters*, Wiley Online Library, v. 27, n. 4, p. 336–350, 2003. Citado na página 678.
- PETCHEY, O. L.; GASTON, K. J. Functional diversity (fd), species richness and community composition. *Ecology Letters*, Blackwell Science Ltd, v. 5, n. 3, p. 402–411, 2002. ISSN 1461-0248. Disponível em: <<http://dx.doi.org/10.1046/j.1461-0248.2002.00339.x>>. Citado na página 72.
- PHAN, S. et al. Evaluation of deep convolutional neural networks for glaucoma detection. *Japanese journal of ophthalmology*, Springer, v. 63, n. 3, p. 276–283, 2019. Citado 4 vezes nas páginas 169, 170, 177 e 178.
- PIANKA, E. *Evolutionary Ecology. 5a Edição*. [S.l.]: Editora Harper Collins College Publishers, 1994. Citado 2 vezes nas páginas 55 e 139.
- PIANOSI, F. et al. Sensitivity analysis of environmental models: A systematic review with practical workflow. *Environmental Modelling and Software*, v. 79, p. 214–232, 2016. Citado 3 vezes nas páginas 312, 425 e 520.
- PICHLMAIR, M.; KAYALI, F. Levels of sound: On the principles of interactivity in music video games. In: CITESEER. *DiGRA Conference*. [S.l.], 2007. Citado na página 572.
- PICZAK, K. J. Esc: Dataset for environmental sound classification. In: *Proceedings of the 23rd ACM international conference on Multimedia*. [S.l.: s.n.], 2015. p. 1015–1018. Citado 2 vezes nas páginas 578 e 579.
- PICZAK, K. J. ESC: Dataset for Environmental Sound Classification. In: *Proceedings of the 23rd Annual ACM Conference on Multimedia*. [S.l.]: ACM Press, 2015. p. 1015–1018. ISBN 978-1-4503-3459-4. Citado na página 597.
- PINHEIRO, P. O.; COLLORBERT, R. Recurrent convolutional neural networks for scene labeling. *31st International Conference on Machine Learning, Beijing, China*, 2014. Citado na página 300.
- PINHEIRO, T. et al. Performance and data traffic analysis of mobile cloud environments. In: IEEE. *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.], 2018. p. 4100–4105. Citado 2 vezes nas páginas 310 e 490.
- PINHEIRO, T. F. da S. et al. Performance prediction for supporting mobile applications' offloading. *The Journal of Supercomputing*, Springer, v. 74, n. 8, p. 4060–4103, 2018. Citado 2 vezes nas páginas 310 e 490.
- PIRES, A.; NETO, G. Compound local binary pattern para reconhecimento de expressões faciais. *Universidade Federal do Maranhão, SAE o Luis, Brasil*, 2015. Citado na página 123.
- PIRES, G. M. et al. Arquitetura para um sistema de diagnóstico auxiliado por computador. *Departamento de Informática, Universidade Federal da Paraíba-Brasil*, 2006. Citado 2 vezes nas páginas 121 e 218.
- PIZER, S. M. et al. Adaptive histogram equalization and its variations. *Comput. Vision Graph. Image Process.*, Academic Press Professional, Inc., USA, v. 39, n. 3, p. 355–368, set. 1987. ISSN 0734-189X. Disponível em: <[https://doi.org/10.1016/S0734-189X\(87\)80186-X](https://doi.org/10.1016/S0734-189X(87)80186-X)>. Citado na página 156.
- PIZER, S. M. et al. Contrast-limited adaptive histogram equalization: speed and effectiveness. In: IEEE COMPUTER SOCIETY. *[1990] Proceedings of the First Conference on Visualization in Biomedical Computing*. [S.l.], 1990. p. 337–338. Citado na página 219.
- PLATT, J. Sequential minimal optimization: A fast algorithm for training support vector machines. 1998. Citado 2 vezes nas páginas 57 e 126.

- POLINSKY, I. et al. nm-variant systems: Adversarial-resistant software rejuvenation for cloud-based web applications. In: *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*. [S.l.: s.n.], 2020. p. 235–246. Citado na página 444.
- PONTES, D. L. de B. *Câncer benigno e maligno*. 2013. Disponível em: <https://www.einstein.br/noticias/noticia/cancer-benigno-maligno>. Citado na página 80.
- PONTI et al. Como funciona o deep learning. 2018. Citado 6 vezes nas páginas 84, 85, 250, 252, 265 e 268.
- PONTI, M. A.; COSTA, G. B. P. da. Como funciona o deep learning. *arXiv preprint arXiv:1806.07908*, 2018. Citado na página 236.
- PORIA, S. et al. Fusing audio, visual and textual clues for sentiment analysis from multimodal content. *Neurocomputing*, Elsevier, v. 174, p. 50–59, 2016. Citado 2 vezes nas páginas 655 e 657.
- PORWAL, P. et al. Indian diabetic retinopathy image dataset (idrid): a database for diabetic retinopathy screening research. In: *Journal Data*. [S.l.]: Publishing Press, 2018. Citado na página 44.
- PRAMANA, M. D.; LESTYEÀ, A.; AMIRUDDIN, A. Development of a secure access control system based on two-factor authentication using face recognition and otp sms-token. In: *2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*. [S.l.: s.n.], 2020. p. 52–57. Citado 3 vezes nas páginas 691, 701 e 710.
- PRATES, R. F. de C. *Detecção de Placas Veiculares: Uma Abordagem Adaptativa para a Localização de Placas Utilizando Descritor HOG*. Tese (Doutorado) — Universidade Federal de Ouro Preto, 2014. Citado na página 269.
- PREMSANKAR, M. d. F. G.; TALEB, T. Edge computing for the internet of things: A case study. v. 5, p. 1275–1284, 2018. Citado 7 vezes nas páginas 317, 405, 406, 409, 493, 495 e 500.
- Prentašić, P.; Lončarić, S. Detection of exudates in fundus photographs using convolutional neural networks. In: *2015 9th International Symposium on Image and Signal Processing and Analysis (ISPA)*. [S.l.: s.n.], 2015. p. 188–192. ISSN 1845-5921. Citado 2 vezes nas páginas 205 e 213.
- PRENTAŠIĆ, P.; LONČARIĆ, S. Detection of exudates in fundus photographs using deep neural networks and anatomical landmark detection fusion. *Computer Methods and Programs in Biomedicine*, v. 137, p. 281 – 292, 2016. ISSN 0169-2607. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0169260716308033>. Citado 2 vezes nas páginas 205 e 213.
- PROGRAMMERSOUGHT. From inception, rexnext to xception to mobilenets, shufflenet, mobilenetv2, shufflenetv2. URL: <http://www.programmersought.com/article/4817156943/> (visited on 04/05/2019), 2017. Citado 2 vezes nas páginas 268 e 287.
- PROSDOCIMI, F. *Introdução à Bioinformática*. [S.l.: s.n.], 2007. Citado na página 611.
- PROVOST, F.; DOMINGOS, P. Well-trained pets: Improving probability estimation trees. Citeseer, 2000. Citado 2 vezes nas páginas 38 e 46.
- PURI, R.; JAIN, V. Barcode detection using opencv-python. *Science*, v. 4, n. 1, p. 97–99, 2019. Citado na página 143.
- QIAN, H.; ANDRESEN, D. Reducing mobile device energy consumption with computation offloading. In: *IEEE. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2015 16th IEEE/ACIS International Conference on*. [S.l.], 2015. p. 1–8. Citado 2 vezes nas páginas 341 e 342.
- Qin, Y. et al. Simultaneous accurate detection of pulmonary nodules and false positive reduction using 3d cnns. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.: s.n.], 2018. p. 1005–1009. ISSN 2379-190X. Citado 2 vezes nas páginas 67 e 77.
- QUIGLEY, H. A.; BROMAN, A. T. The number of people with glaucoma worldwide in 2010 and 2020. *British journal of ophthalmology*, BMJ Publishing Group Ltd, v. 90, n. 3, p. 262–267, 2006. Citado na página 51.

- R. EMELYANOV K., K. A. K. *RadIO library for data science research of CT images*. 2017. Citado na página 240.
- RABIDAS, R. et al. Benign-malignant mass classification in mammogram using edge weighted local texture features. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Medical Imaging 2016: Computer-Aided Diagnosis*. [S.l.], 2016. v. 9785, p. 97851X. Citado 3 vezes nas páginas 118, 119 e 132.
- RAFEGAS, I.; VANRELL, M. Color encoding in biologically-inspired convolutional neural networks. *Vision Research*, p. –, 2018. ISSN 0042-6989. Disponível em: (<https://www.sciencedirect.com/science/article/pii/S0042698918300592>). Citado na página 208.
- RAGHAVENDRA, U. et al. Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images. *Information Sciences*, Elsevier, v. 441, p. 41–49, 2018. Citado 2 vezes nas páginas 9 e 10.
- RAHMANI, A. M. et al. Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach. *Future Generation Computer Systems*, Elsevier, v. 78, p. 641–658, 2018. Citado 4 vezes nas páginas 496, 514, 521 e 522.
- RAHMANI, A.-M. et al. Smart e-health gateway: Bringing intelligence to internet-of-things based ubiquitous healthcare systems. In: IEEE. *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*. [S.l.], 2015. p. 826–834. Citado na página 514.
- RAJAN, D.; ZUCK, R.; POELLABAUER, C. Workload-aware dual-speed dynamic voltage scaling. In: IEEE. *Embedded and Real-Time Computing Systems and Applications, 2006. Proceedings. 12th IEEE International Conference on*. [S.l.], 2006. p. 251–256. Citado na página 335.
- RAJAPUT, G.; RESHMI, B.; SIDRAMAPPA, C. Automatic localization of fovea center using mathematical morphology in fundus images. *International Journal of Machine Intelligence*, v. 3, n. 4, 2011. Citado 2 vezes nas páginas 299 e 305.
- RAJMAN, M.; BESANÇON, R. Text mining: Natural language techniques and text mining applications. In: _____. *Data Mining and Reverse Engineering: Searching for semantics. IFIP TC2 WG2.6 IFIP Seventh Conference on Database Semantics (DS-7) 7–10 October 1997, Leysin, Switzerland*. Boston, MA: Springer US, 1998. p. 50–64. ISBN 978-0-387-35300-5. Disponível em: (https://doi.org/10.1007/978-0-387-35300-5_3). Citado na página 664.
- RAKHLIN, A. et al. Deep convolutional neural networks for breast cancer histology image analysis. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 737–744. Citado 7 vezes nas páginas 81, 93, 104, 105, 192, 263 e 274.
- RAMESWARI, R. et al. Automated access control system using face recognition. *Materials Today: Proceedings*, v. 45, p. 1251–1256, 2021. ISSN 2214-7853. International Conference on Advances in Materials Research - 2019. Disponível em: (<https://www.sciencedirect.com/science/article/pii/S2214785320333113>). Citado 4 vezes nas páginas 691, 706, 707 e 710.
- RASMUSSEN, C. E. Gaussian processes in machine learning. In: SPRINGER. *Summer School on Machine Learning*. [S.l.], 2003. p. 63–71. Citado 2 vezes nas páginas 14 e 16.
- RASTGOO, M. et al. Classification of melanoma lesions using sparse coded features and random forests. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Medical Imaging 2016: Computer-Aided Diagnosis*. [S.l.], 2016. v. 9785, p. 97850C. Citado 2 vezes nas páginas 142 e 146.
- RATHORE, M. M. et al. Urban planning and building smart cities based on the internet of things using big data analytics. *Computer Networks*, Elsevier, v. 101, p. 63–80, 2016. Citado 2 vezes nas páginas 570 e 589.
- RECOMMENDATION, I. 709-6, *Parameter Values for the HDTV Standards for Production and International Programme Exchange*. 2015. Citado na página 16.
- REIS, A. B. S. et al. Estudo de índices de diversidade como descritores de textura para classificação de imagens de faces. Universidade Federal do Maranhão, 2012. Citado 2 vezes nas páginas 105 e 227.

REIS ALEXANDRE SOARES CASTRO, M. N.; BURGOYNE, C. F.; CHAUHAN, B. C. Optic disc margin anatomy in patients with glaucoma and normal controls with spectral domain optical coherence tomography. *Ophthalmology - Journal - Elsevier*, v. 119, n. 4, p. 738–747, 2012. Citado na página 294.

REISIG, W. *Understanding petri nets: modeling techniques, analysis methods, case studies*. [S.l.]: Springer, 2013. Citado na página 314.

REMUZZI, A.; REMUZZI, G. Covid-19 and italy: what next? *The Lancet*, Elsevier, v. 395, n. 10231, p. 1225–1228, 2020/08/11 2020. Disponível em: [https://doi.org/10.1016/S0140-6736\(20\)30627-9](https://doi.org/10.1016/S0140-6736(20)30627-9). Citado na página 182.

REN, J.-M.; WU, M.-J.; JANG, J.-S. R. Automatic music mood classification based on timbre and modulation features. *IEEE Transactions on Affective Computing*, IEEE, n. 3, p. 236–246, 2015. Citado 2 vezes nas páginas 646 e 656.

REN, S. et al. Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2015. p. 91–99. Citado na página 212.

REN, X. et al. A novel image classification method with cnn-xgboost model. In: . [S.l.: s.n.], 2017. p. 378–390. ISBN 978-3-319-64184-3. Citado na página 152.

REZA, A. W.; ESWARAN, C.; HATI, S. Automatic tracing of optic disc and exudates from color fundus images using fixed and variable thresholds. *Journal of Medical Systems*, v. 33, n. 1, p. 73, May 2008. ISSN 1573-689X. Disponível em: <https://doi.org/10.1007/s10916-008-9166-4>. Citado 2 vezes nas páginas 205 e 213.

RHO, S.; HAN, B.-j.; HWANG, E. Svr-based music mood classification and context-based music recommendation. In: ACM. *Proceedings of the 17th ACM international conference on Multimedia*. [S.l.], 2009. p. 713–716. Citado 4 vezes nas páginas 645, 657, 658 e 659.

RICOTTA, C. A parametric diversity measure combining the relative abundances and taxonomic distinctiveness of species. *Diversity and Distributions*, Wiley Online Library, v. 10, n. 2, p. 143–146, 2004. Citado na página 56.

RIM, H. et al. Transparent method offloading for slim execution. In: IEEE. *Wireless Pervasive Computing, 2006 1st International Symposium on*. [S.l.], 2006. p. 1–6. Citado na página 336.

RIMONE, M. I. A. G. RIM-ONE R2. 2015. Disponível em: <http://medimrg.webs.ull.es/download/602/>. Acesso em: 07 set. 2019. Citado na página 10.

RIPLEY, B. D. *Pattern Recognition and Neural Networks*. [S.l.]: Cambridge University Press, 1996. Citado na página 153.

RIZZO, L. Netmap: a novel framework for fast packet i/o. In: *21st USENIX Security Symposium (USENIX Security 12)*. [S.l.: s.n.], 2012. p. 101–112. Citado na página 361.

RIZZO, L. Revisiting network i/o apis: the netmap framework. *Communications of the ACM*, ACM, v. 55, n. 3, p. 45–51, 2012. Citado na página 364.

RIZZO, L.; LANDI, M. Netmap: Memory mapped access to network devices. *ACM SIGCOMM Computer Communication Review*, ACM, v. 41, n. 4, p. 422–423, 2011. Citado na página 364.

ROCHA, P. et al. 10gbe network card performance evaluation: A strategy based on sensitivity analysis. In: IEEE. *2019 IEEE Latin-American Conference on Communications (LATINCOM)*. [S.l.], 2019. p. 1–6. Citado na página 431.

ROCHA, S. V. *Diferenciação do Padrão de malignidade e benignidade de massas em imagens de mamografias usando padrões locais binários, Geoestatística e Índice de diversidade*. Tese (Tese de Doutorado. Curso de Pós- Graduação em Engenharia de Eletricidade) — Universidade Federal do Maranhão, 2014. Citado na página 68.

ROCHA, S. V. d. et al. Texture analysis of masses in digitized mammograms using gleason and menhinick diversity indexes. *Revista Brasileira de Engenharia Biomédica*, SciELO Brasil, v. 30, n. 1, p. 27–34, 2014. Citado 3 vezes nas páginas 117, 119 e 132.

ROCHA, S. V. d. et al. Diferenciação do padrão de malignidade e benignidade de massas em imagens de mamografias usando padrões locais binários, geoestatística e índice de diversidade. Universidade Federal do Maranhão, 2014. Citado 3 vezes nas páginas 120, 121 e 218.

RODRIGUES, T. S. et al. Inflammasomes are activated in response to sars-cov-2 infection and are associated with covid-19 severity in patients. *Journal of Experimental Medicine*, The Rockefeller University Press, v. 218, n. 3, 2021. Citado na página 184.

ROLKE, D. et al. Large-scale monitoring of effects of clothianidin-dressed oilseed rape seeds on pollinating insects in northern germany: effects on honey bees (*apis mellifera*). *Ecotoxicology*, Springer, v. 25, n. 9, 2016. Citado 2 vezes nas páginas 381 e 382.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *International Conference on Medical image computing and computer-assisted intervention*. [S.l.], 2015. p. 234–241. Citado 4 vezes nas páginas 41, 42, 45 e 237.

RONNEBERGER PHILIPP FISCHER, T. B. O. U-net: Convolutional networks for biomedical image segmentation. In: Spring International Publishing Switzerland. [S.l.]: Publishing Press, 2015. Citado na página 46.

ROSEBROCK, A. *Deep Learning for Computer Vision with Python*, 1^a Edição. [S.l.]: PYIMAGESEARCH, 2017. Citado 2 vezes nas páginas 295 e 297.

ROSEBROCK, A. *Deep Learning for Computer Vision with Python: ImageNet Bundle*. [S.l.]: PyImageSearch, 2017. Citado 3 vezes nas páginas 85, 251 e 266.

ROSENFIELD, G. H.; FITZPATRICK-LINS, K. A coefficient of agreement as a measure of thematic classification accuracy. *Photogrammetric engineering and remote sensing*, v. 52, n. 2, p. 223–227, 1986. Citado na página 288.

ROTTENBERG, J. Mood and emotion in major depression. *Current Directions in Psychological Science*, SAGE Publications Sage CA: Los Angeles, CA, v. 14, n. 3, p. 167–170, 2005. Citado na página 636.

ROUHI, R. et al. Benign and malignant breast tumors classification based on region growing and cnn segmentation. *Expert Systems with Applications*, Elsevier, v. 42, n. 3, p. 990–1002, 2015. Citado 3 vezes nas páginas 117, 119 e 132.

RUSSELL, J. A. A circumplex model of affect. *Journal of personality and social psychology*, American Psychological Association, v. 39, n. 6, p. 1161, 1980. Citado na página 636.

RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3rd. ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2009. ISBN 0136042597, 9780136042594. Citado na página 153.

SAADATSERESHT, A. M. M.; TALEAI, M. Evacuation planning using multiobjective evolutionary optimization approach. *European Journal of Operational Research*, 2009. Citado 2 vezes nas páginas 548 e 550.

SABA, T. et al. Region extraction and classification of skin cancer: A heterogeneous framework of deep cnn features fusion and reduction. *Journal of medical systems*, Springer, v. 43, n. 9, p. 289, 2019. Citado 3 vezes nas páginas 223, 224 e 230.

SABOUR, S.; FROSST, N.; HINTON, G. E. Dynamic routing between capsules. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2017. p. 3856–3866. Citado 5 vezes nas páginas 162, 166, 167, 172 e 173.

SAHA, S. A comprehensive guide to convolutional neural networks the eli5 way. URL: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53/> (visited on 30/04/2019), 2018. Citado 3 vezes nas páginas 251, 266 e 282.

SAHANI, M. et al. Web-based online embedded door access control and home security system based on face recognition. In: *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*. [S.l.: s.n.], 2015. p. 1–6. Citado 3 vezes nas páginas 691, 706 e 710.

- SAINATH, T. N. et al. Convolutional, long short-term memory, fully connected deep neural networks. In: IEEE. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). [S.l.], 2015. p. 4580–4584. Citado na página 449.
- SAIPULLAH, K. M. Opencv based real-time video processing using android smartphone. *International Journal of Computer Technology and Electronics Engineering*, International Journal of Computer Technology and Electronics Engineering, v. 1, n. 3, p. 1–6, 2012. Citado na página 334.
- SAK, H.; SENIOR, A. W.; BEAUFAYS, F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. 2014. Citado na página 449.
- SALAMON, J.; JACOBY, C.; BELLO, J. P. A dataset and taxonomy for urban sound research. In: *Proceedings of the 22nd ACM international conference on Multimedia*. [S.l.: s.n.], 2014. p. 1041–1044. Citado 3 vezes nas páginas 578, 579 e 597.
- SALOTTI, B. M. Reapresentação voluntária de demonstrações contábeis de 2007 no brasil de acordo com a lei 11.638/07. 2010. Citado na página 336.
- SAMIS, T. doença de pele. 2006. [\(https://revistas.pucsp.br/index.php/RFCMS/article/view/74/pdf/\)](https://revistas.pucsp.br/index.php/RFCMS/article/view/74/pdf/). [Online; accessed 1-September-2006]. Citado na página 135.
- SANTOS, A. P. d. O. et al. Desenvolvimento de descritores de imagens para reconhecimento de padrões de plantas invasoras (folhas largas e folhas estreitas). Universidade Federal de São Carlos, 2009. Citado na página 8.
- SANTOS, C. S. dos; RIBEIRO, A. S. Apicultura uma alternativa na busca do desenvolvimento sustentável. *Revista verde de agroecologia e desenvolvimento sustentável*, v. 4, n. 3, p. 01–06, 2009. Citado na página 383.
- SANTOS, G. L. et al. Analyzing the availability and performance of an e-health system integrated with edge, fog and cloud infrastructures. *Journal of Cloud Computing*, Springer, v. 7, n. 1, p. 16, 2018. Citado 4 vezes nas páginas 312, 313, 521 e 522.
- SANTOS, G. L. et al. The internet of things for healthcare: optimising e-health system availability in the fog and cloud. *International Journal of Computational Science and Engineering*, Inderscience Publishers (IEL), v. 21, n. 4, p. 615–628, 2020. Citado na página 310.
- SARANYA, S. M.; VIJAYALAKSHMI, M. Interactive mobile live video learning system in cloud environment. In: IEEE. 2011 International Conference on Recent Trends in Information Technology (ICRTIT). [S.l.], 2011. p. 673–677. Citado na página 516.
- SATRIA, D.; PARK, D.; JO, M. Recovery for overloaded mobile edge computing. *Future Generation Computer Systems*, Elsevier, v. 70, p. 138–147, 2017. Citado na página 492.
- SATRIA, D.; PARK, D.; JO, M. Recovery for overloaded mobile edge computing. *Future Generation Comp. Syst.*, v. 70, p. 138–147, 2017. Citado na página 494.
- SATYANARAYANAN, M. et al. Pervasive computing: Vision and challenges. *IEEE Personal communications*, v. 8, n. 4, p. 10–17, 2001. Citado na página 335.
- SAÚDE, M. D. Ministério da saúde: Câncer de mama. In: . [s.n.], 2019. Disponível em: [\(<http://portalsms.saude.gov.br/saude-de-a-z/cancer-de-mama>\)](http://portalsms.saude.gov.br/saude-de-a-z/cancer-de-mama). Acesso em: 10.05.2019. Citado na página 120.
- SBD. Sociedade Brasileira de Diabetes: Diretrizes SBD - Retinopatia Diabética. 2014–2015. Acessado em 9 de setembro de 2020. Disponível em: [\(<https://www.diabetes.org.br/profissionais/images/pdf/diabetes-tipo-1/012-Diretrizes-SBD-Retinopatia-Diabetica-pg149.pdf>\)](https://www.diabetes.org.br/profissionais/images/pdf/diabetes-tipo-1/012-Diretrizes-SBD-Retinopatia-Diabetica-pg149.pdf). Citado na página 204.
- SBD. Sociedade Brasileira de Diabetes: Diretrizes SBD - Retinopatia Diabética. 2019–2020. Acessado em 9 de setembro de 2020. Disponível em: [\(<https://www.diabetes.org.br/profissionais/images/DIRETRIZES-COMPLETA-2019-2020.pdf>\)](https://www.diabetes.org.br/profissionais/images/DIRETRIZES-COMPLETA-2019-2020.pdf). Citado na página 203.
- SBM. Sociedade Brasileira de Mastologia. Estimativa 2018 Incidência de mulheres com câncer de mama e menos de 35 anos está entre 4% e 5%. 2018. Citado na página 262.

- SCANLON, P. H. Diabetic retinopathy. In: Elsevier Ltd. [S.l.: s.n.], 2019. Citado na página 37.
- SCHLEUTER, D. et al. A user's guide to functional diversity indices. *Ecological monographs*, Wiley Online Library, v. 80, n. 3, p. 469–484, 2010. Citado na página 72.
- SCHMEIL, M. A. Saúde e tecnologia da informação e comunicação. *Fisioterapia em movimento*, SciELO Brasil, v. 26, n. 3, p. 477–478, 2013. Citado na página 614.
- SCHÖLKOPF, B. et al. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002. (Adaptive computation and machine learning). ISBN 9780262194754. Disponível em: <<https://books.google.com.br/books?id=y8ORL3DWt4sC>>. Citado na página 154.
- SCHÖRKHUBER, C.; Klapuri, A. Constant-q transform toolbox for music processing. In: *7th Sound and Music Computing Conference, Barcelona, Spain*. [S.l.: s.n.], 2010. p. 3–64. Citado na página 573.
- SCHULLER, B.; DORFNER, J.; RIGOLL, G. Determination of nonprototypical valence and arousal in popular music: features and performances. *EURASIP Journal on Audio, Speech, and Music Processing*, Hindawi Publishing Corp., v. 2010, p. 5, 2010. Citado 3 vezes nas páginas 645, 653 e 654.
- SCHULLER, B.; WENINGER, F.; DORFNER, J. Multi-modal non-prototypical music mood analysis in continuous space: Realiability and performances. In: *Proc. 12th Intern. Society for Music Information Retrieval Conference (ISMIR) 2011, ISMIR, Miami, FL, USA*. [S.l.: s.n.], 2011. Citado 2 vezes nas páginas 645 e 652.
- SEBESTYEN, G.; SAPLACAN, G.; KRUCZ, L. Cardionet—a distributed e-health system for patients with cardio-vascular diseases. In: *Workshop on medical informatics”, part of ICCP*. [S.l.: s.n.], 2010. Citado na página 516.
- SEBESTYEN, G.; TIREA, A.; ALBERT, R. Monitoring human activity through portable devices. *Carpathian Journal of Electronic & Computer Engineering*, v. 5, n. 1, 2012. Citado na página 516.
- SEERAM, E. Computed tomography: Physical principles and recent technical advances. *Journal of Medical Imaging and Radiation Sciences*, v. 41, n. 2, p. 87 – 109, 2010. ISSN 1939-8654. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1939865410000317>>. Citado na página 68.
- SELTMAN, H. J. Experimental design and analysis. *Online at: http://www.stat.cmu.edu/, hseltman/309/Book/Book.pdf*, 2012. Citado 3 vezes nas páginas 338, 366 e 402.
- SEPÚLVEDA, G. *Genómica comparativa*. 2019. Disponível em: <<http://knoow.net/ciencterravida/biologia/genomica-comparativa/>>. Acesso em: 11 ago. 2020. Citado na página 608.
- SERIZEL, R. et al. Acoustic features for environmental sound analysis. In: *Computational analysis of sound scenes and events*. [S.l.]: Springer, 2018. p. 71–101. Citado 4 vezes nas páginas 572, 573, 575 e 590.
- SETHARES, W. A. Consonance-based spectral mappings. *Computer Music Journal*, JSTOR, v. 22, n. 1, p. 56–72, 1998. Citado na página 655.
- SETIO, A. A. A. et al. Pulmonary nodule detection in ct images: False positive reduction using multi-view convolutional networks. *IEEE Transactions on Medical Imaging*, v. 35, n. 5, p. 1160–1169, May 2016. ISSN 0278-0062. Citado 3 vezes nas páginas 66, 67 e 77.
- SETIO, A. A. A. et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge. *Medical image analysis*, Elsevier, v. 42, p. 1–13, 2017. Citado na página 240.
- SEVERO, E. B. Quantificação automática do glaucoma utilizando imagens de fundo do olho. Universidade Federal do Pampa, 2014. Citado na página 8.
- SFGATE. *Satellite image shows incredible fiery heat of Camp Fire*. 2018. Disponível em: <<http://bit.ly/32MvObc>>. Citado 2 vezes nas páginas 551 e 552.

- SHAAMILI, R.; RANJITH, R.; SUPRIYA, P. Intelligent traffic light system for unhampered mobility of emergency vehicles. In: IEEE. 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS). [S.l.], 2018. p. 360–363. Citado na página [427](#).
- SHAH, A. S.; ILHAN, H.; TURELI, U. Modeling and performance analysis of the ieee 802.11 p mac for vanets. In: IEEE. 2019 42nd Int. Conf. on Telecommunications and Signal Processing (TSP). [S.l.], 2019. p. 393–396. Citado 3 vezes nas páginas [426](#), [427](#) e [486](#).
- SHAHABI, K.; WILSON, J. P. Casper: Intelligent capacity-aware evacuation routing. *Computers, Environment and Urban Systems*, 2014. Citado na página [541](#).
- SHAHROUDNEJAD, A. et al. Improved explainability of capsule networks: Relevance path by agreement. In: IEEE. 2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP). [S.l.], 2018. p. 549–553. Citado na página [166](#).
- SHAN, F. et al. Lung infection quantification of covid-19 in ct images with deep learning. *arXiv preprint arXiv:2003.04655*, 2020. Citado 2 vezes nas páginas [181](#) e [182](#).
- SHANNON, C. E. A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, ACM New York, NY, USA, v. 5, n. 1, p. 3–55, 2001. Citado na página [220](#).
- SHANNON, C. E.; WEAVER, W. The mathematical theory of information. University of Illinois press, 1949. Citado 2 vezes nas páginas [102](#) e [124](#).
- SHARMA, G.; UMAPATHY, K.; KRISHNAN, S. Trends in audio signal feature extraction methods. *Applied Acoustics*, Elsevier, v. 158, p. 107020, 2020. Citado na página [573](#).
- SHAVETOV, S.; SIVTSOV, V. Access control system based on face recognition. In: 2020 7th International Conference on Control, Decision and Information Technologies (CoDIT). [S.l.: s.n.], 2020. v. 1, p. 952–956. Citado 4 vezes nas páginas [691](#), [695](#), [709](#) e [710](#).
- SHEHA, M. A.; MABROUK, M. S.; SHARAWY, A. Automatic detection of melanoma skin cancer using texture analysis. *International Journal of Computer Applications*, Citeseer, v. 42, n. 20, p. 22–26, 2012. Citado na página [278](#).
- SHELHAMER, E.; LONG, J.; DARRELL, T. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. Citado 3 vezes nas páginas [295](#), [300](#) e [302](#).
- SHEN, W. et al. Multi-scale convolutional neural networks for lung nodule classification. In: SPRINGER. *International Conference on Information Processing in Medical Imaging*. [S.l.], 2015. p. 588–599. Citado na página [24](#).
- SHEYAWYE, T. N.; MEKONNEN, A. A. Benign-malignant lung nodule classification with geometric and appearance histogram features. *arXiv preprint arXiv:1605.08350*, 2016. Citado na página [24](#).
- SHI, C. et al. Serendipity: enabling remote computing among intermittently connected mobile devices. In: ACM. *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. [S.l.], 2012. p. 145–154. Citado 2 vezes nas páginas [335](#) e [336](#).
- SHI, W. et al. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, IEEE, v. 3, n. 5, p. 637–646, 2016. Citado na página [514](#).
- SHI, W. et al. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, v. 3, p. 637–646, 2016. Citado na página [517](#).
- SHI, Y.-Y. et al. A tempo feature via modulation spectrum analysis and its application to music emotion classification. In: IEEE. *Multimedia and Expo, 2006 IEEE International Conference on*. [S.l.], 2006. p. 1085–1088. Citado 2 vezes nas páginas [645](#) e [652](#).
- SHIRINBAB, S.; LUNDBERG, L.; CASALICCHIO, E. Performance evaluation of container and virtual machine running cassandra workload. In: IEEE. *Cloud Computing Technologies and Applications (CloudTech), 2017 3rd International Conference of*. [S.l.], 2017. p. 1–8. Citado 2 vezes nas páginas [412](#) e [413](#).

SHUMWAY, R. H.; STOFFER, D. S. *Time series analysis and its applications: with R examples*. [S.l.]: Springer, 2017. Citado na página 447.

SIEBERTZ, K.; BEBBER, D. V.; HOCHKIRCHEN, T. *Statistische Versuchsplanung: design of experiments (DoE)*. [S.l.]: Springer-Verlag, 2017. Citado 3 vezes nas páginas 338, 366 e 402.

SIEGEL, R. L. et al. Colorectal cancer statistics, 2017. *CA: a cancer journal for clinicians*, Wiley Online Library, v. 67, n. 3, p. 177–193, 2017. Citado 2 vezes nas páginas 23 e 24.

SILVA, A.; SOARES, A. Controle adaptativo de semáforo com pelotões veiculares mais acurados. In: SBC. 37o. *Simpósio Brasileiro de Redes e Sistemas Distribuídos (SBRC)*. [S.l.], 2019. Citado na página 486.

SILVA, A. C. d. Virtualização: um estudo sobre conceitos e técnicas. Universidade Federal Fluminense, 2017. Citado na página 400.

SILVA, A. de Lima Silva e. *Monitoramento Não Invasivo de Colmeias Através da IOT*. 2017. Citado 2 vezes nas páginas 388 e 389.

SILVA, A. M. M. da; PATROCÍNIO, A. C.; SCHIABEL, H. Processamento e análise de imagens médicas. *Revista Brasileira de Física Médica*, v. 13, n. 1, p. 34–48, 2019. Citado na página 216.

SILVA, B. et al. Mercury: An integrated environment for performance and dependability evaluation of general systems. In: *Proceedings of Industrial Track at 45th Dependable Systems and Networks Conference (DSN)*. [S.l.: s.n.], 2015. Citado 4 vezes nas páginas 330, 402, 439 e 535.

SILVA, B. da et al. Evaluation of classical machine learning techniques towards urban sound recognition on embedded systems. *Applied Sciences*, Multidisciplinary Digital Publishing Institute, v. 9, n. 18, p. 3885, 2019. Citado 3 vezes nas páginas 577, 578 e 585.

SILVA, C. C. et al. Diagnóstico de glaucoma em imagens de fundo de olho utilizando estatística espacial. Universidade Federal do Maranhão, 2018. Citado 2 vezes nas páginas 9 e 10.

SILVA, E. A. da; MENDONCA, G. V. 4 - digital image processing. In: CHEN, W.-K. (Ed.). *The Electrical Engineering Handbook*. Burlington: Academic Press, 2005. p. 891 – 910. ISBN 978-0-12-170960-0. Citado na página 206.

SILVA, F. A. et al. Mobile cloud performance evaluation using stochastic models. *IEEE Transactions on Mobile Computing*, IEEE, v. 17, n. 5, p. 1134–1147, 2017. Citado 2 vezes nas páginas 422 e 517.

SILVA, F. A. et al. Mobile cloud performance evaluation using stochastic models. *IEEE Transactions on Mobile Computing*, IEEE, v. 17, n. 5, p. 1134–1147, 2018. Citado 3 vezes nas páginas 310, 490 e 503.

SILVA, F. A. et al. Planning mobile cloud infrastructures using stochastic petri nets and graphic processing units. In: IEEE. *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*. [S.l.], 2015. p. 471–474. Citado 2 vezes nas páginas 310 e 422.

SILVA, F. A. et al. Um middleware para provisionamento de contextos para redes veiculares. *XXXI Simpósio Brasileiro de Redes de Comutadores e Sistemas Distribuídos-SBRC*, p. 615–628, 2013. Citado 2 vezes nas páginas 468 e 469.

SILVA-FILHO, A. G. et al. Energy-aware technology-based dvfs mechanism for the android operating system. In: IEEE. *Computing System Engineering (SBESC), 2012 Brazilian Symposium on*. [S.l.], 2012. p. 184–187. Citado 2 vezes nas páginas 334 e 335.

SILVA, G. L. F. d. et al. Diagnóstico de nódulos pulmonares em imagens de tomografia computadorizada usando redes neurais convolucionais evolutivas. Universidade Federal do Maranhão, 2017. Citado na página 41.

SILVA, I. A. D.; BATALHA, M. A. Taxonomic distinctness and diversity of a hyperseasonal savanna in central brazil. *Diversity and distributions*, Wiley Online Library, v. 12, n. 6, p. 725–730, 2006. Citado na página 55.

- SILVA, R. R. V. et al. *Visão Computacional em Python Utilizando as Bibliotecas. Scikit-image e Scikit-learn.* first. [S.I.]: Livro Anais - Artigos e Minicursos., 2017. v. 1. 407-428 p. (III Escola Regional de Informática do Piauí., v. 1). Sociedade Brasileira de Computação. Citado na página 12.
- SILVA, T. *Diferenciação do padrão de malignidade e benignidade de massas em mamografias utilizando características geométricas e máquina de vetor de suporte.* Tese (Doutorado) — Dissertação de Mestrado. Programa de Pós-Graduação em Ciência da Computação ..., 2016. Citado 2 vezes nas páginas 81 e 93.
- SILVA, T. F. *Diferenciação do padrão de malignidade e benignidade de massas em mamografias utilizando características geométricas e máquina de vetor de suporte.* Tese (Dissertação de Mestrado. Programa de Pós-graduação em Ciência da Computação) — Universidade Federal do Maranhão, 2016. Citado na página 68.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. Citado 7 vezes nas páginas 12, 86, 101, 251, 267, 284 e 285.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014. Citado na página 176.
- SIMPSON, E. H. Measurement of diversity. *nature*, Nature Publishing Group, v. 163, n. 4148, p. 688, 1949. Citado 4 vezes nas páginas 102, 103, 124 e 220.
- SINGADKAR, G. et al. Deep deconvolutional residual network based automatic lung nodule segmentation. *Journal of Digital Imaging*, Springer, p. 1–7, 2020. Citado 3 vezes nas páginas 238, 239 e 244.
- SINGH, D. et al. Classification of covid-19 patients from chest ct images using multi-objective differential evolution-based convolutional neural networks. *European journal of clinical microbiology & infectious diseases : official publication of the European Society of Clinical Microbiology*, April 2020. ISSN 0934-9723. Disponível em: <<https://europepmc.org/articles/PMC7183816>>. Citado na página 182.
- SINGH, P. et al. Architecture for automated tagging and clustering of song files according to mood. *arXiv preprint arXiv:1206.2484*, 2012. Citado na página 645.
- SIVASHANKARI, S.; SHANMUGHAVEL, P. Comparative genomics-a perspective. *Bioinformation*, Biomedical informatics publishing Group, v. 1, n. 9, p. 376, 2007. Citado na página 610.
- SIVASWAMY, J. et al. A comprehensive retinal image dataset for the assessment of glaucoma from the optic nerve head analysis. *JSM Biomedical Imaging Data Papers*, v. 2, n. 1, p. 1004, 2015. Citado na página 11.
- SIVASWAMY, J. et al. Drishti-gs: Retinal image dataset for optic nerve head (onh) segmentation. In: IEEE. 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI). [S.I.], 2014. p. 53–56. Citado 2 vezes nas páginas 59 e 60.
- SIVASWAMY, J. et al. Drishti-gs: Retinal image dataset for optic nerve head (onh) segmentation. In: IEEE. 2014 IEEE 11th international symposium on biomedical imaging (ISBI). [S.I.], 2014. p. 53–56. Citado 2 vezes nas páginas 169 e 170.
- SMITH, B.; WILSON, J. B. A consumer's guide to evenness indices. *Oikos*, JSTOR, p. 70–82, 1996. Citado 3 vezes nas páginas 103, 125 e 126.
- SMITH, D.; CHENG, E.; BURNETT, I. Musical onset detection using mpeg-7 audio descriptors. In: *Proceedings of the 20th International Congress on Acoustics (ICA), Sydney, Australia.* [S.l.: s.n.], 2010. v. 2327, p. 1014. Citado na página 657.
- SMITH, T. F.; WATERMAN, M. S. et al. Identification of common molecular subsequences. *Journal of molecular biology*, Elsevier Science, v. 147, n. 1, p. 195–197, 1981. Citado 2 vezes nas páginas 608 e 613.
- SOARES, H. B. Análise e classificação de imagens de lesões da pele por atributos de cor, forma e textura utilizando máquina de vetor de suporte. Universidade Federal do Rio Grande do Norte, 2008. Citado na página 136.
- SOLOMON, R. C. *The passions: Emotions and the meaning of life.* [S.I.]: Hackett Publishing, 1993. Citado na página 636.

SOMBRA, D. d. S. et al. Monitoramento do desenvolvimento de colônias de abelhas africanizadas sobre a influência do sol e sombra na região semiárida do nordeste brasileiro (mossor-rn). Universidade Federal Rural do Semi-Árido, 2013. Citado na página 382.

SONG, R. et al. extreme gradient boosting for identifying individual users across different digital devices. In: CUI, B. et al. (Ed.). *Web-Age Information Management*. Cham: Springer International Publishing, 2016. p. 43–54. ISBN 978-3-319-39937-9. Citado na página 89.

SONG, Y.; DIXON, S.; PEARCE, M. Evaluation of musical features for emotion classification. In: CITESEER. ISMIR. [S.l.], 2012. p. 523–528. Citado 3 vezes nas páginas 644, 647 e 652.

SOPHARAK, A. et al. Machine learning approach to automatic exudate detection in retinal images from diabetic patients. *Journal of Modern optics*, Taylor & Francis, v. 57, n. 2, p. 124–135, 2010. Citado 2 vezes nas páginas 205 e 213.

SOUZA, E. et al. A modeling approach for cloud infrastructure planning considering dependability and cost requirements. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, IEEE, v. 45, n. 4, p. 549–558, 2014. Citado na página 436.

SOUZA, U. S. *CLASSIFICAÇÃO DE MASSAS NA MAMA A PARTIR DE IMAGENS MAMOGRÁFICAS USANDO ÍNDICE DE DIVERSIDADE DE SHANNON-WIENER*. Dissertação (Mestrado) — PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE/CCET, 2011. Engenharia. Disponível em: [\(<http://tedebc.ufma.br:8080/jspui/handle/tede/467>\)](http://tedebc.ufma.br:8080/jspui/handle/tede/467). Citado na página 152.

SOUZA, U. S. *Classificação de massas na mama a partir de imagens mamográficas usando índice de diversidade de shannon-wiener*. Tese (Dissertação de Mestrado. Programa de Pós-graduação em Ciência da Computação) — Universidade Federal do Maranhão, 2011. Citado 2 vezes nas páginas 68 e 69.

SOUZA, U. S. et al. Classificação de massas na mama a partir de imagens mamográficas usando índice de diversidade de shannon-wiener. Universidade Federal do Maranhão, 2011. Citado 3 vezes nas páginas 54, 56 e 139.

SOUZA, T. I. et al. Um método para detecção e diagnóstico de outliers em dados urbanos via análise multidimensional. In: SBC. *Anais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*. [S.l.], 2018. Citado na página 569.

SRIVASTAVA, R. et al. Using deep learning for robustness to parapapillary atrophy in optic disc segmentation. In: 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI). [S.l.: s.n.], 2015. p. 768–771. ISSN 1945-7928. Citado 2 vezes nas páginas 299 e 305.

STAAL, J. et al. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 2010. Citado na página 299.

STACK, S.; KRYSINSKA, K.; LESTER, D. Gloomy sunday: did the “hungarian suicide song” really create a suicide epidemic? *OMEGA-Journal of death and dying*, SAGE Publications Sage CA: Los Angeles, CA, v. 56, n. 4, p. 349–358, 2008. Citado na página 633.

STALIDZANS, E.; BERZONIS, A. Temperature changes above the upper hive body reveal the annual development periods of honey bee colonies. *Computers and electronics in agriculture*, Elsevier, v. 90, p. 1–6, 2013. Citado na página 382.

STANZIONE, A. et al. Detection of extraprostatic extension of cancer on biparametric mri combining texture analysis and machine learning: preliminary results. *Academic radiology*, Elsevier, 2019. Citado na página 11.

STATISTA. Data volume of consumer-managed internet protocol (ip) traffic in latin america from 2016 to 2021 (in petabytes per month). In: . [s.n.], 2018. Disponível em: [\(<https://www.statista.com/statistics/728329/consumer-managed-ip-traffic-latin-america/>\)](https://www.statista.com/statistics/728329/consumer-managed-ip-traffic-latin-america/). Acesso em: 31.10.2018. Citado na página 360.

STATISTA. Data volume of global consumer ip traffic from 2015 to 2021 (in petabytes per month). In: . [s.n.], 2018. Disponível em: [\(https://www.statista.com/statistics/267202/global-data-volume-of-consumer-ip-traffic/\)](https://www.statista.com/statistics/267202/global-data-volume-of-consumer-ip-traffic/). Acesso em: 31.10.2018. Citado na página 360.

STATISTA Forecast. [\(https://www.statista.com/statistics/245501/multiple-mobile-device-ownership-worldwide/\)](https://www.statista.com/statistics/245501/multiple-mobile-device-ownership-worldwide/). Accessed: 2020-02-14. Citado na página 310.

STEPANOV, A.; SMITH, J. M. Multi-objective evacuation routing in transportation networks. *European Journal of Operational Research*, 2008. Citado 2 vezes nas páginas 548 e 550.

STEPANOV, A.; SMITH, J. M. Multi-objective evacuation routing in transportation networks. *European Journal of Operational Research*, v. 198, n. 2, p. 435 – 446, 2009. ISSN 0377-2217. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0377221708007443>. Citado 2 vezes nas páginas 675 e 679.

STREICH, S. et al. *Music complexity: a multi-faceted description of audio content*. [S.l.]: Universitat Pompeu Fabra Barcelona, Spain, 2006. Citado na página 659.

STRONGMAN, K. T. *The psychology of emotion*. [S.l.]: Wiley New York, 1978. Citado na página 636.

SU, Y. et al. Environment sound classification using a two-stream cnn based on decision-level fusion. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 19, n. 7, p. 1733, 2019. Citado 4 vezes nas páginas 577, 578, 585 e 590.

SUBODH, M. Transfer learning using tensorflow. URL: <https://medium.com/@subodh.malgonde/transfer-learning-using-tensorflow-52a4f6bcde3e/> (visited on 06/05/2019), 2018. Citado 2 vezes nas páginas 253 e 269.

Sucipto, K. et al. Keep your nice friends close, but your rich friends closer — computation offloading using nfc. In: *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. [S.l.: s.n.], 2017. p. 1–9. Citado na página 490.

SZEGEDY, C. et al. Inception-v4, inception-resnet and the impact of residual connections on learning. In: *Thirty-First AAAI Conference on Artificial Intelligence*. [S.l.: s.n.], 2017. Citado na página 285.

SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 2818–2826. Citado 3 vezes nas páginas 267, 284 e 286.

TALEB, I.; OUIS, M. E. A.; MAMMAR, M. O. Access control using automated face recognition: Based on the pca lda algorithms. In: *2014 4th International Symposium ISKO-Maghreb: Concepts and Tools for knowledge Management (ISKO-Maghreb)*. [S.l.: s.n.], 2014. p. 1–5. Citado 4 vezes nas páginas 690, 697, 709 e 710.

TALIB, M. A. et al. Systematic literature review on internet-of-vehicles communication security. *International Journal of Distributed Sensor Networks*, SAGE Publications Sage UK: London, England, v. 14, n. 12, p. 1550147718815054, 2018. Citado na página 421.

TAN, J. H. et al. Segmentation of optic disc, fovea and retinal vasculature using a single convolutional neural network. ARXIV, 2017. Citado 2 vezes nas páginas 299 e 305.

TAN, X. et al. Molecular stratification by bcl2a1 and aim2 provides additional prognostic value in penile squamous cell carcinoma. *Theranostics*, Ivyspring International Publisher, v. 11, n. 3, p. 1364, 2021. Citado na página 184.

TASKIRAN, M.; KAHRAMAN, N.; ERDEM, C. E. Face recognition: Past, present and future (a review). *Digital Signal Processing*, v. 106, p. 102809, 2020. ISSN 1051-2004. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1051200420301548>. Citado 3 vezes nas páginas 685, 686 e 687.

TAUD, H.; MAS, J. Multilayer perceptron (mlp). In: *Geomatic Approaches for Modeling Land Change Scenarios*. [S.l.]: Springer, 2018. p. 451–455. Citado na página 222.

TAVARES, E.; MACIEL, P.; SILVA, B. Modeling hard real-time systems considering inter-task relations, dynamic voltage scaling and overheads. *Microprocessors and Microsystems*, Elsevier, v. 32, n. 8, p. 460–473, 2008. Citado na página 335.

TAVARES, E. et al. Dynamic voltage scaling in hard real-time systems considering precedence and exclusion relations. In: *IEEE. Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*. [S.l.], 2007. p. 2496–2501. Citado na página 335.

TEAM, O. D. *OpenCv: Open Source Computer Vision Library*. [S.l.]: Erişim Tarihi, 2017. Citado na página 59.

TEIXEIRA, M. Avaliação de desempenho de web services através de redes de petri estocásticas. 2009. Citado 2 vezes nas páginas 471 e 472.

THAM, Y.-C. et al. Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis. *Ophthalmology*, Elsevier, v. 121, n. 11, p. 2081–2090, 2014. Citado 2 vezes nas páginas 162 e 163.

THAYER, R. E. *The biopsychology of mood and arousal*. [S.l.]: Oxford University Press, 1990. Citado na página 636.

THEANO, D. T. *Deep Learning 0.1 Documentation*. [S.l.], 2013. Disponível em: <http://deeplearning.net/tutorial/fcn_2D_segm.html>. Acesso em: 11.9.2018. Citado na página 297.

THIAGARAJAN, N. et al. Who killed my battery?: analyzing mobile browser energy consumption. In: *ACM. Proceedings of the 21st international conference on World Wide Web*. [S.l.], 2012. p. 41–50. Citado 3 vezes nas páginas 334, 341 e 342.

THOMAS, D. et al. Multiscale co-simulation of energyplus and citysim models derived from a building information model. In: *Bausim 2014: Fifth German-Austrian IBPSA Conference*. [S.l.: s.n.], 2014. Citado 3 vezes nas páginas 338, 366 e 402.

THYLEFORS, B.; NEGREL, A. The global impact of glaucoma. *Bulletin of the World Health Organization*, World Health Organization, v. 72, n. 3, p. 323, 1994. Citado na página 8.

TIGRE, M. et al. Modeling the availability of an e-health system integrated with edge, fog and cloud infrastructures. *IEEE ISCC*, p. 416–421, 2018. Citado 2 vezes nas páginas 521 e 522.

TILMAN, D. Functional diversity. in encyclopedia of biodiversity. In: . [S.l.]: (S.A. Levin, ed.). Academic Press, San Diego, p. 109-120, 2001. Citado na página 72.

TIMES, L. A. *As California's deadliest wildfire closed in, evacuation orders were slow to arrive*. 2018. Disponível em: <<https://lat.ms/2s3B1Uf>>. Citado na página 555.

TING, D. S. W.; CHEUNG, G. C. M.; WONG, T. Y. Diabetic retinopathy: global prevalence, major risk factors, screening practices and public health challenges: a review. *Clinical and Experimental Ophthalmology*, 2016. Citado na página 39.

TOMAZELLI, J. G. et al. Avaliação das ações de detecção precoce do câncer de mama no brasil por meio de indicadores de processo: estudo descritivo com dados do sismama, 2010-2011. *Epidemiologia e Serviços de Saúde*, SciELO Public Health, v. 26, p. 61–70, 2016. Citado na página 97.

TONG, L.; LI, Y.; GAO, W. A hierarchical edge cloud architecture for mobile computing. In: *IEEE. INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. [S.l.], 2016. p. 1–9. Citado 2 vezes nas páginas 405 e 495.

TOPI, M. et al. Robust texture classification by subsets of local binary patterns. In: *IEEE. Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*. [S.l.], 2000. v. 3, p. 935–938. Citado 2 vezes nas páginas 269 e 287.

TORQUATO, M. et al. Sware: A methodology for software aging and rejuvenation experiments. *Journal of Information Systems Engineering & Management*, Modestum LTD, v. 3, n. 2, p. 15, 2018. Citado 2 vezes nas páginas 444 e 454.

TORQUATO, M.; VIEIRA, M. An experimental study of software aging and rejuvenation in dockerd. In: IEEE. 2019 15th European Dependable Computing Conference (EDCC). [S.l.], 2019. p. 1–6. Citado 5 vezes nas páginas 444, 450, 451, 454 e 457.

TORRALBO-MUÑOZ, J. L. et al. Smartfridge: The intelligent system that controls your fridge. In: IEEE. 2018 Fifth International Conference on Internet of Things: Systems, Management and Security. [S.l.], 2018. p. 200–207. Citado na página 515.

Torres, W. et al. Functional diversity applied to the false positive reduction in breast tissues based on digital mammography. In: 2018 IEEE Symposium on Computers and Communications (ISCC). [S.l.: s.n.], 2018. p. 01120–01125. ISSN 1530-1346. Citado na página 71.

TORREY, L.; SHAVLIK, J. Transfer learning. In: *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. [S.l.]: IGI Global, 2010. p. 242–264. Citado 4 vezes nas páginas 85, 252, 268 e 283.

TRAMONTE, M. S. et al. Delay in diagnosis of breast cancer in a public oncologic hospital. *Medicina (Ribeirão Preto. Online)*, v. 49, n. 5, p. 451–462, 2016. Citado 2 vezes nas páginas 80 e 262.

TRAN, T. X. et al. Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges. *arXiv preprint arXiv:1612.03184*, 2016. Citado na página 493.

Tran, T. X. et al. Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges. *IEEE Communications Magazine*, v. 55, n. 4, p. 54–61, April 2017. ISSN 0163-6804. Citado na página 490.

TRESTIAN, R. et al. Energy consumption analysis of video streaming to android mobile devices. In: IEEE. *Network Operations and Management Symposium (NOMS), 2012 IEEE*. [S.l.], 2012. p. 444–452. Citado 3 vezes nas páginas 334, 341 e 342.

TRINH, C.; YAO, L. Energy-aware mobile edge computing for low-latency visual data processing. p. 128–133, 2017. Citado 5 vezes nas páginas 313, 401, 405, 406 e 495.

TRIVEDI, K. *Probability and Statistics with Reliability, Queueing, and Computer Science Applications*. 2. ed. [S.l.]: Wiley Interscience Publication, 2002. Citado 3 vezes nas páginas 310, 424 e 517.

TROHIDIS, K. et al. Multi-label classification of music into emotions. In: ISMIR. [S.l.: s.n.], 2008. v. 8, p. 325–330. Citado na página 634.

TROHIDIS, K. et al. Multi-label classification of music by emotion. *EURASIP Journal on Audio, Speech, and Music Processing*, Springer, v. 2011, n. 1, p. 4, 2011. Citado na página 643.

TRUCCO, E. et al. Validating retinal fundus image analysis algorithms: Issues and a proposal validating retinal fundus image analysis algorithms. *Investigative ophthalmology & visual science*, The Association for Research in Vision and Ophthalmology, v. 54, n. 5, p. 3546–3559, 2013. Citado 2 vezes nas páginas 59 e 60.

TSOCHATZIDIS, L. et al. Computer-aided diagnosis of mammographic masses based on a supervised content-based image retrieval approach. *Pattern Recognition*, Elsevier, v. 71, p. 106–117, 2017. Citado na página 162.

TSUJI, T. et al. Classification of optical coherence tomography images using a capsule network. *BMC ophthalmology*, BioMed Central, v. 20, n. 1, p. 1–9, 2020. Citado na página 172.

TUCCITTO, A. et al. Cancer stem cell challenges in melanoma characterization and treatment. In: *Cancer Stem Cell Resistance to Targeted Therapy*. [S.l.]: Springer, 2019. p. 115–135. Citado na página 277.

TURBIANI, R. *Cegueira afeta 39 milhões de pessoas no mundo; conheça suas principais causas*. 2019. Disponível em: <https://www.bbc.com/portuguese/geral-48634186>. Citado na página 7.

- TZANETAKIS, G.; COOK, P. Marsyas: A framework for audio analysis. *Organised sound*, Cambridge University Press, v. 4, n. 3, p. 169–175, 2000. Citado na página 650.
- TZANETAKIS, G.; ERMOLINSKIY, A.; COOK, P. R. Pitch histograms in audio and symbolic music information retrieval. In: ISMIR. [S.l.: s.n.], 2002. Citado na página 658.
- UJLAMBKAR, A. M.; ATTAR, V. Z. Automatic mood classification model for indian popular music. In: IEEE. *Modelling Symposium (AMS), 2012 Sixth Asia*. [S.l.], 2012. p. 7–12. Citado na página 645.
- UMBAUGH, S. E. *Computer vision and image processing: a practical approach using cvip tools with cdrom*. [S.l.]: Prentice Hall PTR, 1997. Citado na página 54.
- ÜNVER, H. M.; AYAN, E. Skin lesion segmentation in dermoscopic images with combination of yolo and grabcut algorithm. *Diagnostics*, Multidisciplinary Digital Publishing Institute, v. 9, n. 3, p. 72, 2019. Citado 2 vezes nas páginas 224 e 230.
- Ushizima, D. M.; Lorena, A. C.; de Carvalho, A. C. P. L. F. Support vector machines applied to white blood cell recognition. In: *Fifth International Conference on Hybrid Intelligent Systems (HIS'05)*. [S.l.: s.n.], 2005. p. 6 pp.–. Citado na página 154.
- UTIYAMA, R. Y. et al. Melanoma metastático em cólon: relato de caso. *Arquivos Médicos*, v. 64, n. 3, p. 261–4, 2019. Citado na página 217.
- V., B. S.; UNNIKRISHNAN, A.; BALAKRISHNAN, K. Gray level co-occurrence matrices: Generalisation and some new features. *CoRR*, abs/1205.4831, 2012. Disponível em: <http://arxiv.org/abs/1205.4831>. Citado na página 72.
- VALARMATHIE, P. et al. Classification of mammogram masses using selected texture, shape and margin features with multilayer perceptron classifier. *Biomedical Research*, 2016. Citado 3 vezes nas páginas 117, 119 e 132.
- van Ginneken, B. et al. Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. In: *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*. [S.l.: s.n.], 2015. p. 286–289. Citado 2 vezes nas páginas 150 e 157.
- VANDAMME, P. et al. Polyphasic taxonomy, a consensus approach to bacterial systematics. *Microbiological reviews*, Am Soc Microbiol, v. 60, n. 2, p. 407–438, 1996. Citado na página 55.
- VAPNIK, V. The support vector method of function estimation. In: _____. *Nonlinear Modeling: Advanced Black-Box Techniques*. Boston, MA: Springer US, 1998. p. 55–85. ISBN 978-1-4615-5703-6. Citado 2 vezes nas páginas 88 e 154.
- VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 2013. Citado na página 88.
- VAQUERO, L. M. et al. A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, ACM, v. 39, n. 1, p. 50–55, 2008. Citado na página 310.
- VARGAS, A.; PAES, A.; VASCONCELOS, C. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: *Proceedings of the XXIX Conference on Graphics, Patterns and Images*. [S.l.: s.n.], 2016. p. 1–4. Citado na página 99.
- VEELING, B. S. et al. Rotation equivariant cnns for digital pathology. In: SPRINGER. *International Conference on Medical image computing and computer-assisted intervention*. [S.l.], 2018. p. 210–218. Citado 3 vezes nas páginas 248, 249 e 259.
- VEERASWAMY, A. et al. The simulation of urban-scale evacuation scenarios with application to the swinley forest fire. *Safety Science*, 2018. Citado 2 vezes nas páginas 549 e 550.
- VELÁSQUEZ, W.; MUÑOZ-ARCENTALES, A.; RODRIGUEZ, J. S. A case study: Ingestion analysis of wsn data in databases using docker. In: IEEE. *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*. [S.l.], 2018. p. 1–6. Citado 3 vezes nas páginas 400, 412 e 413.

- VESAL, S. et al. Classification of breast cancer histology images using transfer learning. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 812–819. Citado 3 vezes nas páginas 104, 105 e 263.
- VETRO, A.; ARDITO, L.; MORISIO, M. Definition, implementation and validation of energy code smells: an exploratory study on an embedded system. Citeseer, 2013. Citado na página 335.
- VIEIRA, D. S. C. et al. Carcinoma de mama: novos conceitos na classificação. *Revista Brasileira de Ginecologia e Obstetricia*, SciELO Brasil, 2008. Citado na página 262.
- VIERA, A. J.; GARRETT, J. M. et al. Understanding interobserver agreement: the kappa statistic. *Fam med*, v. 37, n. 5, p. 360–363, 2005. Citado na página 17.
- Vinícius dos Santos Ferreira, M. et al. Convolutional neural network and texture descriptor-based automatic detection and diagnosis of glaucoma. *Expert Systems with Applications*, v. 110, p. 250–263, 2018. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417418303567>. Citado na página 63.
- VIRMANI, C. et al. Applications of machine learning in cyber security. In: *Handbook of Research on Machine and Deep Learning Applications for Cyber Security*. [S.l.]: IGI Global, 2020. p. 83–103. Citado na página 448.
- VIRTANEN, T.; PLUMBLEY, M. D.; ELLIS, D. *Computational analysis of sound scenes and events*. [S.l.]: Springer, 2018. Citado na página 592.
- VIRTANEN, T.; PLUMBLEY, M. D.; ELLIS, D. Introduction to sound scene and event analysis. In: *Computational analysis of sound scenes and events*. [S.l.]: Springer, 2018. p. 3–12. Citado 3 vezes nas páginas 570, 571 e 575.
- VOGADO, L. H. et al. Leukemia diagnosis in blood slides using transfer learning in cnns and svm for classification. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 72, p. 415–422, 2018. Citado 3 vezes nas páginas 575, 594 e 595.
- VOSPER, M.; GRAHAM, D.; CLOKE, P. *Principles and Applications of Radiological Physics E-Book*. [S.l.]: Elsevier Health Sciences, 2011. Citado na página 152.
- VYAS, G.; DUTTA, M. K. Automatic mood detection of indian music using mfccs and k-means algorithm. In: IEEE. *Contemporary Computing (IC3), 2014 Seventh International Conference on*. [S.l.], 2014. p. 117–122. Citado na página 644.
- WAHYUDONO, B.; OGI, D. Implementation of two factor authentication based on rfid and face recognition using lbp algorithm on access control system. In: *2020 International Conference on ICT for Smart Society (ICISS)*. [S.l.: s.n.], 2020. CFP2013V-ART, p. 1–6. Citado 4 vezes nas páginas 691, 702, 703 e 710.
- WALT, S. van der et al. scikit-image: image processing in Python. *PeerJ*, v. 2, p. e453, 6 2014. ISSN 2167-8359. Disponível em: <https://doi.org/10.7717/peerj.453>. Citado 2 vezes nas páginas 170 e 172.
- WAN, J. et al. Vcmia: A novel architecture for integrating vehicular cyber-physical systems and mobile cloud computing. *Mobile Networks and Applications*, Springer, v. 19, n. 2, p. 153–160, 2014. Citado na página 422.
- WAN, J. et al. Context-aware vehicular cyber-physical systems with cloud support: architecture, challenges, and solutions. *IEEE Communications Magazine*, IEEE, v. 52, n. 8, p. 106–113, 2014. Citado na página 422.
- WANG, B. *Implementation of a dynamic programming algorithm for DNA Sequence alignment on the Cell Matrix architecture*. Tese (Doutorado) — Utah State University, Department of Computer Science, 2002. Citado na página 612.
- WANG, D. et al. Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*, 2016. Citado 2 vezes nas páginas 248 e 249.
- WANG, H. et al. Breast mass classification via deeply integrating the contextual information from multi-view data. *Pattern Recognition*, Elsevier, v. 80, p. 42–52, 2018. Citado 3 vezes nas páginas 118, 119 e 132.

- WANG, L. et al. Visual tracking with fully convolutional networks. In: *The IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2015. Citado na página 41.
- WANG, S. et al. A deep learning algorithm using ct images to screen for corona virus disease (covid-19). *medRxiv*, Cold Spring Harbor Laboratory Press, 2020. Disponível em: <https://www.medrxiv.org/content/early/2020/04/24/2020.02.14.20023028>. Citado 2 vezes nas páginas 154 e 160.
- WANG, S. et al. A deep learning algorithm using ct images to screen for corona virus disease (covid-19). *MedRxiv*, Cold Spring Harbor Laboratory Press, 2020. Citado 4 vezes nas páginas 182, 191, 192 e 201.
- WANG, S. et al. A survey on mobile edge networks: Convergence of computing, caching and communications. *IEEE Communications Surveys and Tutorials*, IEEE, v. 5, p. 6757–6779, 2017. Citado na página 401.
- WANG, W.-J. et al. Adaptive scheduling for parallel tasks with qos satisfaction for hybrid cloud environments. *The Journal of Supercomputing*, Springer, v. 66, n. 2, p. 783–811, 2013. Citado na página 516.
- Wang, X. et al. A weakly-supervised framework for covid-19 classification and lesion localization from chest ct. *IEEE Transactions on Medical Imaging*, p. 1–1, 2020. Citado 2 vezes nas páginas 155 e 160.
- WANG, Z. et al. Classification of breast cancer histopathological images using convolutional neural networks with hierarchical loss and global pooling. In: SPRINGER. *International Conference Image Analysis and Recognition*. [S.l.], 2018. p. 745–753. Citado na página 263.
- Wang, Z.; Liu, Q.; Dou, Q. Contrastive cross-site learning with redesigned net for covid-19 ct classification. *IEEE Journal of Biomedical and Health Informatics*, v. 24, n. 10, p. 2806–2813, 2020. Citado 2 vezes nas páginas 155 e 160.
- WEAVER, A. Biometric authentication. *Computer*, v. 39, n. 2, p. 96–97, 2006. Citado na página 683.
- WEBB, C. O.; LOSOS, A. E. J. B. Exploring the phylogenetic structure of ecological communities: An example for rain forest trees. *The American Naturalist*, [The University of Chicago Press, The American Society of Naturalists], v. 156, n. 2, p. 145–155, 2000. ISSN 00030147, 15375323. Citado 3 vezes nas páginas 184, 185 e 186.
- WEINREB, R. N.; KHAW, P. T. Primary open-angle glaucoma. *THE LANCET*, 2004. Citado 2 vezes nas páginas 293 e 294.
- WEKA. WEKA. 2018. <https://www.cs.waikato.ac.nz/ml/weka/>. [Online; accessed 19-November-2017]. Citado 2 vezes nas páginas 140 e 143.
- WEN-BO, H. et al. Automatic detection of optic disc in retina image using cnn and crf. In: IEE SmartWorld. [S.l.]: Publishing Press, 2018. Citado 2 vezes nas páginas 43 e 49.
- WHO, W. H. O. WHO Coronavirus Disease (COVID-19) Dashboard. 2020. Disponível em: <https://covid19.who.int/>. Acesso em: (18 de outubro de 2020). Citado 4 vezes nas páginas 150, 151, 181 e 182.
- WIECZORKOWSKA, A. et al. Creating reliable database for experiments on extracting emotions from music. In: *Intelligent Information Processing and Web Mining*. [S.l.]: Springer, 2005. p. 395–402. Citado na página 643.
- Win, K. Y.; Choomchuay, S. Automated detection of exudates using histogram analysis for digital retinal images. In: *2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*. [S.l.: s.n.], 2016. p. 1–6. Citado na página 204.
- WIRED. *These Wind Patterns Explain Why California's Wildfires Are So Bad*. 2018. Disponível em: <http://bit.ly/2DdgGhL>. Citado na página 551.
- WITTEN, I. H. et al. *Data Mining, Fourth Edition: Practical Machine Learning Tools and Techniques*. 4th. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016. ISBN 0128042915, 9780128042915. Citado 2 vezes nas páginas 73 e 221.

- WIZBICKI, A. S.; BATTISTI, G. Reconhecimento de padrões em imagens aplicando visão computacional. *Salão do Conhecimento*, v. 2, n. 01, 2014. Citado na página 221.
- WRIGHT, M. N.; WAGER, S.; PROBST, P. Ranger: A fast implementation of random forests. *R package version 0.5. 0, URL https://CRAN.R-project.org/package=ranger*, 2016. Citado 5 vezes nas páginas 88, 108, 126, 153 e 190.
- WU, C.-W.; VINTON, M. Blind bandwidth extension using k-means and support vector regression. In: IEEE. 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). [S.l.], 2017. p. 721–725. Citado 2 vezes nas páginas 573 e 574.
- WU, D.; CUI, Y. Disaster early warning and damage assessment analysis using social media data and geo-location information. *Decision Support Systems*, Elsevier, 2018. Citado 2 vezes nas páginas 669 e 670.
- XAVIER, M. G. et al. Containers or hypervisors, which is better for database consolidation? 2016. Citado na página 400.
- XIA, C. et al. Lane scheduling around crossroads for edge computing based autonomous driving. *Journal of Systems Architecture*, Elsevier, v. 95, p. 1–8, 2019. Citado na página 517.
- XIA, F. et al. Phone2cloud: Exploiting computation offloading for energy saving on smartphones in mobile cloud computing. *Information Systems Frontiers*, Springer, v. 16, n. 1, p. 95–111, 2014. Citado 2 vezes nas páginas 335 e 336.
- XIAN, C.; LU, Y.-H.; LI, Z. Adaptive computation offloading for energy conservation on battery-powered systems. In: IEEE. *Parallel and Distributed Systems, 2007 International Conference on*. [S.l.], 2007. v. 2, p. 1–8. Citado 2 vezes nas páginas 335 e 336.
- XIAO, Y. et al. A system-level model for runtime power estimation on mobile devices. In: IEEE. *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*. [S.l.], 2010. p. 27–34. Citado na página 334.
- XIAO, Z. et al. An intelligent traffic light control approach for reducing vehicles co2 emissions in vanet. In: *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. [S.l.: s.n.], 2015. Citado na página 486.
- XIAOLI, W. et al. Reduction of stochastic petri nets for reliability analysis. *The Eighth International Conference on Electronic Measurement and Instruments*, ICEMI, 2007. Citado na página 311.
- XIE, J. et al. A survey of machine learning techniques applied to software defined networking (sdn): Research issues and challenges. *IEEE Communications Surveys & Tutorials*, IEEE, v. 21, n. 1, p. 393–430, 2018. Citado na página 448.
- XIE, J.; ZHU, M. Investigation of acoustic and visual features for acoustic scene classification. *Expert Systems with Applications*, Elsevier, v. 126, p. 20–29, 2019. Citado na página 573.
- XING, B. et al. Emotion-driven chinese folk music-image retrieval based on de-svm. *Neurocomputing*, Elsevier, v. 148, p. 619–627, 2015. Citado na página 646.
- XING, B. et al. Human-centric music medical therapy exploration system. In: ACM. *Proceedings of the 2013 ACM SIGCOMM workshop on Future human-centric multimedia networking*. [S.l.], 2013. p. 3–8. Citado na página 644.
- XUE, L.-Y. et al. Optic disk detection and segmentation for retinal images using saliency model based on clustering. In: *Journal of Computers Vol. 29 Nº 5*. [S.l.]: Publishing Press, 2017. p. 66–79. Citado 2 vezes nas páginas 43 e 49.
- XUE, S. Y. J. et al. Deep membrane systems for multitask segmentation in diabetic retinopathy. In: Elsevier B.V. [S.l.]: Publishing Press, 2019. Citado 3 vezes nas páginas 38, 43 e 49.
- YADAV, D.; SARATHI, M. P.; DUTTA, M. K. Classification of glaucoma based on texture features using neural networks. In: IEEE. *Contemporary Computing (IC3), 2014 Seventh International Conference on*. [S.l.], 2014. p. 109–112. Citado 2 vezes nas páginas 59 e 63.

- Yadav, G.; Maheshwari, S.; Agarwal, A. Contrast limited adaptive histogram equalization based enhancement for real time video system. In: *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. [S.l.: s.n.], 2014. p. 2392–2397. Citado na página 156.
- YAKHCHI, M. et al. Neural network based approach for time to crash prediction to cope with software aging. *Journal of Systems Engineering and Electronics*, BIAI, v. 26, n. 2, p. 407–414, 2015. Citado 2 vezes nas páginas 450 e 451.
- YANG, H.; CHEN, Y.-P. Data mining in lung cancer pathologic staging diagnosis: Correlation between clinical and pathology information. *Expert Systems with Applications*, Elsevier, v. 42, p. 6168–6176, 2015. Citado na página 24.
- YANG, L.; YANG, S.-H.; PLOTNICK, L. How the internet of things technology enhances emergency response operations. *Technological Forecasting and Social Change*, Elsevier, v. 80, n. 9, p. 1854–1867, 2013. Citado na página 516.
- YANG, X.; DONG, Y.; LI, J. Review of data features-based music emotion recognition methods. *Multimedia Systems*, Springer, v. 24, n. 4, p. 365–389, 2018. Citado na página 650.
- YANG, X. et al. On the design of solfeggio audio machine assessment system. In: IEEE. *2019 IEEE 11th International Conference on Communication Software and Networks (ICCSN)*. [S.l.], 2019. p. 234–238. Citado na página 573.
- YANG, Y.-H.; CHEN, H. H. Prediction of the distribution of perceived music emotions using discrete samples. *IEEE Transactions on Audio, Speech, and Language Processing*, IEEE, v. 19, n. 7, p. 2184–2196, 2011. Citado 4 vezes nas páginas 645, 652, 655 e 656.
- YANG, Y.-H.; CHEN, H. H. Ranking-based emotion recognition for music organization and retrieval. *IEEE Transactions on Audio, Speech, and Language Processing*, IEEE, v. 19, n. 4, p. 762–774, 2011. Citado 2 vezes nas páginas 645 e 656.
- YAO, L. et al. V2x routing in a vanet based on the hidden markov model. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 19, n. 3, p. 889–899, 2017. Citado 2 vezes nas páginas 426 e 427.
- YEH, C.-H.; LIN, H.-H.; CHANG, H.-T. An efficient emotion detection scheme for popular music. In: IEEE. *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*. [S.l.], 2009. p. 1799–1802. Citado na página 637.
- YEH, C.-H. et al. Popular music representation: chorus detection & emotion recognition. *Multimedia tools and applications*, Springer, v. 73, n. 3, p. 2103–2128, 2014. Citado 3 vezes nas páginas 645, 650 e 659.
- YIN, D. B. M. et al. Electronic door access control using myaccess two-factor authentication scheme featuring near-field communication and eigenface-based face recognition using principal component analysis. In: *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication*. New York, NY, USA: Association for Computing Machinery, 2016. (IMCOM '16). ISBN 9781450341424. Disponível em: <https://doi.org.ez17.periodicos.capes.gov.br/10.1145/2857546.2857548>. Citado 4 vezes nas páginas 691, 708, 709 e 710.
- YIN, X. et al. Availability-aware service function chain placement in mobile edge computing. In: IEEE. *2020 IEEE World Congress on Services (SERVICES)*. [S.l.], 2020. p. 69–74. Citado 2 vezes nas páginas 312 e 313.
- YOON, D. C. et al. Digital radiographic image processing and analysis. *Dental Clinics of North America*, v. 62, n. 3, p. 341 – 359, 2018. ISSN 0011-8532. Emerging Imaging Technologies in Dentomaxillofacial Radiology. Citado na página 206.
- YOUENS-CLARK, K. et al. Cmap 1.01: a comparative mapping application for the internet. *Bioinformatics*, v. 25, n. 22, p. 3040–3042, 2009. Citado na página 609.
- YOUNES, M. B.; BOUKERCHE, A. Intelligent traffic light controlling algorithms using vehicular networks. *IEEE trans. on vehicular technology*, IEEE, v. 65, n. 8, p. 5887, 2015. Citado na página 486.

- YOUNES, M. B.; BOUKERCHE, A. A performance evaluation of an efficient traffic congestion detection protocol (ecode) for intelligent transportation systems. *Ad Hoc Networks*, Elsevier, v. 24, p. 317–336, 2015. Citado na página 486.
- YU, J. et al. Automated optic nerve analysis for diagnostic support in glaucoma. In: IEEE. *Computer-Based Medical Systems, 2005. Proceedings. 18th IEEE Symposium on*. [S.l.], 2005. p. 97–102. Citado na página 58.
- YU, L. et al. Automated melanoma recognition in dermoscopy images via very deep residual networks. *IEEE transactions on medical imaging*, IEEE, v. 36, n. 4, p. 994–1004, 2017. Citado 4 vezes nas páginas 279, 280, 281 e 292.
- Yu, S.; Xiao, D.; Kanagasingam, Y. Exudate detection for diabetic retinopathy with convolutional neural networks. In: 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). [S.l.: s.n.], 2017. p. 1744–1747. ISSN 1558-4615. Citado 2 vezes nas páginas 205 e 213.
- YUAN, L.; MENG, X.-Y. Reliability analysis of a warm standby repairable system with priority in use. *Applied Mathematical Modelling*, Elsevier, v. 35, n. 9, p. 4295–4303, 2011. Citado 2 vezes nas páginas 325 e 436.
- YUANZHE; SHANGGUANG. An energy-aware edge server placement algorithm in mobile edge computing. In: IEEE International Conference on Edge Computing (EDGE). [S.l.: s.n.], 2018. Citado 2 vezes nas páginas 313 e 406.
- YUHENG, S.; HAO, Y. Image segmentation algorithms overview. *arXiv preprint arXiv:1707.02051*, 2017. Citado na página 236.
- ZAKARYA, M. et al. Epcaware: a game-based, energy, performance and cost efficient resource management technique for multi-access edge computing. *IEEE Transactions on Services Computing*, IEEE, 2020. Citado na página 313.
- ZELJKOVIĆ, V. et al. Personal access control system using moving object detection and face recognition. In: 2014 International Conference on High Performance Computing Simulation (HPCS). [S.l.: s.n.], 2014. p. 662–669. Citado 6 vezes nas páginas 690, 702, 704, 705, 709 e 710.
- ZENG, W.; CHURCH, R. L. *Finding shortest paths on real road networks: the case for A**. Zenodo, 2009. Disponível em: <https://doi.org/10.1080/13658810801949850>. Citado na página 542.
- ZENG, Z.-Q. et al. Fast training support vector machines using parallel sequential minimal optimization. In: IEEE. 2008 3rd international conference on intelligent system and knowledge engineering. [S.l.], 2008. v. 1, p. 997–1001. Citado 2 vezes nas páginas 140 e 222.
- ZHAI, P. et al. Multi-task learning for lung nodule classification on chest ct. *IEEE Access*, IEEE, v. 8, p. 180317–180327, 2020. Citado 2 vezes nas páginas 31 e 35.
- ZHANG, H. et al. Connecting intelligent things in smart hospitals using nb-iot. *IEEE Internet of Things Journal*, IEEE, v. 5, n. 3, p. 1550–1560, 2018. Citado 3 vezes nas páginas 514, 521 e 522.
- ZHANG, J. et al. Bridge the semantic gap between pop music acoustic feature and emotion: Build an interpretable model. *Neurocomputing*, Elsevier, v. 208, p. 333–341, 2016. Citado 6 vezes nas páginas 638, 639, 646, 652, 654 e 656.
- ZHANG, J. L. et al. Feature selection and feature learning in arousal dimension of music emotion by using shrinkage methods. *Multimedia Systems*, Springer, v. 23, n. 2, p. 251–264, 2017. Citado 3 vezes nas páginas 646, 652 e 659.
- ZHANG, K.; SUN, S. Web music emotion recognition based on higher effective gene expression programming. *Neurocomputing*, Elsevier, v. 105, p. 100–106, 2013. Citado na página 644.
- ZHANG, L. et al. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In: IEEE. *Hardware/Software Codesign and System Synthesis (CODES+ ISSS), 2010 IEEE/ACM/IFIP International Conference on*. [S.l.], 2010. p. 105–114. Citado 4 vezes nas páginas 334, 341, 344 e 346.

- ZHANG, X. et al. Improving cloud gaming experience through mobile edge computing. *IEEE Wireless Communications*, IEEE, 2019. Citado na página 496.
- ZHANG, X. et al. Faster parking and less cruise for public parking spot discovery: Modeling and analysis based on timed petri nets. In: IEEE. 2016 IEEE 13th International Conference on Networking, Sensing, and Control (ICNSC). [S.l.], 2016. p. 1–6. Citado 5 vezes nas páginas 422, 426, 427, 468 e 486.
- ZHANG YI-XIN, H.-B. H.; WEI, S.-H. Clinical characteristics of nonglaucomatous optic disc cupping. *Experimental and Therapeutic Medicine*, v. 1, n. 7, p. 995–999, 2014. Citado na página 294.
- ZHAO, G.; PIETIKAINEN, M. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 29, n. 6, p. 915–928, 2007. Citado na página 26.
- ZHAO, J. et al. COVID-CT-Dataset: A CT Scan Dataset about COVID-19. 2020. Citado 2 vezes nas páginas 155 e 160.
- ZHAO, J. et al. Covid-ct-dataset: a ct scan dataset about covid-19. *arXiv preprint arXiv:2003.13865*, 2020. Citado 4 vezes nas páginas 183, 191, 192 e 201.
- ZHENG, C. et al. Deep learning-based detection for covid-19 from chest ct using weak label. *medRxiv*, Cold Spring Harbor Laboratory Press, 2020. Citado 4 vezes nas páginas 191, 192, 200 e 201.
- ZHENG, J.; WU, Q. Performance modeling and analysis of the ieee 802.11 p edca mechanism for vanet. *IEEE Transactions on Vehicular Technology*, IEEE, v. 65, n. 4, p. 2673–2687, 2015. Citado 2 vezes nas páginas 426 e 427.
- ZHOU, H.; SONG, Y.; SHU, H. Using deep convolutional neural network to classify urban sounds. In: IEEE. TENCON 2017-2017 IEEE Region 10 Conference. [S.l.], 2017. p. 3089–3092. Citado na página 599.
- ZHU, C.; BICHOT, C.-E.; CHEN, L. Visual object recognition using daisy descriptor. In: IEEE. 2011 IEEE International Conference on Multimedia and Expo. [S.l.], 2011. p. 1–6. Citado 2 vezes nas páginas 270 e 287.
- ZHU KUN XIE, K. O. Y.; YANG, H. Hurricane evacuation modeling using behavior models and scenario-driven agent-based simulations. *ScienceDirect*, 2018. Citado na página 556.
- Zhu, R.; Zhang, R.; Xue, D. Lesion detection of endoscopy images based on convolutional neural network features. In: 2015 8th International Congress on Image and Signal Processing (CISP). [S.l.: s.n.], 2015. p. 372–376. Citado 2 vezes nas páginas 150 e 157.
- ZHU, X. et al. An integrated music recommendation system. *IEEE Transactions on Consumer Electronics*, IEEE, v. 52, n. 3, p. 917–925, 2006. Citado na página 644.
- ZILLY, J.; BUHMANN, J. M.; MAHAPATRA, D. Glaucoma detection using entropy sampling and ensemble learning for automatic optic cup and disc segmentation. *Computerized Medical Imaging and Graphics*, Elsevier, v. 55, p. 28–41, 2017. Citado na página 53.
- ZIPSER, D.; ANDERSEN, R. A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, v. 331, n. 6158, p. 679–684, February 1988. ISSN 0028-0836. Disponível em: <https://doi.org/10.1038/331679a0>. Citado na página 152.
- ZOU, K. et al. Statistical validation of image segmentation quality based on a spatial overlap index. *Academic radiology*, 2004. Citado na página 302.
- ZUIDERVELD, K. Contrast limited adaptive histogram equalization. In: _____. *Graphics Gems IV*. USA: Academic Press Professional, Inc., 1994. p. 474–485. ISBN 0123361559. Citado na página 156.
- ZVAREVASHE, K.; OLUGBARA, O. O. A framework for sentiment analysis with opinion mining of hotel reviews. In: IEEE. 2018 Conference on Information Communications Technology and Society (ICTAS). [S.l.], 2018. p. 1–4. Citado na página 665.