

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Esdras de Lima Chaves

**Deteccção de câncer de mama por meio de
imagens infravermelhas utilizando Redes
Neurais Convolucionais**

Uberlândia, Brasil

2019

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Esdras de Lima Chaves

**Detecção de câncer de mama por meio de imagens
infravermelhas utilizando Redes Neurais Convolucionais**

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, Minas Gerais, como
requisito exigido parcial à obtenção do grau
de Bacharel em Ciência da Computação.

Orientador: Marcelo Keese Albertini

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2019

Esdras de Lima Chaves

Detecção de câncer de mama por meio de imagens infravermelhas utilizando Redes Neurais Convolucionais

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 10 de julho de 2019:

Marcelo Keese Albertini
Orientador

Henrique Coelho Fernandes

Jefferson Rodrigo de Souza

Uberlândia, Brasil
2019

Agradecimentos

Aos meus familiares e aos mitos da 55.

Resumo

O câncer de mama é responsável pelo maior número de mortes de mulheres no mundo todo. A detecção precoce da doença é essencial para o aumento das chances de tratamento e cura das pacientes. A termografia infravermelha tem se manifestado como uma técnica promissora para o diagnóstico da doença devido ao baixo custo, não trazer riscos às pacientes e pela possibilidade de ser feito em mulheres jovens. Esse trabalho utiliza Redes Neurais Convolucionais (algoritmos de Deep Learning) em uma base de 440 imagens de 88 pacientes distintas, com o objetivo de classificá-las em portadora ou não de câncer de mama. Foi utilizada transferência de aprendizado das arquiteturas de redes neurais convolucionais: alexnet, googlenet, resnet-18, vgg-16 e vgg-19.

Os melhores resultados foram obtidos utilizando as arquiteturas de rede neural convolucional vgg-16 com 77,5% de acurácia, 85.0% de sensibilidade e 70.0% de especificidade e com a vgg-19 com 77,5% de acurácia, 90.0% de sensibilidade e 65.0% de especificidade.

Palavras-chave: Deep Learning, Redes Neurais Convolucionais, Câncer de Mama, Termografia infravermelha, Transferência de Aprendizado.

Lista de ilustrações

Figura 1 – Modelo não linear de um neurônio - extraída de (HAYKIN, 2007).	15
Figura 2 – Rede neural multi-camadas (ou Perceptron de múltiplas camadas) - extraída de (ATTUX; ATTUX, 1969).	16
Figura 3 – Rede Neural Convolutacional e suas diferentes camadas - extraída de (VARGAS; PAES; VASCONCELOS, 2016).	17
Figura 4 – Exemplo 1 de filtro (nitidez) sobre a imagem - extraída de (CONVO- LUTIONAL..., 2018).	18
Figura 5 – Exemplo 2 de filtro (borrão) sobre a imagem - extraída de (CONVO- LUTIONAL..., 2018).	18
Figura 6 – Exemplo 3 de filtro (detecção de borda) sobre a imagem - extraída de (CONVOLUTIONAL..., 2018).	19
Figura 7 – Rede Neural Convolutacional - extraída de (CONVOLUTIONAL..., 2013).	19
Figura 8 – Representação da camada de convolução (parcial) - extraída de (USHI- ZIMA, 2017).	20
Figura 9 – Camada de pooling - extraída de (KARPATHY, 2017).	20
Figura 10 – Exemplo de matriz de confusão - extraída de (CONFUSION..., 2018).	24
Figura 11 – Paciente normal: estático frontal.	27
Figura 12 – Paciente normal: estático esquerda 90°.	27
Figura 13 – Paciente normal: estático esquerda 45°.	27
Figura 14 – Paciente normal: estático direita 45°.	27
Figura 15 – Paciente normal: estático direita 90°.	27
Figura 16 – Paciente doente: estático frontal.	28
Figura 17 – Paciente doente: estático esquerda 90°.	28
Figura 18 – Paciente doente: estático esquerda 45°.	28
Figura 19 – Paciente doente: estático direita 45°.	28
Figura 20 – Paciente doente: estático direita 90°.	28
Figura 21 – Camadas alexnet - extraída de (ALEXNET..., 2017).	31
Figura 22 – Matriz de confusão AlexNet - 30 épocas.	33
Figura 23 – Matriz de confusão GoogleNet - 20 épocas.	34
Figura 24 – Matriz de confusão ResNet-18 - 30 épocas.	35
Figura 25 – Matriz de confusão vgg-16 - 20 épocas.	36
Figura 26 – Matriz de confusão vgg-19 - 20 épocas.	37

Lista de tabelas

Tabela 1	–	Resultado AlexNet - dados de validação.	32
Tabela 2	–	Resultado AlexNet - dados de testes.	32
Tabela 3	–	Resultado GoogleNet - dados de validação.	33
Tabela 4	–	Resultado GoogleNet - dados de testes.	33
Tabela 5	–	Resultado ResNet-18 - dados de validação.	34
Tabela 6	–	Resultado ResNet-18 - dados de testes.	34
Tabela 7	–	Resultado vgg-16 - dados de validação.	35
Tabela 8	–	Resultado vgg-16 - dados de testes.	35
Tabela 9	–	Resultado vgg-19 - dados de validação.	36
Tabela 10	–	Resultado vgg-19 - dados de testes.	36

Lista de abreviaturas e siglas

BN	Batch Normalization
CNN	Convolutional Neural Network
DL	Deep Learning
FP	Falso positivo
FN	Falso negativo
IA	Inteligência Artificial
ROI	Regiões de Interesses
RNAs	Redes Neurais Artificiais
RNC	Redes Neurais Convolucionais
TIR	Termografia infravermelha
VP	Verdadeiro positivo
VN	Verdadeiro negativo

Sumário

1	INTRODUÇÃO	10
1.1	Justificativa	11
1.2	Objetivos	11
1.3	Método	11
1.4	Resultados Esperados	12
1.5	Organização do Trabalho	12
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	Conceitos Básicos	13
2.1.1	Câncer de mama	13
2.1.2	Termografia infravermelha	14
2.1.3	Redes neurais artificiais	14
2.1.3.1	Perceptron	15
2.1.3.2	Perceptron multicamadas	15
2.1.3.3	Processo de treinamento	16
2.1.3.3.1	Algoritmo backpropagation	17
2.1.4	Redes neurais convolucionais	17
2.1.4.1	Treinamento da rede neural convolucional	21
2.1.5	Transferência de aprendizado	22
2.1.6	Avaliação dos classificadores	23
2.2	Trabalhos Relacionados	24
3	MATERIAIS E MÉTODOS	26
3.1	Base de dados	26
3.2	Métodos e experimentos	30
4	TESTES E RESULTADOS	32
4.1	AlexNet	32
4.2	GoogleNet	33
4.3	ResNet-18	34
4.4	vgg-16	35
4.5	vgg-19	36
5	CONCLUSÃO	38
	REFERÊNCIAS	39

APÊNDICES	42
APÊNDICE A – CÓDIGO MATLAB UTILIZADO NOS EXPERI- MENTOS	43

1 Introdução

O câncer de mama é o tipo de câncer mais comum no Brasil e no mundo. Entre as mulheres, é o tipo mais recorrente e a segunda principal causa de morte nas Américas, além de responder por cerca de 25% dos casos novos no Brasil a cada ano (INCA, 2019). A detecção precoce do câncer de mama é fundamental para o aumento das chances de tratamento e cura com eficácia, que reduz a taxa de mortalidade causada por essa doença (LESSA; MARENGONI, 2016).

A mamografia é o exame mais comum para a detecção do câncer de mama. Se trata de uma radiografia das mamas feita por um equipamento de raios X chamado mamógrafo. A mamografia, no entanto, possui alguns pontos negativos como problemas na detecção da doença no caso de mamas densas (caso das pacientes jovens), a exposição aos raios X e o desconforto para as pacientes durante o exame.

A termografia infravermelha é uma técnica que mede a radiação infravermelha emitida por um corpo/objeto para determinar sua temperatura através de algum dispositivo, como por exemplo uma câmera infravermelha. Células cancerígenas produzem um calor acentuado, devido à sua maior atividade metabólica e vascularização em relação às células saudáveis. Este calor chega até a superfície através da condução térmica e pode ser detectado por uma câmera infravermelha.

Os diagnósticos de câncer de mama utilizando somente imagens termográficas, ainda não são aceitos pela comunidade médica, sendo utilizado como um exame secundário. Todavia, a termografia infravermelha aplicada na área médica é um campo fértil para pesquisa, visto que essa técnica não oferece danos nocivos à paciente além de ter um baixo custo. A fim de aumentar a confiabilidade desses diagnósticos, várias técnicas de análise de imagens podem e estão sendo utilizadas. As técnicas que possuem melhores resultados, são as baseadas em Inteligência Artificial (IA).

Deep Learning (DL) é uma subárea da IA e do Aprendizado de Máquina, que utiliza algoritmos inspirados na estrutura e funcionamento das células do cérebro (neurônios), chamados de Redes Neurais Artificiais (RNAs), com o intuito de ensinar ao software a reconhecer padrões em representações digitais de imagens, sons e outros tipos de dados. Redes Neurais Convolucionais (RNC) são redes que tratam especificamente da análise de imagens. Esses algoritmos têm contribuído em diversas áreas, inclusive a área médica, por apresentarem uma ótima performance e flexibilidade em ambientes não muito bem restritos (JAIN; MAO; MOHIUDDIN, 1996).

1.1 Justificativa

A detecção precoce do câncer de mama aumenta significativamente as chances de cura da paciente e evita as complicações e sequelas causadas pelos estágios mais avançados da doença, o que diminui a taxa de mortalidade em decorrência desse tipo de câncer. A proposta e desenvolvimento de técnicas mais efetivas, precisas e menos nocivas às pacientes, para o diagnóstico da doença, principalmente em seu estágio inicial, é de extrema importância.

Pensando nisso, o trabalho propõe um sistema inteligente de detecção de câncer de mama, baseado na análise de imagens termográficas e algoritmos de DL (RNC), que podem ser uma alternativa aos exames convencionais de diagnóstico da doença. Ao final, os resultados indicaram se essa abordagem é ou não uma candidata à substituição dos exames tradicionais.

Os resultados do trabalho interessa ao conjunto de pessoas alvos da doença (as mulheres), aos profissionais especialistas em oncologia, além de pesquisadores da área de processamento de imagem e DL.

1.2 Objetivos

O objetivo deste trabalho é desenvolver uma abordagem que identifique as pacientes, por meio de imagens, que apresentam a doença, ou seja, o sistema deverá ser capaz de classificar as pacientes em normais (sem doença) e com câncer. Para isso, RNC são empregadas em um banco de dados de imagens termográficas (infravermelhas) de pacientes que apresentavam a doença bem como pacientes sem a doença.

1.3 Método

O método aplicado nesse trabalho é composto dos seguintes passos:

- Foi realizado uma revisão bibliográfica sobre o uso de imagens termográficas para a detecção do câncer de mama.
- Estudo do estado da arte em redes neurais.
- Construção de um banco de dados de imagens termográficas de pacientes portadoras e não portadoras de câncer de mama.
- Utilização de RNC, para a classificação das imagens termográficas na base utilizada.

- Foi realizado experimentos em diferentes arquiteturas de CNN consideradas pela literatura e comprovadas como altamente poderosas na tarefa de classificação de imagens.

1.4 Resultados Esperados

A implementação de uma ferramenta automática, baseada em RNC para a detecção do câncer de mama em imagens termográficas. Além disso, que sua performance seja boa o suficiente para que se inicie uma discussão sobre o uso de imagens termográficas como exame primário na detecção do câncer de mama, na comunidade médica.

1.5 Organização do Trabalho

A organização do trabalho se dá da seguinte forma:

No capítulo 2 encontra-se o Referencial Teórico, que apresenta os conceitos de Câncer de mama, Termografia, Redes Neurais Artificiais (RNA), Redes Neurais Convolucionais (RNC) e métricas relacionadas à avaliação dos modelos de deep learning e técnicas de aprendizado fundamentais para o entendimento do trabalho, além da revisão da literatura que apresenta trabalhos relacionados. O capítulo 3 descreve os materiais e a metodologia teórica e prática desenvolvida nesta monografia. O capítulo 4 apresenta os testes, resultados e análise. Por fim, no capítulo 5 encontra-se a conclusão e sugestões para trabalhos futuros.

2 Fundamentação Teórica

Esse capítulo tem como objetivo expor alguns dos conceitos e técnicas utilizadas. Serão apresentados conceitos relacionados ao câncer de mama, termografia infravermelha (TIR) e as técnicas de DL: RNAs e RNC, junto com a apresentação de métricas de avaliação e técnicas de aprendizado.

2.1 Conceitos Básicos

2.1.1 Câncer de mama

Câncer é o nome dado às doenças nas quais as células começam a se comportar de maneira atípica, aumentando, crescendo e se dividindo desordenadamente (ARAÚJO, 2009). Os efeitos desse comportamento sobre o hospedeiro são agressivos, visto que o câncer, ou tumor, pode destruir tecidos adjacentes e se espalhar pelo corpo, fenômeno chamado de "metástase" (BRASIL MINISTÉRIO DA SAÚDE, 1990). Os tumores benignos são pontuais, crescem lentamente, são muito similares aos tecidos saudáveis e podem, na maioria dos casos, ser totalmente removidos por meio de cirurgia. Os tumores malignos multiplicam-se rapidamente e possuem a capacidade de provocar metástases, a cura depende do diagnóstico precoce e do tratamento adotado (PONTES, 2013).

O câncer de mama, ou carcinoma, não possui métodos simples de rastreamento, por isso faz-se necessário a realização de exames clínicos diversos, entre eles o auto-exame, mamografia ou ultrassonografia. Os cânceres de mama benigno, são conhecidos como fibroadenomas, enquanto os malignos podem ser chamados de carcinomas ductais, ou lobulares, isso depende de onde eles se originam, ductos (ou canais lactíferos) ou nos lóbulos, respectivamente. Enquanto o câncer não se espalhou pelo tecido ao redor de sua origem ou para outras partes do corpo (conhecido como câncer "invasivo"), ele é conhecido como "*in situ*".

Dentre os exames de diagnósticos existentes, o mais importante atualmente é a mamografia, realizada em mulheres com idade a partir de 35 anos. A mamografia se trata de um exame de raio-x de baixa dosagem. A imagem gerada pelo raio-x é usada para detectar anomalias de massa e calcificações. O exame não é realizado em mulheres com idade inferior aos 35 anos, pois mamas jovens possuem uma densidade maior, o que aumenta o nível de radiação absorvida pela mama, criando espaço para problemas futuros causados pela radiação, além do que, uma densidade maior, gera menos contraste no raio-x o que não garante a confiabilidade do exame. As pacientes que se encaixam na faixa etária não contemplada, são encaminhadas para o ultrassom.

2.1.2 Termografia infravermelha

A termografia infravermelha parte do princípio de que todo corpo com temperatura acima do zero absoluto emite radiação infravermelha (NG, 2009). O infravermelho é uma faixa de espectro localizado entre a luz visível e microondas, aproximadamente entre $0.74\mu\text{m}$ e 1mm que, a partir das ondas refletidas pelo material pode-se determinar a temperatura do objeto.

A termografia infravermelha consiste em obter imagens dessa radiação térmica emitida pelo corpo sendo analisado através de um equipamento (uma câmera, por exemplo).

Na medicina, a termografia mede padrões de calor da superfície da pele humana com o intuito de perceber anomalias resultantes da alteração no metabolismo das células, resposta imunológica do corpo, entre outras, que podem indicar algum tipo de doença. A termografia infravermelha também traz benefícios em relação a outros tipos de exame tradicionalmente realizados, como por exemplo, natureza não nociva (não produz radiação ionizante), por ser um processo não invasivo, não causa problemas de higiene, pois não há contato com o paciente, além de possuir vantagens em relação à portabilidade de equipamento e rapidez na obtenção e recuperação dos dados (NG, 2009).

2.1.3 Redes neurais artificiais

RNAs são modelos computacionais baseados no cérebro humano, muito utilizados atualmente em problemas de:

- Classificação de padrões: associar uma amostra de dado a um conjunto de classe previamente definido.
- Aproximação de funções.
- Problemas relacionados à variantes no tempo.
- Problemas relacionados à reconhecimento de fala, visão computacional e processamento de texto.

A idéia de que os neurônios (como unidades processadoras de informação) são constituintes estruturais do cérebro (CAJAL, 1991) foi a grande inspiração para o desenvolvimento das redes neurais artificiais. As redes neurais artificiais são compostas por um conjunto de neurônios dispostos em camadas, conectados pelo que chamamos de sinapses, ou elos de conexão. (HAYKIN, 2007). Os neurônios recebem sinais de entrada, processam os dados e geram um sinal de saída. De acordo com (HAYKIN, 2007), as redes neurais devem ser capazes de serem não-lineares, adaptáveis, tolerante a falhas e acima de tudo, conseguirem generalizar entradas que não estavam em seu treinamento.

2.1.3.1 Perceptron

A Figura 1, apresenta o modelo de um neurônio de McCulloch e Pitts que é a arquitetura da rede neural mais simples, o Perceptron.

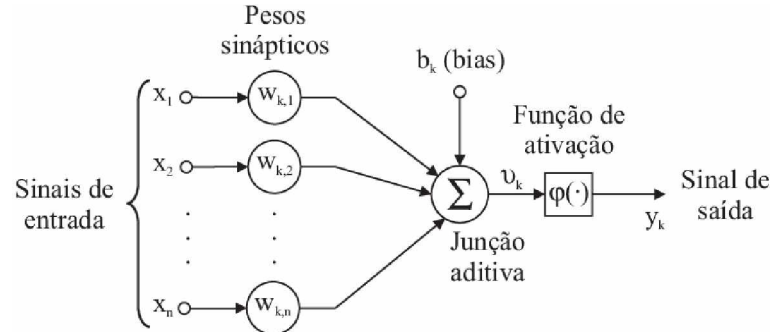


Figura 1 – Modelo não linear de um neurônio - extraída de (HAYKIN, 2007).

Podemos identificar os três elementos básicos do neurônio:

- As sinapses, cada uma caracterizada por um peso ou uma força. Vale notar que um sinal de entrada x_m na entrada da sinapse m conectada ao neurônio k é multiplicada pelo peso sináptico w_{km} .
- O somador, que soma os sinais de entrada ponderados pelas sinapses do neurônio (combinador linear).
- A função de ativação (ou função restritiva) que delimita o intervalo de saída do neurônio.
- Outro elemento a ser considerado, apesar de não ser necessário em todos os casos, é o *bias*, um *threshold*, que define qual o valor do sinal agregado necessário para que o neurônio seja ativado (tenha uma saída).

O Perceptron é uma RNA utilizada para classificar padrões linearmente separáveis, separando a saída em apenas duas classes em lados opostos de um hiperplano. (OLIVEIRA, 2018)

2.1.3.2 Perceptron multicamadas

Um perceptron multicamadas é uma arquitetura de rede na qual os neurônios são organizados em duas ou mais camadas: a camada de entrada (*input layer*), cuja função é somente propagar as entradas da rede para todos os neurônios da camada seguinte (sem nenhuma modificação), a camada de saída (*output layer*) que possui os neurônios processadores vistos na Figura 1. Há também uma terceira classe de camada, as camadas ocultas (*hidden layers*) que se encontram entre a camada de entrada e saída, como pode

ser visto na Figura 2, onde cada neurônio de uma certa camada se conecta com todos os outros da próxima camada, que pode resolver problemas muito mais complexos que um simples Perceptron (que só resolve problemas linearmente separáveis). (CASTRO; CASTRO, 2001).

RNA *feedforward* são redes neurais acíclicas, ou seja, todas as conexões entre os neurônios não formam ciclos. As informações nesse tipo de rede fluem da camada de entrada (*input layers*), para as camadas ocultas (*hidden layers*) e finalmente para a camada de saída (*output layer*).

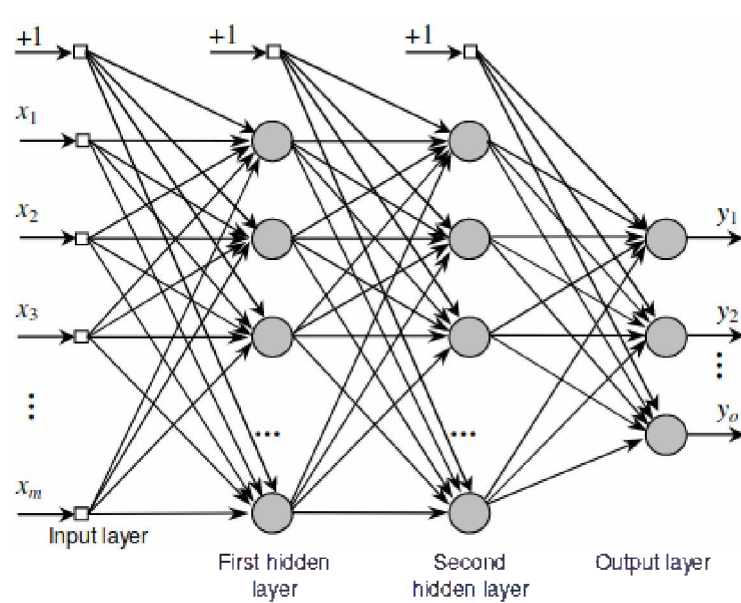


Figura 2 – Rede neural multi-camadas (ou Perceptron de múltiplas camadas) - extraída de (ATTUX; ATTUX, 1969).

2.1.3.3 Processo de treinamento

O treinamento da rede neural, também conhecido como a fase de aprendizado, é um processo iterativo de ajustes dos seus pesos sinápticos, que define a habilidade da rede em aprender e melhorar seu desempenho. A aprendizagem supervisionada é um dos paradigmas utilizados, onde um conjunto de dados (dados de treinamento) previamente rotulados com sua saída desejada são apresentados a rede neural. É definido um *signal de erro* como sendo a diferença entre a resposta desejada e a resposta observada na saída da rede neural. O *signal de erro* é utilizado no ajustes dos pesos da rede e tem como objetivo final minimizar a diferença entre a saída esperada e a observada pela rede (erro) fazendo com que a rede esteja apta a reproduzir os resultados apresentados pelos dados de treinamento. A fase de treinamento para após um valor pré-estabelecido do erro ou de iterações for atingido (HAYKIN, 2007) (LECUN; BENGIO; HINTON, 2015). Existem diversos algoritmos de propagação de erro, porém o mais utilizado no contexto de redes

neurais artificiais profundas é o *backpropagation*, baseado nas noções de vetor gradiente e a regra da cadeia.

2.1.3.3.1 Algoritmo backpropagation

O *backpropagation* é um algoritmo de propagação de erro, ou seja, ele utiliza o erro gerado pela saída da rede para nortear o ajuste dos pesos das sinapses com o objetivo de minimizar o erro da rede nas próximas iterações. O gradiente (ou vetor gradiente) expressa a relação entre duas variáveis, por exemplo: o erro y produzido pela rede, e o valor de um peso x . O algoritmo *backpropagation* calcula o gradiente descendente da rede neural (um vetor de derivadas parciais) de uma função objetivo com respeito aos parâmetros em uma rede neural (LE et al., 2015), esse gradiente é então utilizado junto com a taxa de aprendizagem para ajustar os pesos da rede, dado que ele representa o quão forte foi a influência de um peso específico no erro cometido pela rede.

2.1.4 Redes neurais convolucionais

Uma RNC é uma arquitetura de rede neural artificial, variante do perceptron de múltiplas camadas (multicamadas) que, assim como as redes neurais já citadas anteriormente, são baseadas em um processo biológico: o processamento de dados visuais. As redes neurais convolucionais são comumente usadas em aplicações de detecção, classificação e reconhecimento em imagens e vídeos. A RNC é dividida em alguns estágios principais, como pode ser visto na Figura 3.

Os primeiros estágios são constituídos de dois tipos principais de camadas: as de convolução e as de pooling. Enquanto os próximos estágios são compostos pelas camadas completamente conectadas.

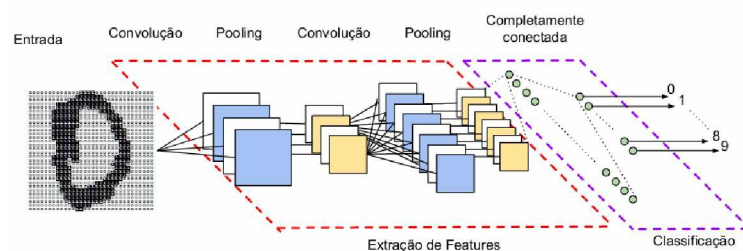


Figura 3 – Rede Neural Convolucional e suas diferentes camadas - extraída de (VARGAS; PAES; VASCONCELOS, 2016).

As primeiras camadas nas quais os dados de entrada passam são as ditas de convolução, responsáveis por extrair características dessas imagens de entrada, em outras palavras, gerar o que é chamado de *feature map*. Os neurônios dessas camadas são responsáveis por fazer o mesmo que alguns processos tradicionais de visão computacional,

aplicar filtros (VARGAS; PAES; VASCONCELOS, 2016). O resultado da aplicação desses filtros pode ser visto nas Figuras 4, 5 e 6. O neurônio olha para uma janela de *pixels* da imagem da camada anterior, através das conexões sinápticas e seus pesos, que representam o filtro (também chamados de *kernel* ou máscara). Os neurônios de uma mesma camada aplicam o mesmo filtro na sua janela de *pixels*, o que faz necessário que os pesos dos neurônios de uma mesma camada sejam compartilhados durante a fase de treinamento (USHIZIMA, 2017). É importante notar que o processamento da imagem pelos neurônios de uma camada de convolução funciona como uma janela deslizante, onde cada neurônio é responsável por uma janela e o deslocamento de uma janela para outra é definido por um *offset*, conhecido como *stride*.

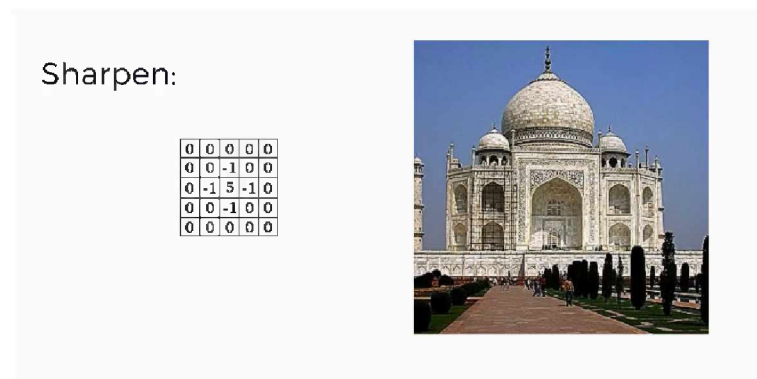


Figura 4 – Exemplo 1 de filtro (nitidez) sobre a imagem - extraída de (CONVOLUTIONAL..., 2018).

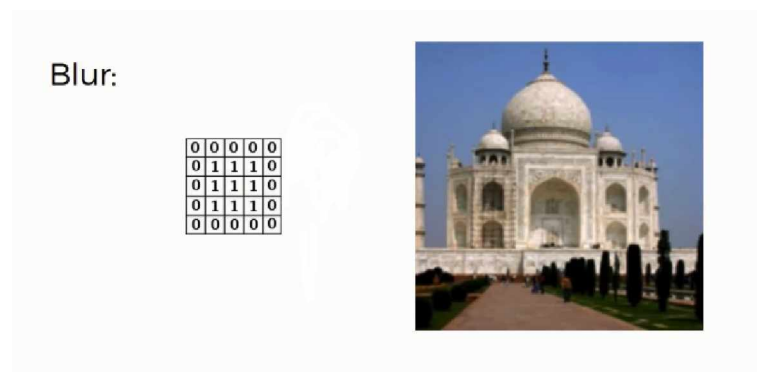


Figura 5 – Exemplo 2 de filtro (borrão) sobre a imagem - extraída de (CONVOLUTIONAL..., 2018).

É comum o uso de mais de um filtro por camada de convolução, ou seja, existe mais de um neurônio responsável por conectar uma mesma região da imagem de entrada (é possível ver o resultado disso observando o resultado de *C1* na Figura 7). A utilização de mais de um filtro resulta em um conjunto de *feature maps*, não mais uma única. Os mapas gerados pelos diversos filtros da camada convolucional são empilhados, formando

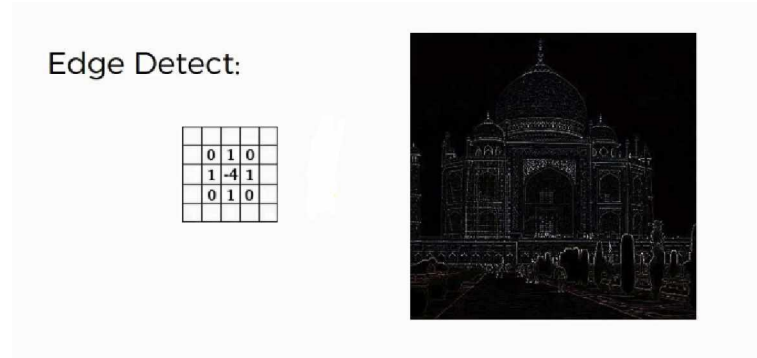


Figura 6 – Exemplo 3 de filtro (detecção de borda) sobre a imagem - extraída de (CONVOLUTIONAL..., 2018).

um tensor cuja profundidade é igual ao número de filtros. Esse tensor será oferecido como entrada para a próxima camada (PONTI; COSTA, 2018).

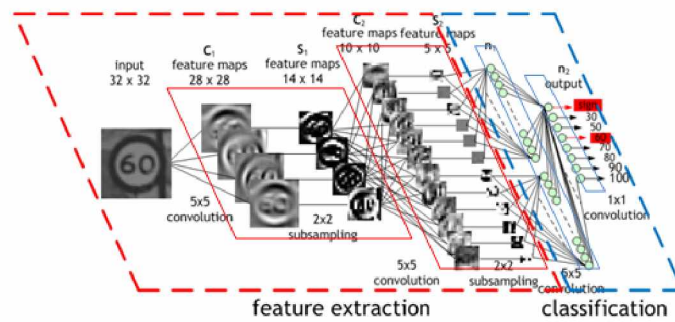


Figura 7 – Rede Neural Convolutacional - extraída de (CONVOLUTIONAL..., 2013).

CNNs do estado da arte utilizam *Batch normalization* (BN) que ocorre depois da camada de convolução. Seu objetivo principal é normalizar os mapas de características (*feature maps*), mantendo a ativação média próxima à 0 e o desvio padrão próximo à 1. Essa técnica acelera o treinamento da rede e reduz a sensibilidade à inicialização da rede. Os filtros de cada camada não são definidos, o processo de aprendizado é responsável por ajustar os pesos dos neurônios, caracterizando assim os filtros. Vale comentar que, a partir de observações após o treinamento, os filtros nas camadas iniciais são mais genéricos que os das camadas mais profundas, que trazem *features* mais complexas.

Após as convoluções, assim como nos perceptron de múltiplas camadas, uma função de ativação é aplicada, geralmente a função *ReLU* (NWANKPA et al., 2018). Na Figura 8 podemos ver um exemplo da aplicação da camada de convolução, o conceito de janela e a aplicação do *pooling* que será explicado à seguir.

Após as camadas de convolução, existem uma camada de *pooling*, que tem como objetivo diminuir a dimensão espacial dos dados de entrada, possibilitando que cama-

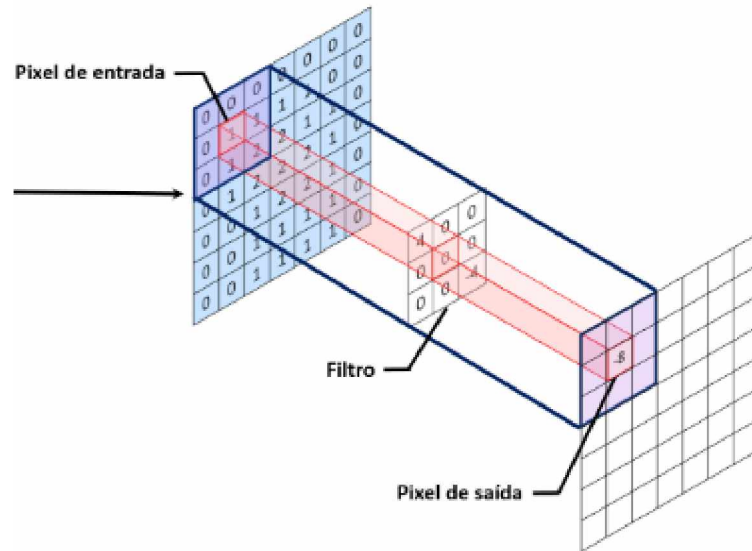


Figura 8 – Representação da camada de convolução (parcial) - extraída de (USHIZIMA, 2017).

das mais profundas possam ter mais filtros sem que o custo computacional aumente por camada, além de evitar o *overfitting* (KARPATHY, 2017). No *pooling*, uma região do mapa de atributos são substituídos por alguma métrica dessa região, por exemplo, o valor máximo (USHIZIMA, 2017). Na Figura 9 temos um exemplo da aplicação da camada de *pooling* que utiliza a métrica do máximo valor, *max pooling*. Há uma certa discussão sobre a necessidade dessa camada, alguns artigos como (SPRINGENBERG ALEXEY DOSO-VITSKIY, 2014) propõe o descarte da camada de *pooling*, enquanto outros demonstram a importância da camada no treinamento de *autoencoders*, redes generativas adversariais, entre outras.

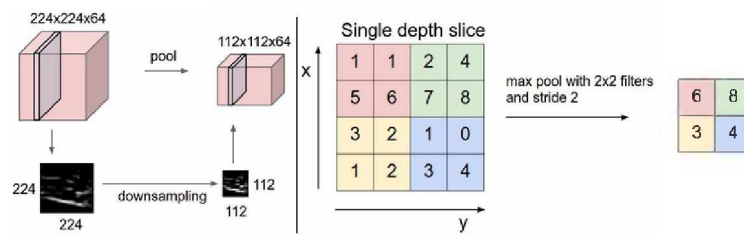


Figura 9 – Camada de pooling - extraída de (KARPATHY, 2017).

A última camada é a camada completamente conectada, a mesma apresentada na seção anterior, cujo objetivo é: a partir das características extraídas pelas camadas de convolução e pooling, classificá-las em alguma classe pré-definida.

2.1.4.1 Treinamento da rede neural convolucional

Inicialmente, todos os valores dos filtros das camadas convolucionais e os pesos das camadas totalmente conectadas são inicializados seguindo algum método ou de maneira aleatória. Em seguida, esses valores são ajustados de forma a otimizar a acurácia da classificação.

A forma de treinamento da CNN mais utilizada é por meio do *backpropagation* (HAYKIN, 2007) explicado anteriormente. O processo de treinamento ocorre da seguinte maneira:

1. Todos os filtros da rede e os pesos das camadas totalmente conectadas são inicializados. Existem diversas maneiras de inicializar os pesos (GLOROT; BENGIO, 2010).
2. A rede realiza o processo de propagação (convolução, ReLU e pooling, além da propagação nas camadas totalmente conectadas) sobre uma imagem de treinamento de entrada.
3. Uma probabilidade para cada classe de saída é calculada, assim como o erro total obtido na camada de saída.

$$Erro = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

Onde $Y_i = Probabilidade Real$ e $\hat{Y}_i = Probabilidade Obtida$

4. Os valores dos filtros e os pesos utilizam o *Backpropagation*. O ajuste é realizado na proporção que o parâmetro a ser ajustado contribuiu para o erro.
5. Os passos 2 ao 4 são repetidos para todas as imagens do conjunto de treinamento.

É chamado época o conjunto total de imagens de treinamento que passam pelos passos 2 à 4 citados anteriormente.

Existem classificações dos passos 3 e 4 dependendo de quando o cálculo do erro será feito e quando o algoritmo de aprendizagem irá atualizar os parâmetros da rede, para apresentar essas classificações. Então, faz-se necessário definir o conceito de *batch* e *iteração*. O batch é um grupo de amostras do conjunto de treinamento de tamanho fixo na qual ao final da passagem o erro é calculado e os ajustes feitos, enquanto a iteração é a quantidade de *batches* necessários para que seja completo uma época. Por exemplo, em um *dataset* de 2000 amostras, podemos dividi-lo em um *batches* de tamanho de 500, o que será necessário 4 iterações para completar uma época. Dito isso, podemos classificar o algoritmo que abrange os passos 3 e 4 em 3 classes distintas:

- **Gradiente descendente batch** - onde o tamanho do *batch* é igual ao tamanho do conjunto de treinamento, ou seja, a atualização dos pesos só ocorre após todos os dados passarem pela rede.

- **Gradiente descendente estocástico** - onde o tamanho do *batch* é igual à 1, ou seja, a atualização dos pesos ocorre para todas as amostras do conjunto de dados.
- **Gradiente descendente mini-batch** - onde o tamanho do *batch* é maior que 1 e menor que o tamanho do conjunto de dados. Tamanhos de mini-batches conhecidos e utilizados na literatura são 32, 64 e 128.

O processo de treinamento termina quando algumas das seguintes condições são satisfeitas: A média do erro obtida pela rede na época atual seja inferior a um limiar ou o número de épocas seja atingido. O término do treinamento indica que a rede (com os valores de pesos e filtros calibrados) está apta para classificar as imagens de um conjunto de treinamento de maneira ótima.

O processo de treinamento somente altera os valores dos filtros e pesos das camadas totalmente conectadas. O tamanho do filtro, o *stride* da convolução (conhecido como **hyperparâmetros**), o padding e o número de camadas não são ajustados, esses são definidos anteriormente ao passo 1 do fluxo citado anteriormente, sendo um dos grandes desafios da área.

2.1.5 Transferência de aprendizado

Transferência de aprendizado é a técnica utilizada em Aprendizado de Máquina na qual um modelo previamente treinado em uma tarefa passa a ser utilizado em outra tarefa relacionada (TORREY; SHAVLIK, 2010). A utilização da transferência de aprendizado é muito popular em *deep learning* visto que é necessário uma grande quantidade de recursos computacionais e uma quantidade de dados enorme para treinar esse tipo de modelo. Redes neurais convolucionais modernas levam semanas para treinar em múltiplas GPUs.

Para a utilização da transferência de aprendizado existem várias abordagens, uma das principais é utilizar redes neurais convolucionais como extratores de características fixo. Uma rede pré-treinada em um banco de dados com milhares (ou até milhões) de imagens é utilizada, remove-se a última camada completamente conectada (utilizada para a classificação das características nas classes selecionadas), substitui-se por uma camada que classifique nas classes desejadas para o novo conjunto de dados e é realizado o treinamento. A rede neural convolucional funcionará como o extrator de características, uma vez que essas características são adquiridas, basta classificá-las na rede completamente conectada que então classifica nas classes desejadas por conta da substituição da última camada (TRANSFER. . . , 2015).

2.1.6 Avaliação dos classificadores

Existem diversas métricas utilizadas para a avaliação dos classificadores, as mais comuns e as utilizadas nessa pesquisa são: acurácia, sensibilidade e especificidade.

Para entender o significado dessas métricas, é necessário definir os conceitos por trás dos seus cálculos, são eles de: verdadeiro positivo (VP), falso positivo (FP), verdadeiro negativo (VN) e falso negativo (FN).

No contexto das imagens médicas, as seguintes definições são utilizadas: Verdadeiro positivo: Número de amostras rotuladas como doentes que são identificadas como doentes. Falso positivo: Número de amostras rotuladas como saudáveis que são identificadas como doentes. Verdadeiro negativo: Número de amostras rotuladas como saudáveis que são identificadas como saudáveis. Falso negativo: Número de amostras rotuladas como doentes que são identificadas como saudáveis. (BARATLOO et al., 2015)

As métricas citadas são extraídas a partir dos conceitos anteriores, como segue:

- **Acurácia:** É a habilidade do modelo diferenciar a imagem nas classes saudáveis e doente corretamente, é calculada a partir das proporções de verdadeiros positivos e verdadeiros negativos sobre todos os casos avaliados.

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$

- **Sensibilidade:** É a habilidade de determinar as imagens rotuladas como doente de forma correta

$$Sensibilidade = \frac{VP}{VP + FN}$$

- **Especificidade:** É a habilidade de determinar as imagens rotuladas como saudáveis de forma correta

$$Especificidade = \frac{VN}{VN + FP}$$

As métricas citadas anteriormente podem ser representadas visualmente pela *Matriz de confusão*, também conhecida como matriz de erro. Na matriz de confusão, de tamanho $n \times n$, onde n é o número de classes diferentes, cada linha representa as instâncias da classe prevista enquanto as colunas representam a classe em si (ou vice-versa). (VISA et al., 2011).

No exemplo da Figura 10, temos o exemplo de que das 447 classificações como sendo classe 1, em 99.3% das vezes a predição estava correta (444 vezes) e em 0.7% estavam erradas. Para a classe 2, das 252 amostras classificadas como pertencentes da classe 2, em 94.4% ele acerta e 5.6% ele erra. Também existe a informação da acurácia localizada na última linha e na última coluna: 97.6%. A sensibilidade e a especificidade podem ser visualizadas na última linha da matriz de confusão, com os valores de 96,9%

e 98,8%, respectivamente. Podemos também extrair os valores dos VP, FP, FN, VN da matriz, são eles: 444, 3, 14 e 238, respectivamente.

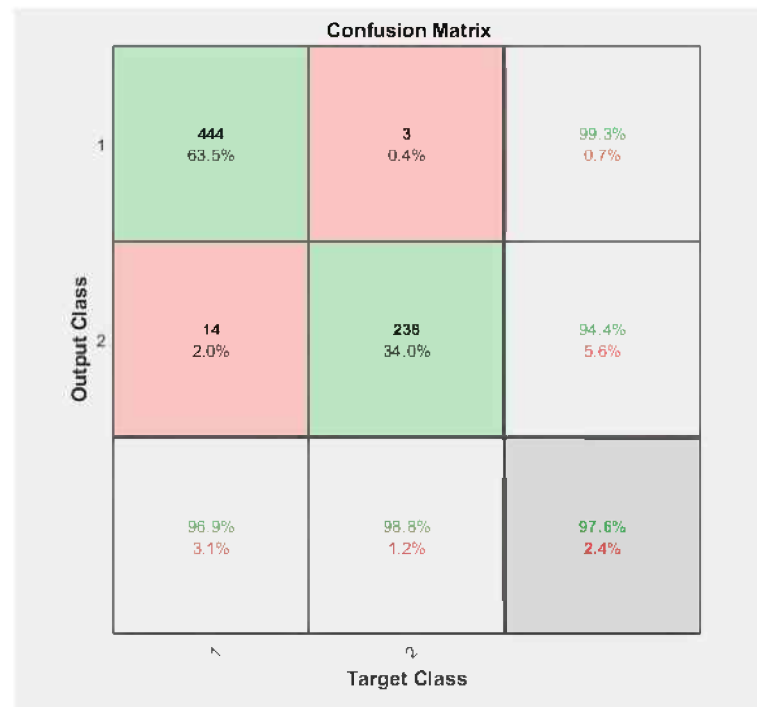


Figura 10 – Exemplo de matriz de confusão - extraída de (CONFUSION..., 2018).

2.2 Trabalhos Relacionados

Existem alguns trabalhos na mesma linha de detecção de câncer de mama utilizando imagens termográficas.

Em (GONÇALVES, 2017) foi utilizado imagens termográficas frontais com os braços levantados de 70 pacientes. As pacientes foram classificadas em três classes distintas: normais (sem alteração), com patologia benigna (nódulos benignos) e com patologia maligna (tumores). A distribuição das pacientes nessas classes se deram do seguinte modo: 21 pacientes sem alterações, 23 com alterações benignas e 26 com alterações malignas. Foi realizado o processo de segmentação das imagens seguido da separação das regiões de interesse (ROI), as mamas, utilizando o algoritmo Otsu e a separação manual quando necessário. Ao final desse processo obteve-se 140 imagens (duas para cada paciente). (GONÇALVES, 2017) extraiu das imagens de cada mama 17 características: média, desvio padrão, mediana, temperatura mínima, temperatura máxima, amplitude térmica, assimetria, curtose, entropia, contraste, homogeneidade, energia, momentos estatísticos (2, 3 e 4) e dimensão Box-Counting (Dimensão Fractal).

A classificação das imagens, utilizando como entrada combinações das características extraídas, foram realizadas utilizando Redes Neurais Artificiais (RNA) e *Support Vector*

Machine (SVM). Os testes em RNA foram feitos ao variar as camadas ocultas em 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 50; enquanto na SVM utilizou-se funções de Kernel Linear, quadrática, cúbica, gaussiana e valores de escalas iguais a 1.5, 5.9 e 23,23, obtendo gaussiana fina, média ou grossa; sigmoide com inclinação 1. (GONÇALVES, 2017). A pesquisa obteve como melhores resultados: 80.95% de acurácia, 83.33% de especificidade para pacientes normais, 85.71% de especificidade para pacientes com alterações benignas e 75% de sensibilidade, utilizando o SVM como classificador com a função cúbica.

Em (ANDRADE, 2017) foi realizada a classificação de pacientes como portadoras ou não de câncer, um conjunto de 65 exames de termografia infravermelha foram coletados, dos quais 32 eram saudáveis e 33 anômalos. Cada exame é composto por 20 termogramas que foram capturados com um intervalo de 15 segundos entre eles. Foi realizado processo de segmentação, separação das mamas e normalização, e finalmente a construção de uma série temporal das mamas transformando as matrizes dos 20 termogramas dos 130 casos em um vetor unidimensional. Características como comprimento da linha, energia média, desvio padrão, assimetria e curtose são extraídas das séries e representam as diferenças entre os padrões saudáveis e anômalos.

Essas características são utilizadas para treinar os classificadores SVM e o K-Star. Os resultados obtidos foram: 95,8% de acurácia, 93,6% de sensibilidade, 95,9% de especificidade para o K-star e 91% de acurácia, 85% de sensibilidade e 91% de especificidade para o SVM.

(RESMINI et al., 2012) conseguiram classificar pacientes sem a patologia e com patologia com uma acurácia de 88,23% utilizando uma SVM, uma acurácia de 88,23% utilizando *Naïve Bayes* e uma acurácia de 88,23% utilizando o classificador IbK. Para alcançar esses resultados, os autores utilizaram como entrada dos classificadores SVM e IbK 721 características de 10 imagens térmicas de voluntárias sem patologia e 24 imagens térmicas de voluntárias com patologia; e para o *Naïve Bayes*, extraíram 356 características dessa imagens, provenientes do projeto “Processamento e análise de imagens aplicadas à mastologia”, que foi elaborado em uma parceria entre o Instituto de Computação da Universidade Federal Fluminense (UFF) e o Departamento de Engenharia Mecânica da Universidade Federal de Pernambuco (UFPE).

(ACHARYA et al., 2012) realizaram um experimento interessante, coletando 50 imagens de termografia infravermelha do banco de dados do Hospital de Singapura, rotuladas em normais (25 imagens) e cancerígenas (25 imagens). Características de textura foram extraídas de uma matriz de co-ocorrência e de *run-length*. Essas características foram aplicadas em uma *Support Vector Machine* que as classificou em ocorrências malignas ou benignas do câncer. Como resultado eles obtiveram 88.10% de acurácia, 85.71% de sensibilidade e 90.48% de especificidade.

3 Materiais e Métodos

3.1 Base de dados

A base de dados utilizada possui imagens de 88 pacientes portadoras e não portadoras de câncer de mama, adquiridas no Banco de dados de imagens mastológicas (SILVA et al., 2014) (OLIVEIRA, 2013) dos alunos e pesquisadores do departamento de Ciência da Computação (IC/UFF). (SILVA et al., 2014). A coleta foi realizada com a câmera FLIR SC620 Thermal Imaging InfraRedCamera que produz imagens de dimensão 640 x 480 pixels onde cada pixel representa um valor de temperatura. As imagens coletadas foram realizadas com a paciente em 5 poses diferentes: estático frontal com os braços levantados, estático lateral esquerdo à 45° com braços levantados, estático lateral esquerdo à 90° com braços levantados, estático lateral direito à 45° com braços levantados, estático lateral direito à 90° com braços levantados. As 5 imagens das 88 pacientes totalizam 440 imagens distintas

As imagens rotuladas de acordo com sua classificação: normais (sem alterações) e com patologia (seja ela benigna ou maligna). As imagens são rotuladas por médicos especialistas utilizando exames convencionais, como: mamografia, biópsia e ressonância magnética.

A divisão das pacientes e suas imagens nas duas classes (normais e com patologia) foram feitas de maneira igual, portanto existem 44 pacientes e 220 imagens rotuladas como normais e o mesmo com pacientes com patologia.

As Figuras 11, 12, 13, 14, 15, retiradas da base de dados, ilustram as amostras de uma paciente normal em todos os 5 ângulos, enquanto as Figuras 16, 17, 18, 19, 20, retiradas da base de dados, ilustram as amostras de uma paciente doente em todos os 5 ângulos.

As imagens foram separadas em dois conjuntos distintos, o conjunto utilizado no processo de treinamento e validação e o conjunto utilizado nos testes posteriores, as proporções da divisão foram:

- Conjunto de treinamento e validação: 200 imagens de pacientes saudáveis e 200 de pacientes doentes (divididos em 80% para treinamento e 20% para validação).
- Conjunto de teste: 20 imagens de pacientes saudáveis e 20 de pacientes doentes.

O conjunto de teste tem como o objetivo validar o modelo (CNN) treinado com dados ainda não utilizados no treinamento, ou seja, nunca visto pela rede, o que nos indica o

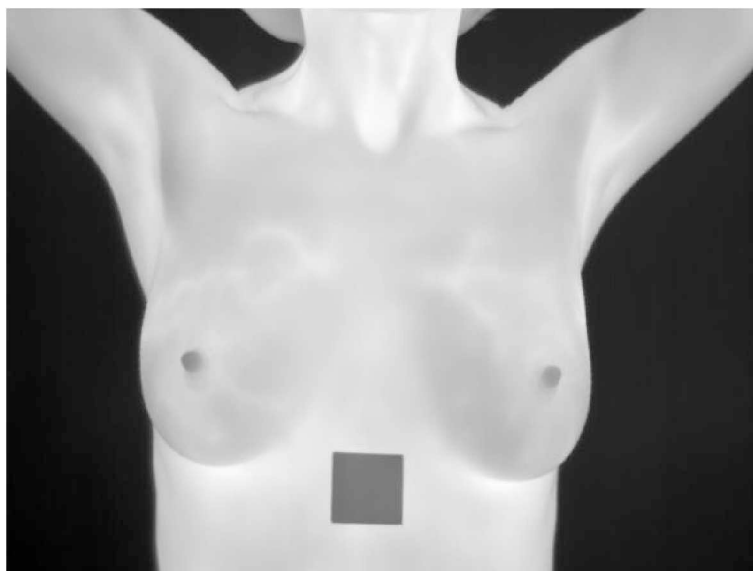


Figura 11 – Paciente normal: estático frontal.



Figura 12 – Paciente normal: estático esquerda 90°.

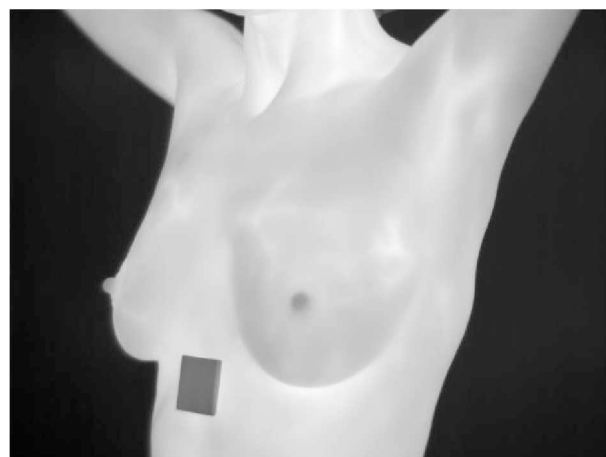


Figura 13 – Paciente normal: estático esquerda 45°.

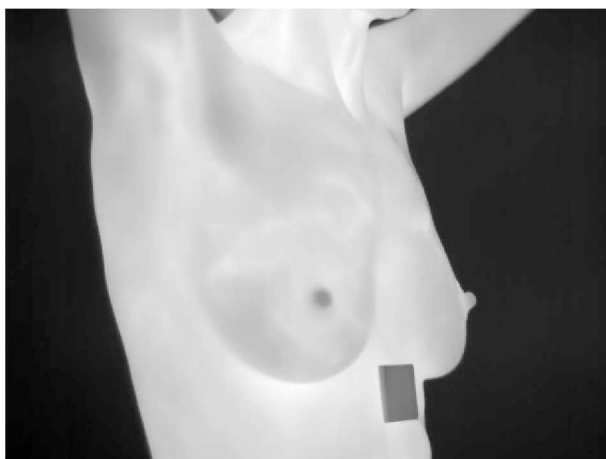


Figura 14 – Paciente normal: estático direita 45°.

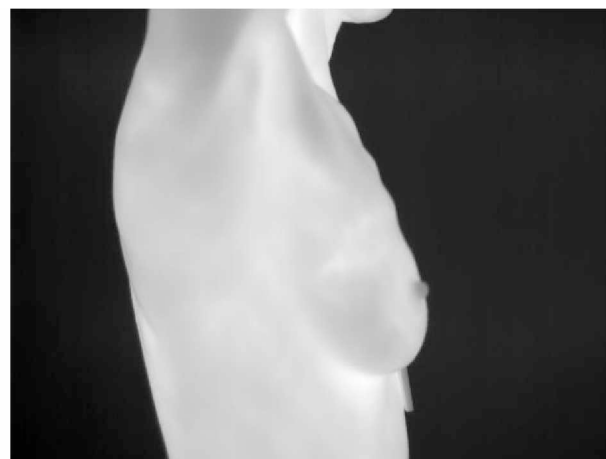


Figura 15 – Paciente normal: estático direita 90°.



Figura 16 – Paciente doente: estático frontal.

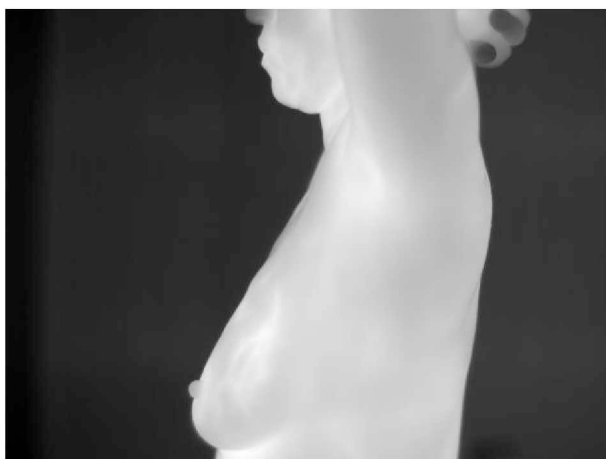


Figura 17 – Paciente doente: estático esquerda 90°.



Figura 18 – Paciente doente: estático esquerda 45°.



Figura 19 – Paciente doente: estático direita 45°.

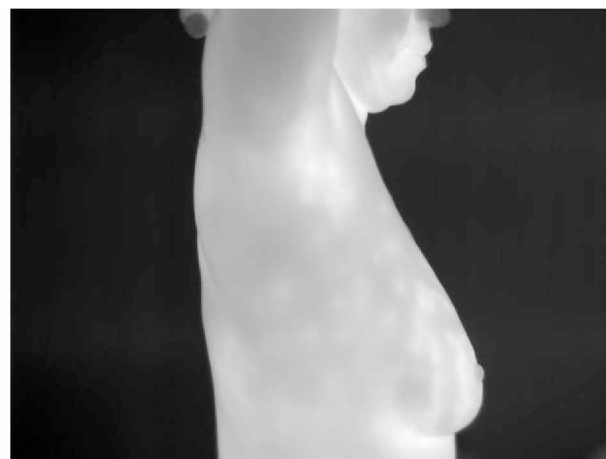


Figura 20 – Paciente doente: estático direita 90°.

real desempenho das classificações da rede para imagens genéricas, enquanto o conjunto de treinamento é utilizado no processo de aprendizado do modelo.

3.2 Métodos e experimentos

Toda a implementação foi feita no ambiente e com a linguagem Matlab, também foi utilizado as *ToolBoxes* de *Deep Learning* e Visão computacional (apêndice A).

Neste trabalho foi utilizado arquiteturas de redes neurais convolucionais já conhecidas e com desempenho comprovado para realizar a classificação das nossas imagens, são elas: alexnet (possui 25 camadas, ilustrado na Figura 21), googlenet (possui 144 camadas), resnet-18 (possui 72 camadas), vgg-16 (possui 41 camadas), vgg-19 (possui 47 camadas).

Em modelos de *deep learning* é necessário a utilização de uma grande quantidade de dados no treinamento. Devido às poucas amostras da base de dados utilizada (440) optou-se por utilizar modelos pré-treinados que nos oferece uma rica gama de representação de características. O processo de treinamento é reduzido à substituição e ao treinamento das camadas completamente conectadas da rede fornecida para que ela consiga classificar as imagens da base de dados. As camadas convolucionais também são treinadas no processo, porém com uma taxa de aprendizado muito menor que as camadas conectadas para não perder os filtros (kernel) da rede previamente treinada.

As seguintes opções de treinamento da rede foram utilizadas para todas as redes:

- Taxa de aprendizado das camadas convolucionais de $1e-4$
- Tamanho do mini-batch de 10
- Taxa de aprendizado das camadas completamente conectadas de 10
- Embaralhamento dos dados ao final de cada época
- Além disso foi aplicada a técnica de *data augmentation* nas imagens, que realiza algumas transformações, como: espelhar as imagens aleatoriamente em relação ao eixo vertical, movimentar aleatoriamente as imagens horizontalmente ou verticalmente em até 30 pixels e aumentar a imagem nos eixos horizontais e verticais aleatoriamente no intervalo de 0.9 e 1.1. Todas essas técnicas de *augmentation* tem como objetivo evitar overfitting, ou seja, evitar que o modelo memorize detalhes específicos das imagens de treinamento. As imagens também tiveram seu tamanho reajustado para o tamanho de entrada da rede em questão em tempo de execução.

As redes neurais convolucionais treinadas foram avaliadas através das métricas citadas no Seção 2.1.6, A avaliação ocorreu em dois momentos distintos, o primeiro com o conjunto de validação e o segundo com o conjunto de testes posteriores.

25x1 `Layer` array with layers:

1	'data'	Image Input	227x227x3 images with 'zerocenter' normalization
2	'conv1'	Convolution	96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]
3	'relu1'	ReLU	ReLU
4	'norm1'	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	'pool1'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
6	'conv2'	Grouped Convolution	2 groups of 128 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]
7	'relu2'	ReLU	ReLU
8	'norm2'	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10	'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
11	'relu3'	ReLU	ReLU
12	'conv4'	Grouped Convolution	2 groups of 192 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
13	'relu4'	ReLU	ReLU
14	'conv5'	Grouped Convolution	2 groups of 128 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
15	'relu5'	ReLU	ReLU
16	'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17	'fc6'	Fully Connected	4096 fully connected layer
18	'relu6'	ReLU	ReLU
19	'drop6'	Dropout	50% dropout
20	'fc7'	Fully Connected	4096 fully connected layer
21	'relu7'	ReLU	ReLU
22	'drop7'	Dropout	50% dropout
23	'fc8'	Fully Connected	1000 fully connected layer
24	'prob'	Softmax	softmax
25	'output'	Classification Output	crossentropyex with 'tench' and 999 other classes

Figura 21 – Camadas alexnet - extraída de ([ALEXNET...](#), 2017).

4 Testes e resultados

Os experimentos foram realizados nas redes citadas no capítulo anterior variando somente o número de épocas de treinamento. As épocas utilizadas foram 10, 20 e 30, número de épocas maiores que 30 começaram a apresentar indicadores de overfitting da rede, por isso não foram consideradas.

Os resultados abaixo mostram a acurácia, especificidade e sensibilidade obtidas com os dados de validação e com os dados de teste pós-treinamento. Essa métricas foram extraídas ao plotar a matriz de confusão dos resultados. As matrizes de confusão são mostradas para os melhores resultados de cada rede utilizando o conjunto de dados de teste.

Os resultados a serem considerados na conclusão da pesquisa são os experimentados nos dados de teste, visto que eles não possuem nenhuma relação com o treinamento e expressam melhor o quão genérica a rede treinada está.

4.1 AlexNet

Tabela 1 – Resultado AlexNet - dados de validação.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	81.3%	65.0%	97.5%
20	83.8%	80.0%	87.5%
30	81.3%	85.0%	77.5%

Tabela 2 – Resultado AlexNet - dados de testes.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	60.0%	30.0%	90.0%
20	72.5%	60.0%	85.0%
30	75.0%	70.0%	80.0%

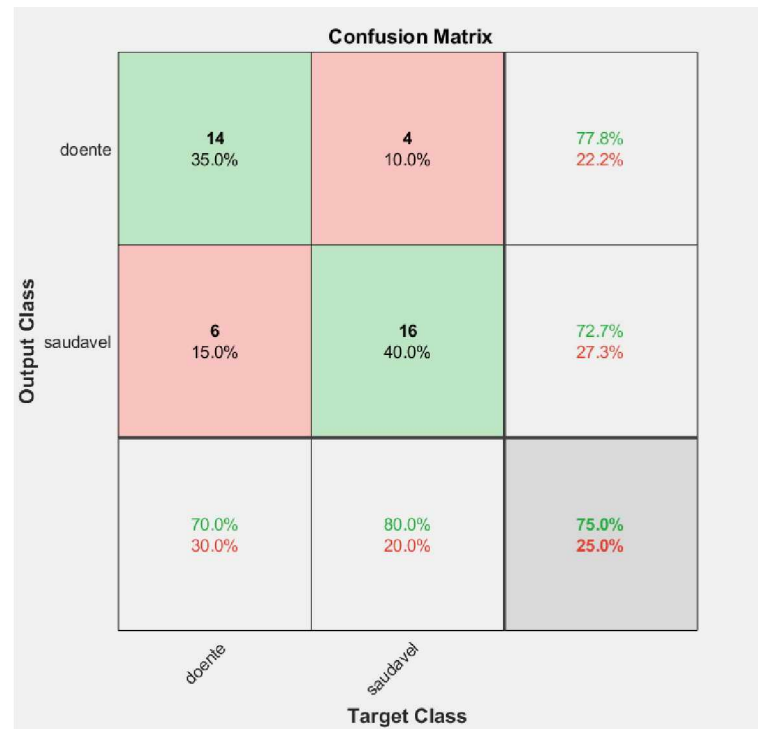


Figura 22 – Matriz de confusão AlexNet - 30 épocas.

4.2 GoogleNet

Tabela 3 – Resultado GoogleNet - dados de validação.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	80.0%	72.5%	87.5%
20	83.8%	72.5%	95.0%
30	85.0%	77.5%	92.5%

Tabela 4 – Resultado GoogleNet - dados de testes.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	62.5%	65.0%	60.0%
20	65.0%	55.0%	75.0%
30	57.5%	60.0%	55.0%

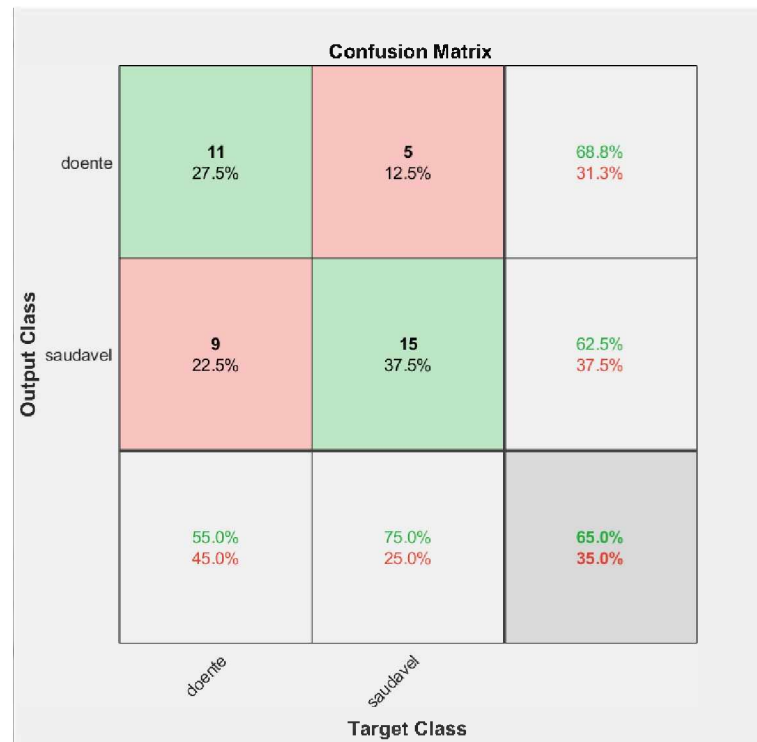


Figura 23 – Matriz de confusão GoogleNet - 20 épocas.

4.3 ResNet-18

Tabela 5 – Resultado ResNet-18 - dados de validação.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	80.0%	65.0%	95.0%
20	90.0%	92.5%	87.5%
30	90.0%	87.5%	92.5%

Tabela 6 – Resultado ResNet-18 - dados de testes.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	60.0%	50.0%	70.0%
20	72.5%	75.0%	70.0%
30	72.5%	90.0%	55.0%

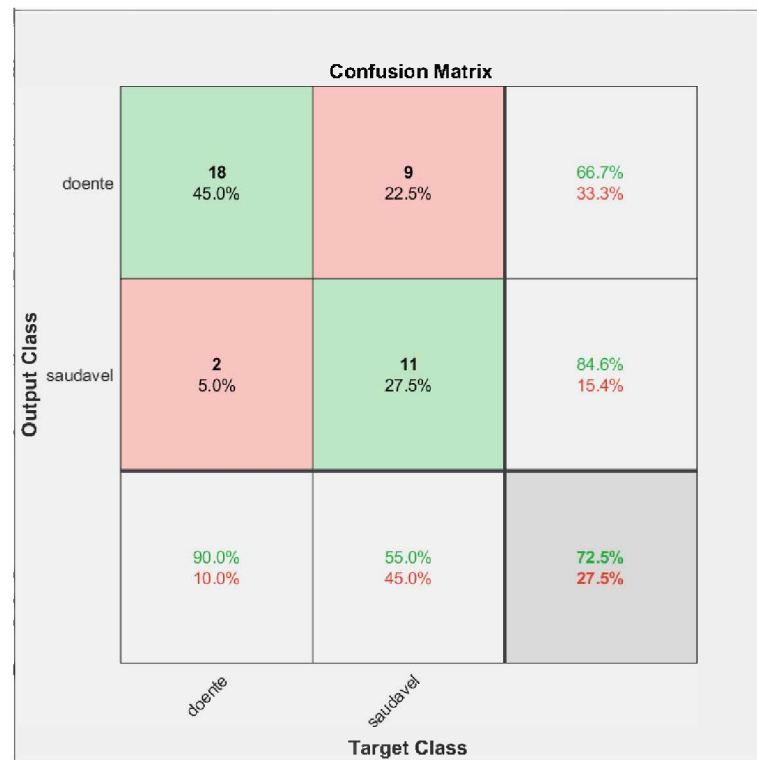


Figura 24 – Matriz de confusão ResNet-18 - 30 épocas.

4.4 vgg-16

Tabela 7 – Resultado vgg-16 - dados de validação.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	83.8%	87.5%	80.0%
20	81.3%	82.5%	80.0%
30	78.8%	60.0%	97.5%

Tabela 8 – Resultado vgg-16 - dados de testes.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	65.0%	65.0%	65.0%
20	77.5%	85.0%	70.0%
30	70.0%	60.0%	80.0%

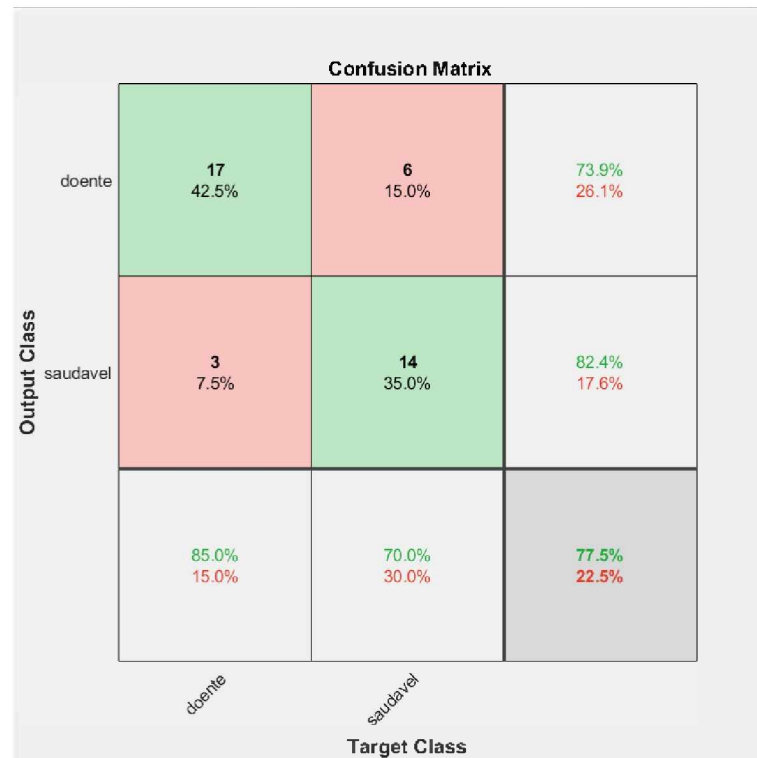


Figura 25 – Matriz de confusão vgg-16 - 20 épocas.

4.5 vgg-19

Tabela 9 – Resultado vgg-19 - dados de validação.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	72.5%	90.0%	55.0%
20	82.5%	97.5%	67.5%
30	85.0%	92.5%	77.5%

Tabela 10 – Resultado vgg-19 - dados de testes.

Número de épocas	Acurácia	Sensibilidade	Especificidade
10	72.5%	90%	55.0%
20	77.5%	90.0%	65.0%
30	75%	70.0%	80.0%

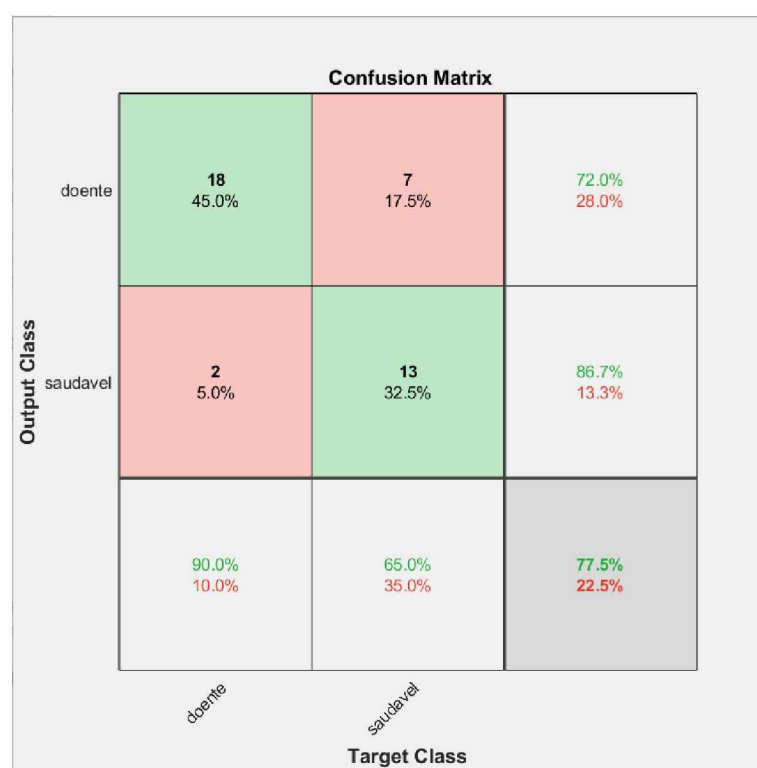


Figura 26 – Matriz de confusão vgg-19 - 20 épocas.

5 Conclusão

Neste trabalho foi utilizado técnicas de arquiteturas populares de Redes Neurais Convolucionais, como: Alexnet, GoogleNet, ResNet-18, vgg-16 e vgg-19, em imagens termográficas com o objetivo de classificá-las nas classes: portadora ou não de câncer de mama. A base de dados utilizada possui imagens de 88 pacientes em 5 posições distintas, totalizando 440 imagens, nas quais metade são rotuladas como doentes (portadoras de câncer de mama) e a outra metade em saudáveis (não portadoras), divididas entre imagens de treinamento/validação e teste, na proporção de 400/40.

Os experimentos foram realizados utilizando transferência de aprendizagem, treinando principalmente as redes completamente conectadas, alterando o número de épocas durante seu treinamento entre 10, 20 e 30.

Os melhores resultados obtidos foi utilizando a RNC VGG-16 e a VGG-19, ambas com treinamento de 20 épocas. A VGG-16 atingiu uma acurácia de 77.5%, sensibilidade de 85% e especificidade de 70%, enquanto a VGG-19 atingiu uma acurácia de 77.5%, sensibilidade de 90% e especificidade de 65%. Os resultados são animadores, por indicarem que a utilização de Redes Neurais Convolucionais em imagens termográficas para diagnóstico de câncer de mama, possuem resultados positivos e pode ser objeto de estudos mais profundos para atacar o problema; porém os resultados ainda não alcançaram valores que justifiquem a substituição dos exames de diagnóstico tradicionais. Fatores que podem ter prejudicado os resultados são: a baixa quantidade de imagens da base de dados e a não indicação da mama onde tumor (caso exista) se localiza, pois possuímos imagens de pacientes em 90° onde só é possível ver uma das mamas existindo a possibilidade daquela mama não possuir o tumor, mas ser classificada como tal

Para pesquisas futuras, propõe-se utilizar uma base de dados maior para melhorar o processo de treinamento da rede e também um estudo mais detalhado sobre a escolha de hiperparâmetros e arquiteturas de RNC utilizando algoritmo genéticos ou outras técnicas a fim de encontrar redes ideais para o problema.

Referências

- ACHARYA, U. R. et al. Thermography based breast cancer detection using texture features and support vector machine. *Journal of medical systems*, Springer, v. 36, n. 3, p. 1503–1510, 2012. Citado na página 25.
- ALEXNET Layers Matlab. 2017. <<https://www.mathworks.com/help/deeplearning/ref/alexnet.html>>. Accessed: 07-06-2019. Citado 2 vezes nas páginas 5 e 31.
- ANDRADE, F. Análise de Imagens de Termografia Dinâmica para Classificação de Alterações na Mama Usando Séries Temporais. Niterói, RJ, Brazil, 2017. Disponível em: <<http://sibgrapi2017.ic.uff.br/>>. Citado na página 25.
- ARAÚJO, M. C. de. Utilização de câmera por infravermelho para avaliação de diferentes patologias em clima tropical e uso conjunto de sistemas de banco de dados para detecção de câncer de mama. *Universidade Federal de Pernambuco*, 2009. Citado na página 13.
- ATTUX, R. R. F.; ATTUX, R. R. F. Perceptron de Múltiplas Camadas *. 1969. Disponível em: <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/ia353_1s07/topico5_07.pdf>. Citado 2 vezes nas páginas 5 e 16.
- BARATLOO, A. et al. Part 1: simple definition and calculation of accuracy, sensitivity and specificity. *Emergency*, v. 3, n. 2, p. 48–49, 2015. Citado na página 23.
- BRASIL MINISTÉRIO DA SAÚDE, S. N. d. A. à. S. C. N. d. c. a. c. p. d. o.-o. Controle do câncer: uma proposta de integração ensino serviço. Rio de Janeiro, Pro-Onco/NUTES, 1990. Citado na página 13.
- CAJAL, S. R. y. *Cajal's degeneration and regeneration of the nervous system*. [S.l.]: History of Neuroscience, 1991. Citado na página 14.
- CASTRO, F. D.; CASTRO, M. D. Redes neurais artificiais. *Porto Alegre, RS: Pontifícia Universidade Católica do Rio Grande do Sul*, 2001. Citado na página 16.
- CONFUSION Matrix. 2018. <<https://www.mathworks.com/help/deeplearning/ref/plotconfusion.html>>. Accessed: 06-06-2019. Citado 2 vezes nas páginas 5 e 24.
- CONVOLUTIONAL Neural Network (CNN). 2013. <<https://developer.nvidia.com/discover/convolutional-neural-network>>. Citado 2 vezes nas páginas 5 e 19.
- CONVOLUTIONAL Neural Networks (CNN): Step 1- Convolution Operation. 2018. <<https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-1-convolution-operation>>. Accessed: 09-05-2019. Citado 3 vezes nas páginas 5, 18 e 19.
- GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. [S.l.: s.n.], 2010. p. 249–256. Citado na página 21.
- GONÇALVES, C. B. Detecção de câncer de mama utilizando imagens termográficas. 2017. Citado 2 vezes nas páginas 24 e 25.

- HAYKIN, S. *Redes neurais: princípios e prática*. [S.l.]: Bookman Editora, 2007. Citado 5 vezes nas páginas 5, 14, 15, 16 e 21.
- INCA. *Câncer de mama*. 2019. <<https://www.inca.gov.br/tipos-de-cancer/cancer-de-mama>>. Citado na página 10.
- JAIN, A. K.; MAO, J.; MOHIUDDIN, K. Artificial neural networks: A tutorial. *Computer*, IEEE, n. 3, p. 31–44, 1996. Citado na página 10.
- KARPATHY, A. Convolutional neural networks for visual recognition. 2017. Disponível em: <<http://cs231n.github.io/convolutional-networks/>>. Citado 2 vezes nas páginas 5 e 20.
- LE, Q. V. et al. A tutorial on deep learning part 1: Nonlinear classifiers and the backpropagation algorithm. *Google Inc., Mountain View, CA*, p. 18, 2015. Citado na página 17.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. 2015. Citado na página 16.
- LESSA, V.; MARENGONI, M. Applying artificial neural network for the classification of breast cancer using infrared thermographic images. In: SPRINGER. *International Conference on Computer Vision and Graphics*. [S.l.], 2016. p. 429–438. Citado na página 10.
- NG, E.-K. A review of thermography as promising non-invasive detection modality for breast tumor. *International Journal of Thermal Sciences*, Elsevier, v. 48, n. 5, p. 849–859, 2009. Citado na página 14.
- NWANKPA, C. et al. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018. Citado na página 19.
- OLIVEIRA, G. B. de. SISTEMA DE PREDIÇÃO DE PROTEÍNAS EFETORAS BASEADO NA ANÁLISE COMPARATIVA DE ALGORITMOS DE CLASSIFICAÇÃO DE INTELIGÊNCIA ARTIFICIAL. 2018. Citado na página 15.
- OLIVEIRA, G. O. S. Desenvolvimento de um banco de dados de imagens médicas acessíveis via web com recuperação de dados baseada no conteúdo. 2013. Citado na página 26.
- PONTES, D. L. de B. *Câncer benigno e maligno*. 2013. <<https://www.einstein.br/noticias/noticia/cancer-benigno-maligno>>. Accessed: 07-07-2019. Citado na página 13.
- PONTI, M. A.; COSTA, G. B. P. da. Como funciona o deep learning. *arXiv preprint arXiv:1806.07908*, 2018. Citado na página 19.
- RESMINI, R. et al. Assistance to the Early Diagnosis of Breast Pathologies Using Thermal Imaging and Data Mining Techniques. *Annals of Computer on the Beach*, p. 305–314, 2012. Citado na página 25.
- SILVA, L. et al. A new database for breast research with infrared image. *Journal of Medical Imaging and Health Informatics*, American Scientific Publishers, v. 4, n. 1, p. 92–100, 2014. Citado na página 26.

SPRINGENBERG ALEXEY DOSOVITSKIY, T. B. M. A. R. J. T. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2014. Disponível em: <http://arxiv.org/abs/1412.6806>. Citado na página 20.

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 na página 22.

TRANSFER Learning. 2015. <http://cs231n.github.io/transfer-learning/>. Accessed: 07-06-2019. Citado na página 22.

USHIZIMA, D. M. Redes Neurais Convolucionais com Tensorflow: Teoria e Prática. p. 382–406, 2017. Citado 3 vezes nas páginas 5, 18 e 20.

VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. 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 3 vezes nas páginas 5, 17 e 18.

VISA, S. et al. Confusion matrix-based feature selection. *MAICS*, v. 710, p. 120–127, 2011. Citado na página 23.

Apêndices

APÊNDICE A – Código Matlab utilizado nos experimentos

```

1 edit(fullfile(matlabroot,'examples','nnet','main','
    findLayersToReplace.m'))
2
3 % O primeiro parametro da funcao 'imageDatastore' deve ser o
    caminho onde suas imagens de treinamento estao localizadas.
4 % Dentro do diretorio 'training' existem duas pastas, cada uma
    representando uma das classes (paciente normal ou com cancer)
    .
5 % O nome do diretorio representa a classe das imagens que ele
    contem (imagens dentro da pasta 'X', possuem o titulo 'X').
6
7
8 imds = imageDatastore('C:\Users\Esdras Chaves\Documents\TCC\
    final_dataset\training','IncludeSubfolders',true,'LabelSource
    ','foldernames');
9 [imdsTrain,imdsValidation] = splitEachLabel(imds,0.8,'randomized
    ');
10
11 % Para testar com outra rede, basta trocar 'alexnet' por:
12 % googlenet, resnet18, vgg16 ou vgg19
13 net = alexnet;
14
15 inputSize = net.Layers(1).InputSize;
16
17 if isa(net,'SeriesNetwork')
18     lgraph = layerGraph(net.Layers);
19 else
20     lgraph = layerGraph(net);
21 end
22
23 [learnableLayer,classLayer] = findLayersToReplace(lgraph);
24 [learnableLayer,classLayer]
25

```

```

26 numClasses = numel(categories(imdsTrain.Labels));
27
28 if isa(learnableLayer, 'nnet.cnn.layer.FullyConnectedLayer')
29     newLearnableLayer = fullyConnectedLayer(numClasses, ...
30         'Name', 'new_fc', ...
31         'WeightLearnRateFactor', 10, ...
32         'BiasLearnRateFactor', 10);
33
34 elseif isa(learnableLayer, 'nnet.cnn.layer.Convolution2DLayer')
35     newLearnableLayer = convolution2dLayer(1, numClasses, ...
36         'Name', 'new_conv', ...
37         'WeightLearnRateFactor', 10, ...
38         'BiasLearnRateFactor', 10);
39 end
40
41 lgraph = replaceLayer(lgraph, learnableLayer.Name,
42     newLearnableLayer);
43
44 newClassLayer = classificationLayer('Name', 'new_classoutput');
45 lgraph = replaceLayer(lgraph, classLayer.Name, newClassLayer);
46
47 layers = lgraph.Layers;
48 connections = lgraph.Connections;
49
50 layers(1:10) = freezeWeights(layers(1:10));
51 lgraph = createLgraphUsingConnections(layers, connections);
52
53 pixelRange = [-30 30];
54 scaleRange = [0.9 1.1];
55 imageAugmenter = imageDataAugmenter( ...
56     'RandXReflection', true, ...
57     'RandXTranslation', pixelRange, ...
58     'RandYTranslation', pixelRange, ...
59     'RandXScale', scaleRange, ...
60     'RandYScale', scaleRange);
61 augimdsTrain = augmentedImageDatastore(inputSize(1:2), imdsTrain,
62     ...
63     'DataAugmentation', imageAugmenter);

```

```
63 augimdsValidation = augmentedImageDatastore(inputSize(1:2),
        imdsValidation);
64 miniBatchSize = 10;
65 valFrequency = floor(numel(augimdsTrain.Files)/miniBatchSize);
66 options = trainingOptions('sgdm', ...
67     'MiniBatchSize',miniBatchSize, ...
68     'MaxEpochs',100, ...
69     'InitialLearnRate',1e-4, ...
70     'Shuffle','every-epoch', ...
71     'ValidationData',augimdsValidation, ...
72     'ValidationFrequency',valFrequency, ...
73     'Verbose',false, ...
74     'Plots','training-progress');
75
76 net = trainNetwork(augimdsTrain,lgraph,options);
77 [YPred,probs] = classify(net,augimdsValidation);
78 accuracy = mean(YPred == imdsValidation.Labels)
79
80 plotconfusion(imdsValidation.Labels,YPred)
81
82 % O primeiro parametro da funcao 'imageDatastore' deve ser o
    caminho onde suas imagens de teste estao localizadas.
83 oTestSet = imageDatastore('C:\Users\Esdras Chaves\Documents\TCC\
    final_dataset\validation','IncludeSubfolders',true,'
    LabelSource','foldernames');
84 oAugTestSet = augmentedImageDatastore(inputSize(1:2),oTestSet);
85 YTestPred = classify(net,oAugTestSet);
86 accuracy = mean(YTestPred == oTestSet.Labels)
87 plotconfusion(oTestSet.Labels,YTestPred)
```