

UNIVERSIDADE FEDERAL DO MARANHÃO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA

João Marcello Mendes Moreira

João Dallyson Sousa de Almeida

**DIAGNÓSTICO AUTOMÁTICO DE GLAUCOMA USANDO REDES EM  
CÁPSULA**

Informações do bolsista

Nome: João Marcello Mendes Moreira

Telefone: (98) 988280-0528

E-mail: joaomarcello.mm@gmail.com

Informações da Instituição/Departamento

Nome: UNIVERSIDADE FEDERAL DO MARANHÃO

Endereço: Av. dos Portugueses, 1966, Bacanga, São Luís, MA

Telefone: (98)3272-8000

E-mail: atendimento@ufma.br

Informações do(a) professor(a) orientador(a)

Nome: João Dallyson Sousa de Almeida

Telefone: (98)98256-5465

E-mail: jdallyson@nca.ufma.br

## RESUMO

O glaucoma é uma doença que degrada o nervo óptico causando perda irreversível da visão e cujos sintomas costumam aparecer somente em estágios avançados da doença. A Organização Mundial da Saúde prevê que em 2020 cerca de 76 milhões de pessoas serão afetadas pelo glaucoma. Projeções feitas para 2040 preveem que este número aumentará para cerca de 111,8 milhões de pessoas. Ainda sem cura, o principal recurso contra o glaucoma é o seu diagnóstico precoce, o qual mostra a importância dos sistemas de diagnóstico assistido por computador (CAD), visto que estes provêm uma segunda opinião ao especialista, ajudando no correto diagnóstico. Assim, este trabalho visa apresentar um método computacional capaz de detectar o glaucoma em imagens de fundo de olho utilizando redes em cápsula, um novo tipo de arquitetura de rede neural convolucional que tem como principal particularidade a presença de camadas contendo cápsulas, que são agrupamentos de neurônios que detectam características do volume de entrada assim como propriedades dessas características, como por exemplo pose, orientação e escala. O método proposto está organizado em quatro etapas: (1) Aquisição de imagens, (2) Pré-processamento, (3) Definição e treinamento da rede em cápsula, e (4) Classificação e Validação dos resultados. Os testes foram realizados em oito bases públicas de imagens de exames de glaucoma, totalizando 2086 imagens. O método proposto obteve 85,29% de acurácia, 81,77% de sensibilidade e 88,28% de especificidade.

**Palavras-chaves:** Glaucoma. Diagnóstico. Redes em Cápsula.

## SUMÁRIO

1	Introdução.....	5
2	Objetivos.....	6
2.1	Objetivo Geral.....	6
2.2	Objetivos Específicos .....	6
3	Trabalhos relacionados .....	6
4	Fundamentação Teórica.....	7
4.1	Redes Neurais Artificiais (RNA) .....	7
4.2	Redes Neurais Convolucionais (CNN).....	8
4.3	Redes em Cápsula .....	10
5	Materiais e métodos .....	11
5.1	Aquisição de Imagens.....	12
5.2	Pré-Processamento .....	13
5.4	Definição e treinamento da rede em cápsula.....	15
5.5	Classificação .....	16
5.6	Validação dos resultados .....	16
6	Resultados.....	17
7	Conclusão .....	22
	Referências .....	22

## 1 Introdução

O termo glaucoma engloba um conjunto de doenças ópticas com inúmeras causas, porém com características comuns, que formam um quadro clínico semelhante e que levam à perda total e irreversível da visão. São elas: o aumento da pressão intraocular, perdas campimétricas, aumento da escavação e atrofia do nervo óptico (Lauretti et al., 1997). Comumente assintomáticos, os portadores de glaucoma só manifestam os sintomas em estágios avançados da doença o que demonstra a importância de um diagnóstico precoce.

Uma estimativa da Organização Mundial da Saúde aponta que cerca de 76 milhões de pessoas, com idade entre 40 e 80 anos, serão afetadas pelo glaucoma em 2020. Estima-se que em 2040, o número de pessoas com glaucoma no mundo inteiro aumentará para 111,8 milhões (Tham et al, 2014).

Nos últimos anos, vem crescendo o número de pesquisas envolvendo diagnósticos assistidos por computador (CAD) que são ferramentas que realizam a detecção de anomalias em exames de forma automatizada e servem como uma segunda opinião para o especialista (Silva, 2018). Nesse contexto, surgem as redes neurais convolucionais (CNN), atualmente o estado da arte em classificação de imagens e utilizadas em vários trabalhos e em diferentes arquiteturas, entre elas VGG, ResNet e U-Net. Contudo, Sabour et al. (2017) apontou problemas em relação às arquiteturas tradicionais de *deep learning*. Segundo os autores, há perda significativa de informações por conta da organização sequencial de múltiplas camadas presentes nessas arquiteturas. Ainda, a presença das camadas de *pooling*, que reduzem a dimensionalidade do volume de entrada, descartam informações importantes que deveriam ser propagadas pela rede. Por fim, várias arquiteturas de CNN são muito dependentes das características espaciais dos dados utilizados para treinamento, fazendo com que rotações ou translações destes dados façam com que sejam classificados incorretamente. Assim, Sabour et al. (2017) propuseram uma nova arquitetura a fim de superar as limitações das arquiteturas convencionais, chamada de Capsnet (redes em cápsula), que, entre outras particularidades, possui camadas contendo cápsulas, que são agrupamentos de neurônios que simbolizam um vetor de  $n$  dimensões, e que agregam várias propriedades particulares de uma característica presente na imagem. Os autores obtiveram o estado da arte na base MNIST para reconhecimento de dígitos manuscritos.

Assim, este trabalho tem como objetivo avaliar a utilização de redes em cápsula para detecção de glaucoma em retinografias.

## **2 Objetivos**

### **2.1 Objetivo Geral**

Investigar a aplicação das redes em cápsulas para detectar a possível presença de glaucoma em retinografias.

### **2.2 Objetivos Específicos**

- Adquirir bases de imagens de retinografias de glaucoma;
- Estudar técnicas de melhoramento de imagens digitais;
- Estudar e reproduzir trabalhos que utilizam Redes em Cápsula;
- Detectar o glaucoma em retinografias;
- Avaliar a efetividade da solução proposta.

## **3 Trabalhos relacionados**

Esta seção apresenta trabalhos recentes existentes na literatura que utilizaram *deep learning* para o diagnóstico de glaucoma.

[Li et al, \(2018\)](#) utilizaram em seu trabalho uma rede neural convolucional com a arquitetura Inception-V3 com 22 camadas, com o objetivo de detectar glaucoma em retinografias. Usando 31.745 imagens para treino e 8.000 para validação da base online privada *LabelMe*, eles obtiveram acurácia de 98,6%, com sensibilidade de 95,6% e especificidade de 92,2%.

O método apresentado por [Raghavendra et al., \(2018\)](#) utiliza uma rede neural convolucional com 18 camadas treinada em 70% de uma base com 589 imagens de pacientes saudáveis e 837 de pacientes com glaucoma obtida da Faculdade de Medicina Kasturba, Índia, conseguindo 98,13% de acurácia, 98% de sensibilidade e 98,30% de especificidade.

Com o objetivo de escolher a melhor arquitetura de CNN que seja capaz de auxiliar no diagnóstico do glaucoma através de técnicas evolutivas, [Lima \(2019\)](#) utilizou a base RIM-ONE versão 2, conseguindo 96,63% de acurácia, 94,87% de sensibilidade, 98% de especificidade e 97,37% de precisão.

Utilizando a arquitetura LeNet e a base RIM-ONE versão 2 para detecção de glaucoma, [Pereira \(2017\)](#) conseguiu 88,57% de acurácia, com o desvio padrão de 1,79%. O método apresentado por [Reis \(2018\)](#) utiliza o Cup-to-Disc-Ratio (CDR), que é a razão entre os diâmetros verticais do disco óptico e de sua cavidade, a arquitetura de CNN U-Net e a base pública DRISHTI-GS1, obtendo a acurácia média de 88,7%, sensibilidade média de 94% e especificidade média de 53,3% na detecção do glaucoma.

O método desenvolvido por [Claro \(2019\)](#) teve como objetivo determinar o melhor conjunto de características para representação de imagens. Para tal, utilizou o descritor de textura GLCM em conjunto com redes neurais pré-treinadas e as bases de dados Rim-One, Hrf, Acrima, Drishti e Jsiac num total de 1675 imagens, obtendo 93,61% de acurácia.

Diferentemente dos trabalhos citados que utilizaram arquiteturas tradicionais de CNN, esta pesquisa visou a utilização de uma nova arquitetura de rede neural, chamada de redes em cápsula, explorando suas vantagens que incluem a melhora dos resultados em bases desbalanceadas e/ou com poucas amostras ([Sánchez et al, 2018](#)) .

## **4 Fundamentação Teórica**

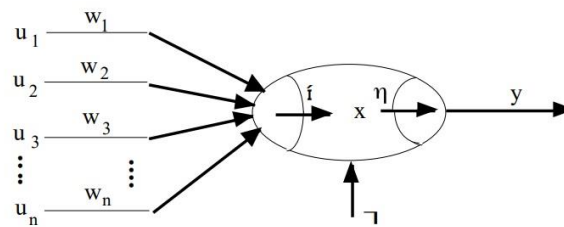
Esta seção apresenta alguns conceitos básicos para compreensão deste trabalho. A [Seção 4.1](#) faz um breve resumo sobre as redes neurais artificiais, a [Seção 4.2](#) descreve as redes neurais convolucionais e a [Seção 4.3](#) apresenta as redes em cápsula.

### **4.1 Redes Neurais Artificiais (RNA)**

Uma rede neural artificial é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que tem a propensão natural para armazenar conhecimento experimental. Assemelha-se ao cérebro em dois aspectos: o conhecimento é adquirido em um processo de aprendizado e as forças de conexão entre

neurônios (pesos sinápticos) são utilizadas para armazenar esse conhecimento (Haykin, 2001). A Figura 1 ilustra um modelo generalizado de um neurônio artificial.

**Figura 1** - Modelo de um neurônio artificial



**Fonte:** Barreto, 2002.

Neste modelo as entradas  $w_i u_i$  são combinadas usando uma função  $\Phi$ , para produzir um estado de ativação do neurônio que através da função  $\eta$  vai produzir a saída do neurônio (correspondente à frequência de descarga do neurônio biológico). Um valor auxiliar  $\theta$  (chamado de *bias*) é geralmente usado para representar uma polarização, valor abaixo do qual a saída é nula (Barreto, 2002).

Assim, uma RNA é um sistema formado por vários neurônios artificiais agrupados em camadas. O modo como essas camadas estão agrupadas e a forma de aprendizagem definem a arquitetura de uma RNA (Ferreira, 2017). Os neurônios de entrada recebem dados do exterior, e os neurônios de saída produzem a saída da rede. Os neurônios internos ficam em camadas conhecidas como *hidden layers* (camadas ocultas), e são responsáveis por, entre outras funções, permitir à rede resolver problemas categorizados como não linearmente separáveis (Barreto, 2002). A quantidade de *hidden layers* nos diz a profundidade da rede (a partir daí, surge o termo *deep learning*).

Uma arquitetura de rede neural que vêm se destacando nos últimos tempos são as redes neurais convolucionais, especializadas em classificação de imagens, e será discutida mais detalhadamente na próxima seção.

## 4.2 Redes Neurais Convolucionais (CNN)

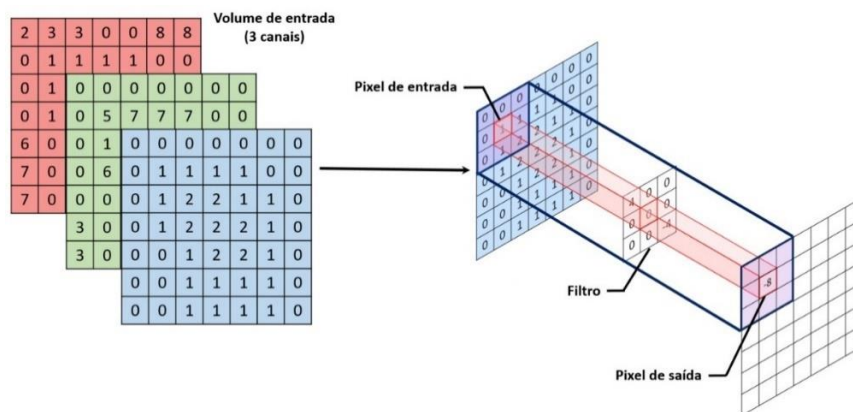
Inspiradas pelo mecanismo de percepção visual dos seres vivos, as redes neurais convolucionais são redes que usam uma arquitetura especial particularmente adaptada para classificar imagens (Goodfellow et al, 2016). Há numerosas variantes de CNNs na literatura, contudo, estas possuem componentes básicos em comum. São eles: camadas convolucionais, camadas de *pooling* e camadas totalmente conectadas (FC).



As camadas convolucionais são responsáveis pela extração de características dos dados recebidos na entrada (Lopes, 2017). É nelas que ocorre uma operação conhecida como convolução, que pode ser interpretada como o somatório da multiplicação de cada elemento da imagem, junto com seus vizinhos locais, pelos elementos da matriz que representa o filtro de convolução (Ferreira, 2017). A convolução está ilustrada na Figura 4.2.

Na convolução, o filtro desloca-se sobre a imagem com um “passo” que informa quantos pixels da imagem devem ser saltados, geralmente 1 ou 2. Os pixels da imagem correspondente com os elementos do filtro são multiplicados e, em seguida somados. Intuitivamente, a rede irá aprender filtros que são ativados quando percebem algum tipo de característica visual na imagem como bordas, cores e padrões (Ribeiro, 2018).

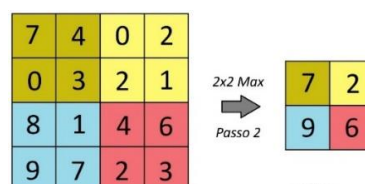
**Figura 2** - Exemplo de convolução entre um filtro 3x3 e o volume de entrada.



Fonte: Araújo, 2017.

As camadas de *pooling* são uma forma de *down-sampling* (Ferreira, 2017), ou seja, seu objetivo é reduzir o tamanho espacial das matrizes resultantes da convolução, consequentemente reduzindo a quantidade de parâmetros a serem aprendidos pela rede e contribuindo para o controle do *overfitting*. Elas calculam o máximo local de uma determinada região do mapa de atributos, reduzindo a quantidade de elementos em cada canal, mas mantendo a quantidade de canais.

**Figura 3** - Aplicação do *max pooling* em uma imagem 4x4 utilizando um filtro 2x2.



Fonte: Próprio aluno.

As camadas totalmente conectadas (FC) são geralmente utilizadas no final de uma CNN, com o objetivo de realizar o mapeamento da representação obtida pelas camadas anteriores para os rótulos de classe (Lopes, 2017). Os neurônios em uma camada FC são conectados com todos os outros neurônios da camada seguinte tal qual uma RNA tradicional.

### 4.3 Redes em Cápsula

Idealizadas por Sabour et al (2017), as redes em cápsula são redes neurais que possuem como principal singularidade a presença de camadas contendo cápsulas, que são um agrupamento de neurônios que representam um vetor que define parâmetros de instanciação (por exemplo, cor e textura) e de pose (como posição, orientação e escala) para as diferentes partes de um objeto (Moraes, 2018). Surgiram da necessidade de superar problemas comuns existentes nas arquiteturas convencionais de CNN entre os quais estão a presença das camadas de *pooling*, que descartam a informação sobre a exata localização da característica detectada o que faz a rede ser incapaz de reconhecer objetos quando estes são sujeitos a rotações ou outros tipos de (Afshar et al, 2019).

O módulo do vetor de saída de uma cápsula representa a probabilidade que a característica representada por essa cápsula esteja presente no dado de entrada. Para tal, é utilizada uma nova função de ativação chamada de “squashing” que faz com que o vetor tenha seu módulo ajustado para 0 ou 1 de modo que vetores com módulos curtos fiquem próximos de zero e vetores longos fiquem próximos de 1. A Equação 1 mostra como é calculada a função de achatamento (*squashing*).

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|} \quad (1)$$

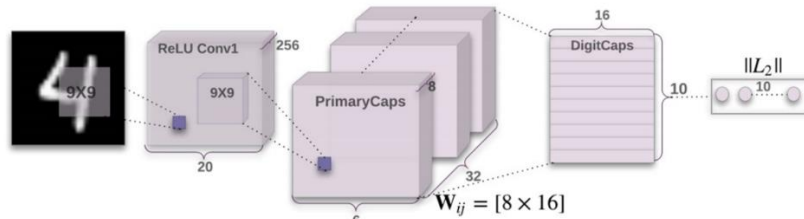
Onde  $v_j$  é a probabilidade da presença da característica da cápsula  $j$  e  $s_j$  é a soma total da entrada.

O vetor de saída de uma cápsula é enviado para a cápsula apropriada da camada seguinte em um processo chamado de “routing-by-agreement” (que substitui a necessidade das camadas de *pooling*), no qual a cápsula computa um “vetor de previsão” multiplicando sua própria saída com uma matriz de pesos. Assim, se uma cápsula de nível

baixo “concorda” com uma cápsula de nível alto, forma-se uma relação de “parte-todo” em que as cápsulas de nível baixo detectam a presença de parte de uma característica enquanto as de nível alto detectam o todo (Patrick et al, 2019).

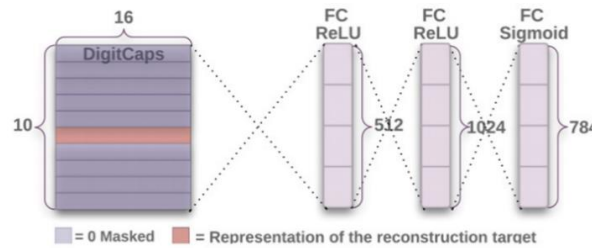
A arquitetura de rede em cápsula descrita por Sabour et al. (2017), a Capsnet, foi utilizada para o problema da identificação de dígitos manuscritos usando a base MNIST. A arquitetura possui duas partes: *encoder* e *decoder*. O *encoder* (Figura 4) possui três camadas: a primeira, Conv1, possui 256 filtros convolucionais de dimensões 9x9, passo 1 e função de ativação ReLu. Esta camada extrai características da entrada e as usa para entrada nas *PrimaryCaps*. A segunda camada (*PrimaryCaps*) é formada por 32 cápsulas 8D enquanto que na *DigitCaps* são 10 cápsulas 16D. O algoritmo “routing-by-agreement” foi utilizado entre as *PrimaryCaps* e as *DigitCaps*. O *decoder* (Figura 5) tenta reconstruir a entrada a partir das cápsulas finais, o que força a rede a preservar o máximo possível de informações da entrada através da rede. Isto funciona como um regularizador que reduz o risco de *overfitting* e ajuda na generalização para novos dados (Mobiny et al, 2018).

**Figura 4 -** Arquitetura da Capsnet (Encoder).



Fonte: Sabour, et al. (2017).

**Figura 5 -** Arquitetura da Capsnet (Decoder).



Fonte: Sabour, et al (2017).

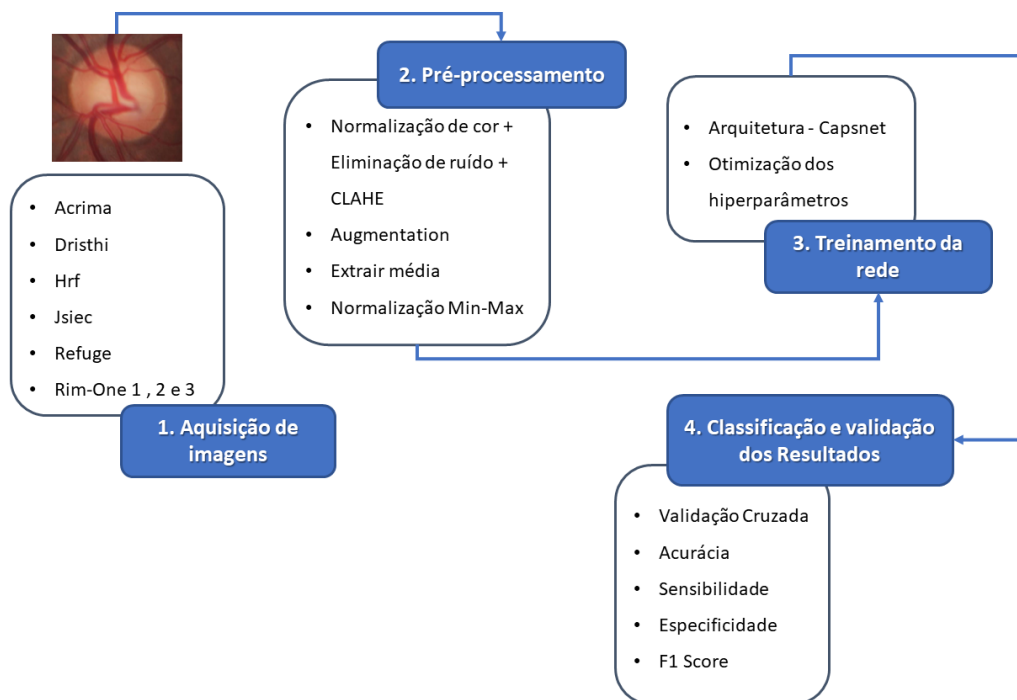
## 5 Materiais e métodos

Esta seção apresenta os materiais utilizados no trabalho que inclui, as bases de dados, *softwares* utilizados e o método proposto. Neste trabalho utilizou-se da biblioteca *Keras* (usando como *backend* o *Tensorflow*), que permite uma fácil e rápida implementação de algoritmos de aprendizado de máquina. Para o desenvolvimento do

método proposto e realização dos experimentos, foi utilizado um computador com as seguintes especificações: Placa de vídeo Nvidia GeForce GTX 1660Ti e 8 GB de RAM. Utilizou-se ainda o Colab, um serviço em nuvem gratuito disponibilizado pela Google que suporta GPU e permite desenvolver aplicações de *deep learning*.

O método proposto se divide nas seguintes etapas: (1) Aquisição de imagens, (2) Pré-processamento, (3) Definição e treinamento da rede em cápsula, e (4) Classificação e Validação dos resultados. O método está ilustrado na [Figura 6](#).

**Figura 6** - Etapas do método proposto.



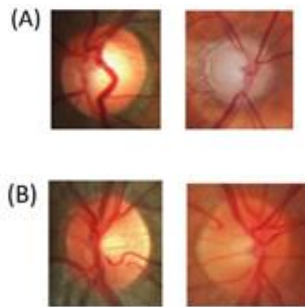
**Fonte:** Próprio aluno.

## 5.1 Aquisição de Imagens

Como primeira etapa, buscou-se bases de imagens de retinografias gratuitas na internet que estivessem rotuladas como *glaucoma* e *saudável*. O conjunto de imagem utilizado neste trabalho inclui imagens das bases: Acrima, Drishti, Hrf, Jsiec, Rim-One versão 1, 2 e 3, e Refuge, todas com a região do disco óptico segmentada. Para a base Rim-One 1, as imagens rotuladas como *Moderate* e *OHT* foram incluídas nesse trabalho na classe *Glaucoma*. Da base *Jsiec*, que possui 1000 retinografias divididas em 30 classes, foram utilizadas somente as classificadas como *normal* e *glaucoma e suspeita*. A base Refuge possui no total 1200 imagens, contudo, somente 400 são rotuladas e somente estas foram

utilizadas. A base Dristhi é geralmente utilizada para trabalhos de segmentação do nervo óptico e originalmente não possui classificação e foi obtida com a Universidade Federal do Piauí. No total, foram reunidas 2086 imagens de retinografias, sendo 848 de pacientes com glaucoma e 1227 de pacientes saudáveis. A [Figura 7](#) apresenta alguns exemplos de imagens da base de dados.

**Figura 7** – Exemplos de imagens da base. (A) Glaucoma, (B) Saudável.



Fonte: Próprio aluno.

**Tabela 1** – Número de imagens da base de dados.

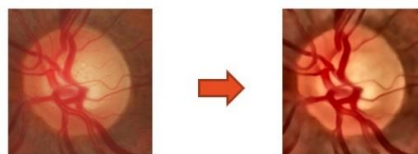
Dataset	Glaucoma	Saudável	Total
Acrima	396	309	705
Dristhi	70	31	101
Hrf	15	15	30
Refuge	40	360	400
Jsiec	13	54	67
Rim-One 1	51	118	169
Rim-One 2	200	255	455
Rim-One 3	74	85	159
<b>Total</b>	<b>859</b>	<b>1227</b>	<b>2086</b>

Fonte: Próprio aluno.

## 5.2 Pré-Processamento

A etapa de pré-processamento teve como objetivo remover ruídos e destacar aspectos da textura da retinografia. Para isso, foram feitos testes utilizando duas técnicas de pré-processamento a fim de comparar os resultados: especificação de histograma e uma técnica similar à que foi aplicada em ([Shanti et al., 2018](#)), normalizando os valores dos pixels da imagem, em seguida, aplicando a remoção de ruídos que, resumidamente, substitui os pixels de uma imagem com a média das cores de pixels similares ([Buade et al, 2011](#)), e por fim o Clahe (*Contrast Limited Adaptive Histogram Equalization*), que é uma equalização de histograma baseada em regiões da imagem ([Pizer et al, 1987](#)). A [Figura 8](#) mostra um exemplo do resultado da aplicação do pré-processamento baseado em ([Shanti, R., 2018](#)).

**Figura 8** - Aplicação da normalização, remoção de ruído e Clahe. À esquerda, a imagem original e à direita, o resultado do processamento.



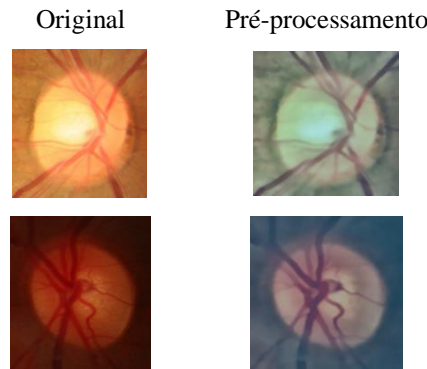
Fonte: Próprio aluno.

Por serem provenientes de diferentes fontes, as imagens da base de dados utilizada apresentam uma grande variação de cores, por isso, após a realização do pré-processamento descrito anteriormente, cada imagem foi submetida ao mesmo pré-processamento utilizado em (Simonyan et al., 2014). Este pré-processamento consiste em subtrair o valor RGB médio, computado no conjunto de treinamento, de cada pixel da imagem antes de passá-la para a rede, com o intuito de padronizar as cores das imagens. Para eliminar os possíveis valores negativos originados nessa etapa, utilizou-se a *min-max normalization*, método que permite limitar os valores para 0 e 1 sem perda de informação e é obtido utilizando a Equação 2:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

Onde  $x$  é o valor original e  $x'$  o valor normalizado. O resultado da normalização é mostrado na Figura 9. Pelo fato da média dos canais de cores das retinografias apresentarem um menor valor no canal azul (tornando a subtração menos significativa nesse canal), as imagens adquirem um tom azulado, tornando esta uma característica em comum entre as imagens.

**Figura 9** - Resultado da extração da média.

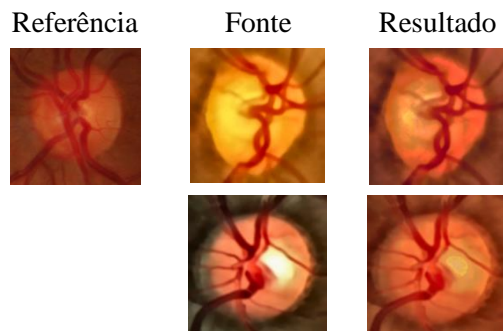


**Fonte:** Próprio aluno.

Além da extração da média, testou-se também a especificação de histograma em uma tentativa de mitigar o problema das diferentes cores das imagens da base. A especificação de histograma é uma técnica usada para modificar um dado de modo que o seu histograma combine com o histograma de uma referência pré-definida (Zhang et al., 1990). Pode ser usada para normalização no processamento de imagens, principalmente em caso onde as imagens foram obtidas de diferentes fontes ou situações (Scikit-image, 2020). Nesta pesquisa, testes foram feitos escolhendo como referência duas imagens da base de dados: uma de um paciente com glaucoma e outra de um paciente saudável. As

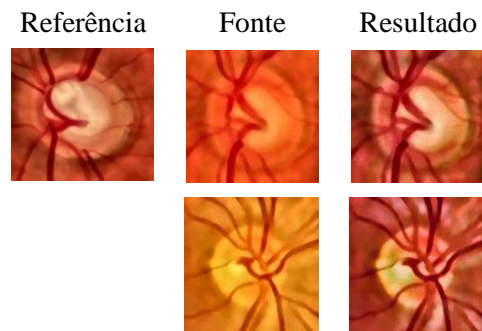
referências foram selecionadas levando em consideração a visibilidade e a qualidade dos detalhes que estas apresentavam. As [Figura 10](#) e [11](#) ilustram a aplicação da especificação de histograma.

**Figura 10** – Especificação de histograma utilizando a imagem de um paciente saudável como referência



Fonte: Próprio aluno.

**Figura 11** - Especificação de histograma utilizando a imagem de um paciente com glaucoma como referência.



Fonte: Próprio aluno.

#### 5.4 Definição e treinamento da rede em cápsula

O objetivo desta etapa é testar diferentes combinações de hiperparâmetros para a Capsnet para a obtenção de uma arquitetura que obtenha o melhor resultado. Os hiperparâmetros escolhidos focaram nas camadas convolucionais da rede, ou seja, no extrator de características, e em aspectos da configuração do treinamento da rede. Após alguns testes iniciais, foi definido um número limite de três camadas convolucionais para as Capsnets testadas.

Para otimizar os hiperparâmetros de forma automática utilizou-se a biblioteca *hyperopt* ([Bergstra et al, 2015](#)). O *hyperopt* possui um método que busca minimizar um determinado valor, dado uma função “objetiva” (que terá o seu resultado minimizado) e um espaço de busca, utilizando-se do algoritmo para otimização de hiperparâmetros conhecido como TPE (*Tree Parzen Estimator*). A vantagem deste algoritmo em relação à outros métodos populares de otimização (como o *Grid search* e *Random search*) é que este escolhe os próximos hiperparâmetros a serem testados baseado nos testes passados, diminuindo o custo computacional necessário para obtenção do melhor resultado assim como a melhora do mesmo ([Bergstra et al, 2011](#)). Ao final, o *hyperopt* retorna a configuração de hiperparâmetros entre o espaço de busca que obteve o melhor resultado para a função “objetiva”.

A função “objetiva” passada ao método incluía as seguintes ações: definir uma Capsnet baseada no espaço de busca passado ao método, configurar o treinamento e treinar a rede. No final, retorna-se o oposto da acurácia obtida na validação (já que buscava-se maximizar a acurácia) ao passar as imagens de treino para a Capsnet. Testes foram feitos tentando minimizar a sensibilidade, contudo, dependendo dos hiperparâmetros selecionados pelo *hyperopt*, a rede chegava a obter 100% de sensibilidade, ou seja, acertava todos os casos de pacientes não saudáveis, mas errava todos os pacientes saudáveis, retornando uma configuração inapropriada. Por esse motivo, escolheu-se a acurácia para ser maximizada.

Entre os hiperparâmetros estão: o tamanho do *batch* (a quantidade de amostras utilizadas para atualização dos pesos a cada iteração do treinamento), taxa de aprendizado, queda da taxa de aprendizado (fator que diminui a taxa de aprendizado com o decorrer das épocas), otimizador, função de ativação, número de épocas, tamanho do filtro das camadas convolucionais, multiplicador do filtro (utilizado para definir a quantidade de filtros das camadas convolucionais) e função de perda.

## **5.5 Classificação**

O objetivo desta etapa é classificar as retinografias em saudáveis e não saudáveis. Além de extrair características da base de dados, as redes definidas com o *hyperopt* também classificam as imagens recebidas. As últimas camadas da Capsnet consistem de camadas completamente conectadas, que conectam todos os neurônio de uma camada com cada neurônio da próxima camada, e são responsáveis pela classificação. Utiliza-se a função sigmoideal como função de ativação na última camada FC.

## **5.6 Validação dos resultados**

Para avaliar o desempenho do método, faz-se necessário a validação dos resultados obtidos. Neste trabalho foram utilizadas as seguintes métricas estatísticas: Acurácia, sensibilidade, especificidade e F1 score.



Para a análise de amostra de determinada doença com caso positivo e negativo, os resultados dos testes de classificação dos casos analisados podem ser divididos em quatro grupos (Silva, 2018):

- VP : Verdadeiro Positivo - o teste é positivo e o paciente tem glaucoma.
- FP : Falso Positivo - o teste é positivo, mas o paciente não tem glaucoma.
- VN : Verdadeiro Negativo - o teste é negativo e o paciente não tem glaucoma.
- FN : Falso Negativo - o teste é negativo, mas o paciente tem glaucoma.

A sensibilidade informa a porcentagem de pacientes doentes classificados corretamente. A especificidade, a porcentagem de pacientes saudáveis classificados corretamente. As Equações 3 e 4 mostram o cálculo da sensibilidade e especificidade, respectivamente.

$$SEN = \frac{VP}{VP + FN} \quad (3)$$

$$ESP = \frac{VN}{VN + FP} \quad (4)$$

O F1 score pode ser interpretado como uma média ponderada da precisão e sensibilidade e é calculado como na Equação 5:

$$F1 = \frac{2 * PREC * SEN}{PREC + SENS} \quad (5)$$

A acurácia significa a porcentagem de acertos entre os pacientes saudáveis e não saudáveis e pode ser definida como o número de casos da base classificados corretamente dividida pelo número total de amostras na base, como na Equação 6:

$$ACU = \frac{VP + VN}{VP + FP + VN + FN} \quad (6)$$

A base de dados utilizada (2086 imagens), foi dividida em 90% para treino e validação da rede e 10% para teste.

## 6 Resultados

Esta seção apresenta os resultados obtidos neste estudo para cada abordagem utilizada. Primeiramente, apresentam-se os resultados dos primeiros testes realizados. Foram utilizados três diferentes combinações de técnicas de pré-processamento: (1) Especificação de histograma utilizando a imagem de um paciente saudável como referência, (2) Especificação de histograma utilizando a imagem de um paciente com

glaucoma como referência e (3) Normalização + remoção de ruído + Clahe. A [Tabela 2](#) mostra o espaço de busca utilizado para o *hyperopt*. Nesses testes, utilizou-se *data augmentation*, que incluiu rotações das imagens da classe minoritária, para resolver o desbalanceamento da base e validação cruzada com 5 *folders*. Antes de inseridas na rede, as imagens tiveram suas dimensões alteradas para 112x112. A [Tabela 3](#) mostra os resultados obtidos para as três técnicas de pré-processamento.

**Tabela 2** - Espaço de busca utilizado

Hiperparâmetro	Valores
tamanho do batch	entre 10 e 45
taxa de aprendizado	entre $5 \times 10^{-5}$ e $10^{-3}$
queda da taxa de aprendizado	entre 0,1 e 0,9
otimizador	sgd, adadelta, adagrad, adam, adamax e nadam
função de ativação	relu, tanh, elu
épocas	entre 40 e 55
tamanho do filtro	5, 7, 9, 11
multiplicador do filtro	entre 0,5 e 1,2
função de perda	categorical crossentropy, mse, mean squared logarithmic error, squared hinge, categorical hinge

**Fonte:** próprio aluno.

**Tabela 3** - Resultado do teste para os três pré-processamentos.

Pré-processamento	ACU. (%)	ESP. (%)	SEN. (%)	F1. (%)
(1)	71,32 (+/- 0,01)	70,17 (+/- 0,02)	72,99 (+/- 0,02)	67,55 (+/- 0,01)
(2)	79,75 (+/- 0,01)	82,96 (+/- 0,02)	75,11 (+/- 0,03)	75,18 (+/- 0,01)
(3)	<b>85,29 (+/- 0,01)</b>	<b>88,28 (+/- 0,01)</b>	<b>81,77 (+/- 0,04)</b>	<b>83,55 (+/- 0,02)</b>

**Fonte:** Próprio aluno.

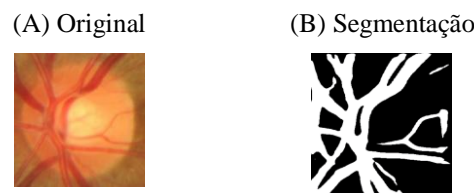
Analisando a [Tabela 3](#), percebe-se que as técnicas de pré-processamento (1) e (2), obtiveram resultados regulares, com a técnica (2) conseguindo melhores resultados que a técnica (1) indicando que a utilização da imagem de um paciente com glaucoma como referência para a especificação de histograma foi mais significativo. Contudo, a técnica de pré-processamento (3) obteve os melhores resultados gerais e por isso somente esta foi utilizada nos testes seguintes. A [Tabela 4](#) ilustra a melhor combinação de hiperparâmetros retornados pelo *hyperopt* para a técnica de pré-processamento (3).

**Tabela 4** - Melhores hiperparâmetros da técnica de pré-processamento (3)

Hiperparâmetro	Valor otimizado
tamanho do <i>batch</i>	20
taxa de aprendizado	0,00014678376495035433
queda da taxa de aprendizado	0,8922622850240577
otimizador	adamax
função de ativação	elu
épocas	40
tamanho do filtro	9
multiplicador do filtro	0,8097814944621812
função de perda	mean squared logarithmic error

**Fonte:** próprio aluno.

Partindo do fato de que os vasos sanguíneos existentes na região do disco óptico auxiliam no diagnóstico do glaucoma, os testes seguintes incluíram uma nova abordagem: uma Capsnet com duas entradas, uma para a retinografia e outra para a máscara dos vasos sanguíneos. Contudo, as bases de dados utilizadas neste trabalho não incluem a segmentação dos vasos. Como a extração dos vasos está fora do escopo desta pesquisa e buscando agilização na execução do método proposto, pesquisou-se técnicas de segmentação disponíveis na internet. O trabalho escolhido utiliza a VesselNet, uma variação da arquitetura de rede neural convolucional U-Net, baseada no trabalho de [Huang et al \(2016\)](#). Este método obteve AUC de 0,9841 na base DRIVE, que é amplamente utilizada em trabalhos de segmentação de vasos sanguíneos em retinografia. A [Figura 12](#) mostra o resultado deste método aplicado à base utilizada nesta pesquisa.

**Figura 12** - Resultado da segmentação dos vasos sanguíneos.

**Fonte:** Próprio aluno.

De posse das máscaras, definiu-se uma arquitetura de Capsnet que permite a entrada de duas imagens: a retinografia e a máscara dos vasos sanguíneos, que chamamos de Vessel Capsnet. Cada uma das duas camadas de entrada leva a um bloco de 3 camadas convolucionais, num total de 6 camadas, como mostra o exemplo da [Figura 13](#). Estes blocos são conectados no final por uma camada que concatena as características extraídas de cada bloco em um único vetor que alimentará as cápsulas da camada *PrimaryCapsule* da rede.

**Figura 13** - Exemplo de arquitetura utilizada. O tamanho e a quantidade de filtros são apenas ilustrativos já que estes são parâmetros otimizáveis.

Arquitetura utilizada		
Input(shape=112,112,3)		Input(shape=112,112,1)
Conv2D(256, 3x3)		Conv2D(256, 3x3)
Conv2D(256, 3x3)		Conv2D(256, 3x3)
Conv2D(256, 3x3)	Concat	Conv2D(256, 3x3)
	PrimaryCaps(dim=8)	
	CapsuleLayer(dim=16)	
	Dense(512)	
	Dense(1024)	
	Dense(37632)	

**Fonte:** Próprio aluno.

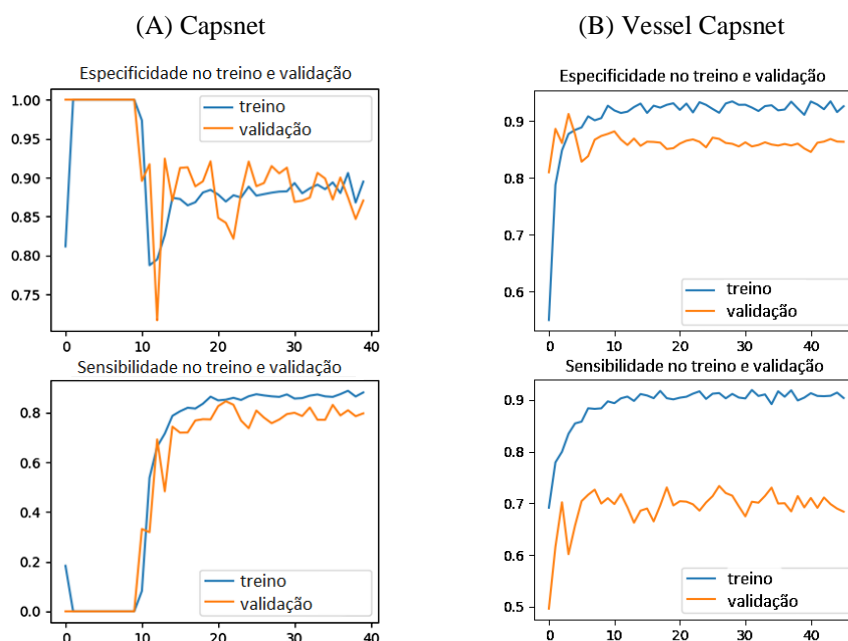
Para os testes, os hiperparâmetros foram otimizados utilizando o mesmo espaço de busca da [Tabela 2](#). Utilizou-se ainda *data augmentation* para tratar o desbalanceamento da base e validação cruzada com 5 *folders*. A comparação do resultado desta abordagem em relação à anterior é mostrado na [Tabela 5](#). A [Figura 14](#) compara a evolução da sensibilidade e especificidade de ambas as abordagens no decorrer das épocas.

**Tabela 5** - Comparação dos resultados.

Método	ACU. (%)	ESP. (%)	SEN. (%)	F1. (%)
Capsnet	<b>85,29 (+/- 0,01)</b>	88,28 (+/- 0,01)	<b>81,77 (+/- 0,04)</b>	<b>83,55 (+/- 0,02)</b>
Vessel Capsnet	83,22 (+/- 0,01)	<b>88,52 (+/- 0,01)</b>	76,87 (+/- 0,02)	80,64 (+/- 0,02)

**Fonte:** Próprio aluno.

**Figura 14** - Evolução dos métodos no decorrer das épocas.



**Fonte:** Próprio aluno.

Apesar de obter um resultado levemente superior na especificidade de acordo com a [Tabela 5](#), o método que utiliza a Vessel Capsnet não conseguiu um resultado apropriado na sensibilidade por conta do *overfitting*. Mas observando a evolução de ambos os métodos na [Figura 14](#), observa-se que a abordagem com Vessel Capsnet obteve no treino valores maiores de especificidade e sensibilidade e o aprendizado da rede se deu ainda nas 10 primeiras épocas do treinamento, o que pode indicar que a combinação dos vasos sanguíneos e retinografia melhora a classificação do glaucoma, apesar do método não conseguir generalizar o conhecimento na validação. Se tratado o *overfitting*, é possível que este método consiga valores superiores aos apresentados pela Capsnet tradicional. A [Tabela 6](#) mostra a comparação do método proposto com trabalhos relacionados. Ela inclui ainda uma abordagem testada que utiliza a arquitetura VGG16 e *transfer learning* a fim de avaliar o resultado da classificação do glaucoma utilizando a Capsnet em comparação às redes tradicionais.

**Tabela 6** - Comparação de resultados com trabalhos relacionados.

Método	Base	Nº de imagens	Performance
Li et al. (2018)	LabelMe (privada)	39.745	ACC: 98,6% SEN: 95,6% ESP: 92,2%
Raghavendra et al. (2018)	Privada	1.426	ACC: 98,13% SEN: 98% ESP: 98,3%
Lima (2019)	Rim-One 2	455	ACC: 96,63% SEN: 94,87% ESP: 98% PREC: 97,37%
Claro et al. (2019)	Hrf, Jsiec, Acrima	802	ACC: 93,61% AUC: 0,975
Pereira (2017)	Rim-One 2	455	ACC: 88,57%
VGG16 + <i>Transfer Learning</i>	Acrima, Dristhi, Hrf, Refuge, Jsiec, Rim-One 1-2-3	2086	ACC: 88,29% SEN: 84,27% ESP: 91,56% F1: 86,75%
<b>Proposto</b>	<b>Acrima, Dristhi, Hrf, Refuge, Jsiec, Rim-One 1-2-3</b>	<b>2086</b>	<b>ACC: 85,29% SEN: 81,77% ESP: 88,28% F1: 83,55%</b>

Fonte: Próprio aluno.

O trabalhos de [Li et al \(2018\)](#) e [Raghavendra et al. \(2018\)](#) obtiveram os melhores resultados, com 98,6% e 98,13% de acurácia, respectivamente. O método proposto apresentou resultados inferiores à abordagem envolvendo a arquitetura VGG16 e *transfer learning*, contudo, novos testes incluindo os vasos sanguíneos e a ampliação do espaço de busca (incluído, por exemplo, dimensão e quantidade das cápsulas nas camadas) para o *hyperopt* podem melhorar o resultado obtido.

## 7 Conclusão

Esta pesquisa visou avaliar a utilização de redes em cápsula para o diagnóstico de glaucoma em retinografias. Utilizou-se a biblioteca *hyperopt* para estimar os melhores hiperparâmetros e definir uma versão otimizada da Capsnet.

Os resultados utilizando a técnica de processamento baseada em (Shanti et al., 2018) mostraram os melhores resultados, alcançando 85,29% de acurácia, 81,77% de sensibilidade e 88,28% de especificidade. Testes incluindo a segmentação dos vasos sanguíneos em conjunto com a retinografia apresentaram resultados promissores e podem futuramente ultrapassar estes resultados.

Para trabalhos futuros, sugere-se: (1) ampliar a base de dados, (2) fazer novos testes envolvendo a combinação retinografia e vasos sanguíneos (3) ampliar o espaço de busca para incluir outros hiperparâmetros da Capsnet, (4) pesquisar outras técnicas de pré-processamento.

## Referências

ACHARYA, U. R. et al. *Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network*. Information Sciences, Vol. 405, 2017, Pág. 81-90, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2017.04.012>.

AFSHAR, P. *Capsule Networks for Brain Tumor Classification Based on MRI Images and Coarse Tumor Boundaries*, 2019. 1368-1372. 10.1109/ICASSP.2019.8683759.

ANUPAMA, A. et al. *Breast Cancer Classification using Capsule Network with Preprocessed Histology Images*, 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, Índia, 2019, pp. 0143-0147, doi: 10.1109/ICCSP.2019.8698043.

ARAÚJO, F. H. D. et al. *Redes Neurais Convolucionais com Tensorflow: Teoria e Prática*. III Escola Regional de Informática do Piauí. Livro Anais - Artigos e Minicursos, v. 1, n. 1, p. 382-406, 2017.

BARRETO, J. M. Introdução às Redes Neurais Artificiais. Laboratório de Conexionismo e Ciências Cognitivas, Universidade Federal de Santa Catarina, 2002. Disponível em: <http://www.inf.ufsc.br/~j.barreto/tutoriais/Survey.pdf>.

BERGSTRA, J. et al. *Algorithms for Hyper-Parameter Optimization*. 25th Annual Conference on Neural Information Processing Systems (NIPS 2011), 2011, Granada, Espanha.

BERGSTRA, J. et al. *Hyperopt: A Python Library for Optimizing the Hyperparameters of Machine Learning Algorithms*. 2015, Computational Science & Discovery. Vol. 8. 10.1088/1749-4699/8/1/014008.

BUADES, A. et al. *Non-Local Means Denoising*. Image Processing On Line, 1, 2011, pp. 208–212. [http://dx.doi.org/10.5201/ipol.2011.bcm\\_nlm](http://dx.doi.org/10.5201/ipol.2011.bcm_nlm).

CLARO, M. et al. *An hybrid feature space from texture information and transfer learning for glaucoma classification*. Journal of Visual Communication and Image Representation, Vol. 64, 2019, 102597, ISSN 1047-3203, <https://doi.org/10.1016/j.jvcir.2019.102597>.

FERREIRA, A. S. Redes Neurais Convolucionais Profundas na Detecção de Plantas Daninhas em Lavoura de Soja. Dissertação (Mestrado) – Universidade Federal de Mato Grosso do Sul, 2017.

GOODFELLOW, I. et al. Deep Learning. ISBN 9780262035613, MIT, 2016.

HAIKYN, S. Redes neurais: princípios e prática. trad. Paulo Martins Engel. 2. ed. Porto Alegre, Bookman, 2001.

HUANG, G. et al. Densely Connected Convolutional Networks. 2016, arXiv:1608.06993v5.

SABOUR, S. et al. *Dynamic Routing Between Capsules*. 2017, arXiv:1710.09829v2.

IESMANTAS, T. e ALZBUTAS, R. *Convolutional capsule network for classification of breast cancer histology images*. 2018, 10.1007/978-3-319-93000-8\_97.

KERAS. *Home - Keras Documentation*. Disponível em: <https://keras.io/>.

KURUP, R. et al. *Effect of Data Pre-processing on Brain Tumor Classification Using Capsulenet*. 2020, 10.1007/978-981-13-8461-5\_13.

- LAURETTI, R. C. e FILHO, A. L. Glaucomas. Medicina, Ribeirão Preto, 1997.
- LI, Z. et al. *Efficacy of a Deep Learning System for Detecting Glaucomatous Optic Neuropathy Based on Color Fundus Photographs*. 2018, Vol. 125, Issue 8, Pág. 1199-1206, ISSN 0161-6420, <https://doi.org/10.1016/j.optha.2018.01.023>.
- LIMA, A. C. M. Aprendizagem Profunda Aplicada ao Diagnóstico de Glaucoma. Dissertação (Mestrado) – Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, 2019.
- LOPES, U. K. Redes Neurais Convolucionais Aplicadas ao Diagnóstico de Tuberculose Por Meio de Imagens Radiológicas. Dissertação (Mestrado) – Universidade do Vale do Rio dos Sinos, Programa de Pós-Graduação em Computação Aplicada, 2017.
- MOBINY, A. e NGUYEN, H. V. *Fast CapsNet for Lung Cancer Screening*. Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. MICCAI 2018. Lecture Notes in Computer Science, vol. 11071. Springer, Cham. [https://doi.org/10.1007/978-3-030-00934-2\\_82](https://doi.org/10.1007/978-3-030-00934-2_82).
- MORAES, L. A. G. Utilização de Redes em Cápsula para Classificação de Regiões Promotoras em Sequências de DNA. Dissertação (Mestrado) – Programa de Pós-graduação em Ciência da Computação, Universidade Federal de Ouro Preto, 2018.
- OPENCV. OpenCV. 2020. Disponível em: <https://opencv.org/>.
- PATRICK, M. K. et al. *Capsule Network – A Survey*. Journal of King Saud University - Computer and Information Sciences, 2019, ISSN 1319-1578, <https://doi.org/10.1016/j.jksuci.2019.09.014>.
- PEREIRA, R. M. P. Estudo do Desempenho de Redes Neurais Convolucionais Aplicada ao Reconhecimento de Símbolos Musicais, Glaucoma e Texto. Monografia (graduação) – Curso de Ciência da Computação, Universidade Federal do Maranhão, São Luís, 2017.
- PIZER, S. M. et al. *Adaptive Histogram Equalization and its Variations*. Computer Vision, Graphics and Image Processing, vol. 39, 1987, pp. 355–368.
- RAGHAVENDRA, U. et al. *Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images*. Information Sciences, Vol. 441, 2018, Pág. 41-49, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2018.01.051>.



REIS, N. C. Diagnóstico Automático de Glaucoma em Imagens de Retinografia baseado no CDR usando U-Net. VII Jornada de Informática do Maranhão, 2018.

RIBEIRO, J. L. M. Meta-aprendizado aplicado ao Problema de Reconhecimento de Expressões Faciais. Monografia (Graduação) - Curso de Ciência da Computação, Universidade Federal do Maranhão, São Luís, 2018.

SANCHEZ, A. J. et al. *Capsule Networks against Medical Imaging Data Challenges*. 2018, arXiv:1807.07559v1.

SCIKIT-IMAGE. *Scikit-Image: Image processing in python*, 2020.

SHANTI, R. e PRABAKARAN, S. *Detection of Microaneurysms and Hemorrhages in Fundus Image for Glaucoma Diagnosis*. 2018, International Journal of Engineering and Technology. Vol. 7. 10.14419/ijet.v7i2.24.12092.

SILVA, M. G. Diagnóstico do glaucoma em retinografias usando descritores de textura. Centro de Ciências Exatas e Tecnologia, Universidade Federal do Maranhão, 2018.

SIMONYAN, K. e ZISSERMAN, A. *Very Deep Convolutional Networks For Large-Scale Image Recognition*. 2014; arXiv 1409.1556.

THAM, Y. C. et al. *Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis*. *Ophthalmology*, 2014; 121(11), 2081–2090. <https://doi.org/10.1016/j.ophtha.2014.05.013>.

WORLD HEALTH ORGANIZATION. *World Report on Vision*. ISBN 978-92-4-151657-0, 2019.

ZHANG, K. Y. J. e MAIN, P. *Histogram Matching as a New Density Modification Technique for Phase Refinement and Extension of Protein Molecules*. *Acta Cryst. A*46, 41-46, 1990. <https://doi.org/10.1107/S0108767389009311>.