

UNIVERSIDADE FEDERAL DO MARANHÃO – UFMA  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ITALO FRANCYLES SANTOS DA SILVA**

**DETECÇÃO AUTOMÁTICA DA PRESENÇA DE PATOLOGIA NA  
VISÃO BASEADA EM IMAGENS DO TESTE DE BRÜCKNER**

SÃO LUÍS – MA

2019

**ITALO FRANCYLES SANTOS DA SILVA**

**DETECÇÃO AUTOMÁTICA DA PRESENÇA DE PATOLOGIA NA  
VISÃO BASEADA EM IMAGENS DO TESTE DE BRÜCKNER**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. João Dallyson Sousa de Almeida

SÃO LUÍS – MA

2019

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).  
Núcleo Integrado de Bibliotecas/UFMA

Silva, Italo Francyles Santos da.

Detecção automática da presença de patologia na visão baseada em imagens do teste de BRÜCKNER / Italo Francyles Santos da Silva. - 2019.

97 p.

Orientador(a): João Dallyson Sousa de Almeida.

Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, Maranhão, 2019.

1. Análise de Cores. 2. Análise de Textura. 3. Reflexo Vermelho. 4. SVM. 5. Teste de Brückner. I. Almeida, João Dallyson Sousa de. II. Título.

**ITALO FRANCYLES SANTOS DA SILVA**

**DETECÇÃO AUTOMÁTICA DA PRESENÇA DE PATOLOGIA NA  
VISÃO BASEADA EM IMAGENS DO TESTE DE BRÜCKNER**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFMA como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Dissertação aprovada em \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_.

---

**Prof. Dr. João Dallyson Sousa de  
Almeida**  
Orientador

---

**Prof. Dr. Geraldo Braz Júnior**  
Membro da Banca Examinadora

---

**Prof. Dr. Jorge Antônio Meireles  
Teixeira**  
Membro da Banca Examinadora

---

**Prof. Dr. Sidnei Alves de Araújo**  
Membro da Banca Examinadora

SÃO LUÍS – MA  
2019

*À minha família e aos verdadeiros amigos*

## AGRADECIMENTOS

A Deus. Até aqui, sempre me ajudou o Senhor.

À minha família, por todo amor, carinho, e dedicação. Em especial, à minha avó, Maria Santana e à minha mãe, Elza Helena, pelo incentivo a continuar a jornada acadêmica. E também ao meu pai, Paulo Henrique, pelo apoio diuturno.

Ao meu orientador, João Dallyson, por todo apoio e pela paciência.

Ao professor Dr. Jorge Meirelles, por auxiliar o desenvolvimento desta pesquisa, fornecendo-nos tempo e conhecimento necessários.

Aos professores Geraldo, Anselmo, Aristófanés e Daniel pelo suporte, conselhos e conhecimento transmitido.

Aos meus amigos do Núcleo de Computação Aplicada, pelo companheirismo e ensinamentos. Além dos bons momentos de descontração.

Aos amigos do grupo Los Computers, pela amizade e todo apoio dado ao longo desta caminhada.

Aos amigos de longa data do grupo Terceirão. Apesar da distância, a amizade permanece forte.

Aos amigos Valter, Rogério e Keila pelos momentos de descontração em meio às turbulências passadas.

A todos que me acompanharam nesta jornada, e que contribuíram direta ou indiretamente na minha formação acadêmica.

*“A lei da mente é implacável. O que você pensa, cria; o que você sente, você atrai;  
o que você acredita, torna-se realizado.”*

**Buda**

## RESUMO

De acordo com o Conselho Brasileiro de Oftalmologia - CBO (2014), cerca de 4 milhões de brasileiros possuem alguma deficiência visual, sendo que desses, 33 mil crianças são cegas em decorrência de doenças que poderiam ter sido evitadas ou tratadas em tempo. Por ser de interesse à saúde pública, o Teste de Brückner, ou exame do reflexo vermelho, também conhecido como teste do olhinho, tem se consolidado como estratégia de diagnóstico precoce e prevenção de patologias oculares, o que pode trazer impactos significativos para a sociedade. Neste contexto, propõe-se um método automático de detecção de patologia na visão em imagens do Teste de Brückner, baseado em algoritmos de processamento e análise de imagens digitais e aprendizado de máquina, com intuito de contribuir no desenvolvimento de sistemas de diagnóstico assistido por computador de maior precisão e por conseguinte auxiliar no combate e prevenção das doenças. O método proposto utiliza técnicas de análise de cores e textura nas imagens do reflexo retiniano e faz uso de aprendizado de máquina para classificar os casos em saudáveis ou patológicos. O método proposto alcança 95,25% de acurácia, 84,66% de sensibilidade e 98,90% de especificidade utilizando o classificador Máquina de Vetores de Suporte.

**Palavras-chave:** Teste de Brückner, Reflexo Vermelho, Análise de Textura, Análise de Cores, Máquina de Vetores de Suporte.

## ABSTRACT

According to Brazilian Council of Ophthalmology - CBO (2014), 4 million people have some visual impairment. 33 thousand children are blinded by diseases that could be avoided or treated in time. Brückner Test interests to public health. Also known as red-reflex examination, it is an important way for early diagnosis and prevention of optical diseases. It may bring a positive social impact. In this context, this work proposes an automatic method for optical pathologies detection in Brückner test images. This method is based on image processing and machine learning algorithms, contributing for deployment of an accurate computer-aided diagnosis system that helps to avoid and prevent diseases. The proposed method uses texture and color analysis techniques and machine learning to classify cases in healthy or unhealthy. The proposed method reaches 95.25% accuracy, 84.66% sensibility, and 98.90% specificity by using Support Vector Machine classifier.

**Keywords:** Brückner Test, Red-Reflex, Texture Analysis, Color Analysis, Support Vector Machine

## LISTA DE ILUSTRAÇÕES

Figura 1 – Estrutura externa do olho humano. . . . .	25
Figura 2 – Resposta do Teste de Brückner para dois pacientes diferentes. . . . .	27
Figura 3 – Etapas fundamentais em processamento digital de imagens. . . . .	28
Figura 4 – Espaço de cores RGB . . . . .	29
Figura 5 – Interpretação geométrica dos modelos HSV (A), HLS (B) e HSI (C). . . . .	30
Figura 6 – Exemplos de Elementos Estruturantes. . . . .	34
Figura 7 – Exemplo da operação de dilatação sobre uma imagem (A) resultando em (B). O elemento estruturante é retangular e tem dimensões 10 x 10. . . . .	35
Figura 8 – Exemplo da operação de erosão sobre uma imagem (A) resultando em (B). O elemento estruturante é um quadrado de dimensões 45 x 45. . . . .	36
Figura 9 – Operações aplicadas sobre a imagem (A). Abertura resulta em (B). Fechamento, em (C). O elemento estruturante é um quadrado de dimensões 15 x 15. . . . .	37
Figura 10 – Matrizes de coocorrência em nível de cinza geradas para uma imagem. . . . .	38
Figura 11 – Matrizes de coocorrência em nível de colorido geradas para uma imagem colorida. . . . .	39
Figura 12 – Representação de uma imagem em nível de cinza (A) pelo seu histograma (B). . . . .	41
Figura 13 – Divisão de uma imagem em quatro blocos. Calcula-se um histograma para cada bloco. . . . .	42
Figura 14 – Rede <i>Multilayer Perceptron</i> . . . . .	46
Figura 15 – Arquitetura básica de uma CNN contendo convoluções, subamostragens, mapas de características (MC) e a camada completamente conectada. . . . .	49
Figura 16 – Ilustração da camada de convolução. . . . .	49
Figura 17 – Camada de subamostragem com a aplicação da operação <i>Max Pooling</i> . . . . .	50
Figura 18 – Divisão da imagem em grade para a predição de <i>bounding boxes</i> . . . . .	51
Figura 19 – Ilustração da arquitetura de uma rede completamente convolucional. . . . .	53
Figura 20 – Ilustração da arquitetura de uma rede U-Net. . . . .	54
Figura 21 – Exemplos da base de imagens. . . . .	59
Figura 22 – Exemplos de olhos de pacientes do conjunto de imagens. Diagnóstico realizado pelo especialista. . . . .	60

Figura 23 – Etapas da metodologia. . . . .	61
Figura 24 – <i>DarkConv</i> . Bloco de convoluções utilizado na rede, contendo N filtros e <i>kernel</i> de tamanho k. . . . .	62
Figura 25 – <i>DarkBlock: Darknet Residual Block</i> . Bloco residual utilizado na rede, contendo N filtros e <i>kernel</i> de tamanho k. . . . .	63
Figura 26 – Bloco contendo as últimas camadas da rede, contendo N filtros e <i>kernel</i> de tamanho k. . . . .	63
Figura 27 – Ilustração da saída da rede utilizada. . . . .	64
Figura 28 – Arquitetura YOLOV3 completa utilizada para a tarefa de localização dos olhos. . . . .	65
Figura 29 – Arquitetura U-Net utilizada neste trabalho. . . . .	67
Figura 30 – Etapas para a segmentação da região do reflexo retiniano. . . . .	68
Figura 31 – Método GLCM. A imagem do reflexo (A) é decomposta em canais (B) e, para cada canal se obtêm as matrizes (C). . . . .	70
Figura 32 – Método CLCM. A imagem do reflexo (A) é decomposta em canais (B) e, para cada combinação de canais se obtêm as 13 matrizes (C). . . . .	71
Figura 33 – Calculando histogramas globais e locais. . . . .	73
Figura 34 – Resultados do método localizador de olhos utilizando a rede YOLOv3. . . . .	78
Figura 35 – Exemplos de falsos positivos gerados pelo localizador de olhos. . . . .	78
Figura 36 – Comparativo entre o método de segmentação proposto e outras duas abordagens. . . . .	80
Figura 37 – Olhos saudáveis classificados corretamente para todos os métodos de representação de características. . . . .	87
Figura 38 – Olhos com patologia classificados corretamente para todos os métodos de representação de características. . . . .	88
Figura 39 – Reflexos retinianos semelhantes entre classes. . . . .	88
Figura 40 – Casos falhos de classificação. . . . .	89

## LISTA DE TABELAS

Tabela 1 – Sumarização das abordagens de localização de olhos. . . . .	22
Tabela 2 – Sumarização das abordagens de segmentação do reflexo retiniano. . .	23
Tabela 3 – Precisão da classificação, conforme o índice Kappa. . . . .	57
Tabela 4 – Resultado da etapa de localização de olhos utilizando YOLOv3. . . .	77
Tabela 5 – Resultados obtidos pelas redes U-Net para a segmentação do reflexo retiniano. . . . .	79
Tabela 6 – Resultados da classificação para a abordagem CLCM. . . . .	80
Tabela 7 – Resultados da classificação para a abordagem GLCM. . . . .	81
Tabela 8 – Resultados da classificação para a abordagem HCG. . . . .	82
Tabela 9 – Resultados da classificação para a abordagem MHCG. . . . .	82
Tabela 10 – Resultados da classificação para a abordagem HCL <i>grid</i> 3x3. . . . .	83
Tabela 11 – Resultados da classificação para a abordagem MHCL <i>grid</i> 3x3. . . .	83
Tabela 12 – Resultados da classificação para a abordagem HCL <i>grid</i> 5x5. . . . .	84
Tabela 13 – Resultados da classificação para a abordagem MHCL <i>grid</i> 5x5. . . .	84
Tabela 14 – Resultados da classificação para a abordagem combinada CLCM + MHCL <i>grid</i> 5x5. . . . .	85
Tabela 15 – Resultados da classificação para a abordagem combinada GLCM + MHCL <i>grid</i> 5x5. . . . .	85
Tabela 16 – Resumo dos resultados da etapa de classificação. . . . .	86
Tabela 17 – Artigos publicados que possuem relação com o método proposto. . .	91

## LISTA DE ABREVIATURAS E SIGLAS

A	Acurácia
CAD	<i>Computer-Aided Detection</i>
CADx	<i>Computer-Aided Diagnosis</i>
CBO	Conselho Brasileiro de Oftalmologia
CLCM	<i>Color-Level Cooccurrence Matrix</i>
CNN	<i>Convolutional Neural Network</i>
CV	Coefficiente de Variação
E	Especificidade
ED	Entropia da Diferença
ES	Entropia da Soma
FCN	<i>Fully Convolutional Network</i>
FN	Falso Negativo
FP	Falso Positivo
GLCM	<i>Gray-Level Cooccurrence Matrix</i>
HGC	Histograma Global de Cores
HLC	Histograma Local de Cores
IOU	<i>Intersection Over Union</i>
MLP	<i>Multilayer Perceptron</i>
MSSD	Quadrado Médio das Diferenças Sucessivas
RBF	<i>Radial Basis Function</i>
RNA	Rede Neural Artificial
ROI	Região de Interesse

ROC	<i>Receiver Operating Characteristic</i>
S	Sensibilidade
SMA	Segundo Momento Angular
SVM	<i>Support Vector Machine</i>
VD	Variância da Diferença
VS	Variância da Soma
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo

## SUMÁRIO

1	INTRODUÇÃO . . . . .	16
1.1	Objetivos . . . . .	18
1.2	Objetivo Geral . . . . .	18
1.3	Objetivos Específicos . . . . .	18
1.4	Contribuições . . . . .	18
1.5	Organização do trabalho . . . . .	19
2	TRABALHOS RELACIONADOS . . . . .	20
3	FUNDAMENTOS TEÓRICOS . . . . .	25
3.1	O Olho Humano . . . . .	25
3.2	Teste de Brückner . . . . .	26
3.3	Processamento de Imagens Digitais . . . . .	27
3.3.1	Modelos de Representação de Cores . . . . .	28
3.3.2	Morfologia Matemática . . . . .	34
3.4	Descritores de Características . . . . .	37
3.4.1	Baseados em Textura . . . . .	37
3.4.1.1	Matrizes de Coocorrência de Níveis de Cinza . . . . .	38
3.4.1.2	Matrizes de Coocorrência de níveis de Cores . . . . .	39
3.4.2	Baseados em cores . . . . .	40
3.4.2.1	Histograma de cores . . . . .	40
3.4.3	Abordagem Estatística . . . . .	42
3.5	Redes Neurais . . . . .	45
3.5.1	Redes <i>Multilayer Perceptron</i> . . . . .	45
3.6	Aprendizagem Profunda . . . . .	47
3.6.1	Redes Neurais Convolucionais . . . . .	48
3.6.2	YOLO - <i>You Only Look Once</i> . . . . .	50
3.6.3	U-Net: Rede Totalmente Convolucional . . . . .	52
3.7	Máquina de vetores de suporte (SVM) . . . . .	54
3.8	Métricas de Avaliação . . . . .	55
3.9	Considerações Finais . . . . .	58

4	MATERIAIS E MÉTODOS . . . . .	59
4.1	Base de Imagens . . . . .	59
4.2	Método Proposto . . . . .	60
4.2.1	Localização dos olhos . . . . .	61
4.2.1.1	Arquitetura da rede YOLO . . . . .	62
4.2.1.2	Treinamento, Validação e Teste . . . . .	65
4.2.2	Segmentação do Reflexo Retiniano utilizando U-Net . . . . .	66
4.2.2.1	Estratégia de Segmentação . . . . .	67
4.2.2.2	Treinamento da U-Net . . . . .	69
4.2.3	Extração de características . . . . .	69
4.2.3.1	Extração de Características de Textura . . . . .	70
4.2.3.1.1	GLCM . . . . .	70
4.2.3.1.2	CLCM . . . . .	71
4.2.3.1.3	Descrição e vetores de características . . . . .	71
4.2.3.2	Extração de características de cor . . . . .	72
4.2.3.2.1	Histogramas de cores global e local . . . . .	72
4.2.3.2.2	Extração de métricas do histograma . . . . .	73
4.2.4	Treinamento e Classificação utilizando SVM . . . . .	73
4.3	Hardware e Softwares Utilizados . . . . .	75
4.4	Considerações Finais . . . . .	75
5	RESULTADOS E DISCUSSÃO . . . . .	77
5.1	Localização dos Olhos . . . . .	77
5.2	Segmentação do Reflexo Retiniano . . . . .	78
5.3	Classificação . . . . .	79
5.3.1	Avaliação das características de textura . . . . .	80
5.3.2	Avaliação das características de cores . . . . .	81
5.3.3	Combinação de características de textura e cor . . . . .	85
5.3.4	Estudos de caso . . . . .	87
6	CONCLUSÃO . . . . .	90
6.1	Trabalhos Futuros . . . . .	91
6.2	Produções científicas . . . . .	91

**REFERÊNCIAS . . . . . 92**

## 1 INTRODUÇÃO

O Censo Oftalmológico 2014 (CBO, 2014), realizado pelo Conselho Brasileiro de Oftalmologia (CBO), aponta que, no Brasil, cerca de 1,1 milhão de pessoas possuem cegueira e 4 milhões têm problemas visuais, desconsiderando as pessoas com erros refrativos. Cerca de 100 mil crianças têm alguma deficiência visual, sendo que 33 mil são cegas por doenças que poderiam ter sido evitadas ou tratadas. Segundo o órgão, o Brasil registra um indicador digno das nações desenvolvidas em relação ao número de especialistas: um para cada grupo de 11.604 pessoas. A Organização Mundial de Saúde (OMS) estabelece como ideal a proporção de um para 17.000. Entretanto, devido à concentração de profissionais em cidades grandes, um número significativo de municípios carece de oftalmologistas, o que contribui para o problema da saúde ocular brasileira. Nas Regiões Sul e Centro-Oeste, por exemplo, estão, respectivamente, 15%, e 7% desses profissionais. No Nordeste, encontram-se 19%. Os percentuais apresentam proporções adequadas à recomendação da OMS. Porém, é identificado que nos estados de Alagoas, Sergipe, Paraíba e Maranhão existe déficit.

Ainda segundo o CBO, o acompanhamento oftalmológico desde os primeiros anos de vida e a incorporação dos especialistas às iniciativas de atenção primária são salutares para a sociedade, pois trazem impactos significativos. Erros refrativos e demais patologias oculares podem comprometer o desempenho dos indivíduos em suas atividades cotidianas, escolares e/ou laborais. Portanto, o diagnóstico e tratamento precoce são importantes para uma melhora na qualidade de vida, principalmente na infância - período no qual as chances de correção da visão são maiores.

Um exame simples, indolor, e capaz de detectar alterações visuais é o Teste de Brückner. Também conhecido como exame do reflexo vermelho, esse teste pode ser usado como parte da rotina de exames de crianças recém-nascidas, contribuindo, assim, para a identificação prematura de problemas oculares, incluindo a catarata congênita, erros refrativos e retinoblastoma (SUN et al., 2016). Esse exame também é praticado para o diagnóstico do estrabismo (MILLER et al., 1995).

Durante o exame, o especialista utiliza um aparelho chamado oftalmoscópio o qual atua como fonte luminosa e incide luz branca nos olhos do paciente. Ao entrar em contato com a parte interna do olho, que é bastante irrigada por vasos sanguíneos, a luz é

refletida na cor vermelha. Isso mostra que as principais estruturas internas do olho estão transparentes, permitindo que o feixe luminoso atinja a retina normalmente. Entretanto, existem casos em que o reflexo não possui boa qualidade, pois, a luz é refletida com outra tonalidade ou é totalmente absorvida. Isso significa que há alteração que cause obstrução no eixo visual caracterizando assim uma possível presença de patologia. O fenômeno do reflexo retiniano é muitas vezes observado em fotografias com *flash*.

O presente trabalho apresenta um método para detectar a presença de patologia na visão baseando-se em imagens do Teste de Brückner, de forma a proporcionar suporte aos especialistas na análise da saúde visual dos pacientes, mediante a verificação da qualidade apresentada pelo reflexo retiniano durante o exame.

A partir de fotografia obtida por uma câmera comum ou *smartphone* e da utilização de técnicas de processamento de imagens e aprendizado de máquina, o método desenvolvido analisa a imagem da face dos pacientes para localizar a região dos olhos e, em seguida, segmentar a região de interesse que é o reflexo retiniano. Depois, extrai as características dessa região para então classificar cada olho como saudável ou com problema. Pretende-se, dessa forma, contribuir com a atuação profissional, facilitando e melhorando a qualidade da assistência, principalmente em áreas de difícil acesso.

Como dito anteriormente, a má distribuição dos oftalmologistas pelo território brasileiro acarreta para que algumas regiões tenham significativa carência destes profissionais. Nessas regiões, a triagem oftalmológica se apresenta como forte aliada para a detecção prematura de patologias oculares. Como teste de triagem para detectar alterações visuais, o teste de Brückner é uma opção simples, útil e com precisão razoável (JALIS, 2015).

Com os avanços da tecnologia e da computação, por meio da natureza digital das imagens médicas geradas, a pesquisa e desenvolvimento de técnicas de processamento de imagem contribuíram para os sistemas CAD (*Computer Aided Detection*) e CADx (*Computer Aided Diagnosis*), com o intuito de ajudar a equipe médica na interpretação de imagens, aprimorando a eficiência e a precisão de diagnósticos e, conseqüentemente, proporcionar uma outra concepção. Dessa forma, a presente proposta, ao automatizar o exame de Brückner, objetiva corroborar com essas características.

## 1.1 Objetivos

Nesta Seção, serão apresentados os objetivos gerais e específicos a serem alcançados com o desenvolvimento do presente trabalho.

## 1.2 Objetivo Geral

Este trabalho objetiva propôr um método para detectar a presença de patologia na visão em imagens do teste automático de Brückner, proporcionando suporte aos especialistas na análise da saúde visual dos pacientes.

## 1.3 Objetivos Específicos

A realização do objetivo geral é possibilitada com o cumprimento dos seguintes objetivos específicos:

- Construção de um conjunto de imagens de pacientes submetidos ao Teste de Brückner para a realização dos experimentos.
- Desenvolvimento de um método computacional para a localização automática dos olhos do paciente.
- Desenvolvimento de um método computacional para a segmentação automática do reflexo retiniano.
- Análise comparativa entre métodos de análise de textura e cores como descritores das características do reflexo retiniano.

## 1.4 Contribuições

A principal contribuição do presente trabalho é a proposta do método para automatizar o Teste de Brückner utilizando técnicas de processamento de imagens médicas digitais e aprendizado de máquina. Outras contribuições podem ser citadas:

- Construção de uma base de imagens com exemplares de pacientes submetidos ao Teste de Brückner.
- Utilização de uma arquitetura de rede neural convolucional para a localização da região dos olhos, robusta ao processo de captura da imagem do paciente.

- Elaboração do método de segmentação do reflexo retiniano com o uso de uma arquitetura de rede completamente convolucional.

## 1.5 Organização do trabalho

Esta dissertação está organizada da seguinte forma: o **Capítulo 2** apresenta os trabalhos relacionados ao tema investigado; no **Capítulo 3** são apresentados os fundamentos teóricos importantes para a compreensão do presente trabalho, discorrendo-se sobre o Teste de Brückner e as técnicas de processamento de imagens e aprendizado de máquina utilizadas, bem como as medidas de validação dos resultados.

No **Capítulo 4** é descrita a base de imagens e o método proposto, explicitando-se cada etapa desse método e como ele é avaliado. O **Capítulo 5** apresenta os resultados obtidos com pelo método proposto; os quais são discutidos de forma a apontar pontos fortes e fracos por meio de estudos de caso. Por fim, no **Capítulo 6** são apresentadas as conclusões obtidas, mostrando a eficácia do método proposto. Nesse capítulo, também são apresentadas as contribuições da pesquisa, bem como sugestões para trabalhos futuros.

## 2 TRABALHOS RELACIONADOS

Este capítulo apresenta o referencial teórico utilizado para o desenvolvimento deste estudo. Os trabalhos aqui mencionados referem-se ao processo de localização de olhos em imagens, segmentação do reflexo retiniano e também métodos de detecção de patologias oculares a partir da fotografia da região externa do olho.

Para desenvolver um método capaz de auxiliar o especialista na detecção da presença de patologia na visão, é necessária uma etapa para a localização dos olhos em imagens. Zhang et al. (2014) analisam projeções horizontais para identificar o conjunto de pixels que compreendem a região dos olhos. O método detector de olhos, parte central da aplicação, é baseado no princípio *Three Court Five Eyes*. *Three Court* significa que a face pode ser dividida igualmente em três partes na vertical, da linha do cabelo ao final do queixo. *Five Eyes* corresponde à divisão da face na horizontal em cinco partes, de uma extremidade à outra. E cada parte tem tamanho igual à largura do olho humano. Os olhos estão na segunda e quarta fatia do segundo terço da face. O método alcança acurácia média de 97,3% nas imagens com iluminação normal. O pior desempenho acontece no caso dos indivíduos com óculos (74,3% de acurácia média).

Lan e Mu (2016) propõem um método que utiliza a geometria aliada ao operador de SUSAN (SMITH; BRADY, 1997) para a localização do centro do olho e dos cantos a partir de imagens capturadas por web câmeras comuns. O operador SUSAN é utilizado principalmente para encontrar cantos e ângulos em uma imagem. Geometricamente, o centro de objetos circulares pode ser encontrado a partir da análise do campo vetorial dos gradientes da imagem. Os cantos dos olhos, por sua vez, são localizados a partir da aplicação do operador de SUSAN. Avaliado com imagens das bases BioID e FERET, o detector alcança, respectivamente, 99,80% e 99,15% de acurácia.

Benedict e Kumar (2016) apresentaram o método *Geometric Shaped Facial Feature Extraction for Face Recognition* (GSF2EFR) com objetivo de identificar o indivíduo a partir da descoberta do centro e do canto do olho. Esse método faz a aplicação dos Filtros de Gabor na imagem da face para que sejam extraídas as características de forma que, por meio do classificador SVM (*Support Vector Machine*), seja possível distinguir aquelas que dizem respeito às regiões dos olhos das demais áreas da face. Em seguida, o SVM classifica o ponto de centro do olho. Os testes obtiveram acurácia máxima de 85% na

detecção de olhos de um indivíduo da base Yale B e 95% para a base MMI. As respectivas taxas de verdadeiros positivos para esses casos foram 83% e 93%.

O campo *Deep Learning* (Aprendizagem Profunda) tem incentivado muitas pesquisas em como tornar certas tarefas em visão computacional, como a detecção de olhos, menos suscetíveis a erros, principalmente quando aplicadas sobre situações não-controladas. O trabalho de Jerry e Eizenman (2008) apresenta a utilização de uma rede neural convolucional (CNN - *Convolutional Neural Network*) para a detecção de olhos. O experimento foi realizado com uma base proprietária com 3000 exemplares de regiões de olhos recortadas manualmente. Para uma melhor generalização, os dados foram ampliados utilizando-se *data augmentation*. A base passou a conter 30000 imagens. O método com CNN proposto alcança as taxas de 10% para falsos positivos e 99,3% de verdadeiros positivos.

Ye et al. (2014) propõem um detector que utiliza CNNs em dois níveis (cascata). No primeiro nível, a rede neural detecta *bounding boxes* que delimitam regiões candidatas a olho. Em seguida, a outra rede neural é utilizada para julgar quais das *bounding boxes* encontradas são de regiões de olhos. No final, as *bounding boxes* selecionadas pelo segundo nível são fundidas, produzindo o resultado final: a localização do olho. Para o treinamento, foram coletadas 8300 imagens de olhos da base de dados *Public Figures Face Database* (KUMAR et al., 2009a), da qual também foram coletadas 8000 imagens rotuladas como "não-olho". Os testes foram realizados com a base de dados LFW, que contém imagens sob condições irrestritas. Como resultado, a abordagem em cascata alcançou 93% de precisão e 91% de *recall*.

Wu e Ji (2014) utilizaram o modelo *Deep Boltzman Machine* (DBM) (SALAKHUT-DINOV; HINTON, 2009) para capturar os padrões distintos dos olhos de forma que o detector seja eficaz mesmo em situações irrestritas. Após esse processo, parte-se para o treinamento de um classificador, no caso, uma rede neural MLP (*Multilayer Perceptron*), para discriminar essas características de forma eficaz. As acurácias obtidas foram: 100% (base YaleB), 98,12% (base BioID) e 99,22% (base CK+). A Tabela 1 apresenta a sumarização das abordagens relacionadas à localização de olhos.

Na literatura, também são apresentadas propostas para a detecção do reflexo retiniano nos olhos, mais precisamente o de cor vermelha. E estão atreladas à tarefa de correção, visto que o fenômeno é considerado um problema na aquisição de fotografias. As abordagens apresentadas a seguir utilizam bases de imagens proprietárias contendo

Tabela 1 – Sumarização das abordagens de localização de olhos.

<b>Autor</b>	<b>Abordagem</b>	<b>Imagens</b>
Zhang et al. (2014)	- Three Court Five	Base proprietária
Lan e Mu (2016)	- Detecção de cantos (SUSAN)	BioID e FERET
Benedict e Kumar (2016)	- Filtros de Gabor e SVM	YaleB e MMI
Jerry e Eizenman (2008)	- Rede Neural Convolutacional (CNN)	Base proprietária
Ye et al. (2014)	- CNNs em cascata	LFW
Wu e Ji (2014)	- Deep Boltzman Machine	CK+, BioID e YaleB
<b>Neste trabalho</b>	- YOLOv3	Base proprietária

poucos exemplares.

Em Kumar et al. (2009b) é apresentado um sistema que atua na detecção e correção do reflexo vermelho em imagens de forma simples. Primeiro, detecta-se a face por meio da análise de cor de pele, uma importante característica do rosto humano. A imagem é analisada em 3 espaços de cores, HSV, RGB e YCrCb. Para cada canal, os autores definem uma faixa de valores de pixel que correspondem às regiões de pele. Esses valores foram definidos empiricamente, os quais os autores julgam como generalizados para vários tons de pele. Os limites da face são encontrados por meio das projeções vertical e horizontal. Uma vez definido o local da face, a localização do olho é definida por outro valor limiar, de forma que, com a binarização da imagem resultante do processo anterior, sejam recuperadas as manchas oculares (onde o brilho do reflexo retiniano é muito elevado).

O método proposto por Yoo e Park (2009) detecta o olho vermelho utilizando a técnica de crescimento de região. A proposição de regiões candidatas é dada pela avaliação da medida *redness* definida em (GAUBATZ; ULICHNEY, 2002). *Pixels* são considerados vermelhos caso seu *redness* seja superior a um limiar definido empiricamente. O crescimento de região é, portanto, iniciado e os pixels são inclusos em determinada região se: (a) se *redness* e a luminância forem superiores, cada uma, a um limiar particular; (b) se os valores nas componentes Cb e Cr pertencerem a uma faixa determinada empiricamente.

Ali, Khan e Kim (2009) combinam três métricas para medir o *fitness* de cada ponto candidato a íris. O método utiliza *templates* circulares para a detecção da região da íris, onde está o reflexo vermelho. A base de imagens utilizada no experimento relatado é composta por 50 imagens. Em 41 delas o algoritmo capturou precisamente a localização do olho (82% de acurácia). A partir dos centros dos olhos, uma região circular que compreende o reflexo vermelho pode ser segmentada para fins de correção.

O método proposto por Lepisto, Launiainen e Kunttu (2009) também é baseado em forma, produzindo uma detecção voltada para regiões circulares de cor vermelha. A metodologia segue dois passos principais: (1) a avaliação das regiões da imagem por meio do cálculo de *redness*, e (2) a análise da forma das regiões candidatas. Analisando a acurácia obtida com os experimentos (70%) e a baixa taxa de falsos positivos (1%), o autor constata que o método falha nos casos em que a cor do olho se diferencia do *redness* estabelecidos (olho amarelado, por exemplo). A Tabela 2 sumariza as abordagens de segmentação do reflexo vermelho apresentadas nesta Seção.

O presente trabalho, entretanto, aborda a segmentação do reflexo retiniano utilizando uma abordagem dentro do contexto da Aprendizagem Profunda chamada U-Net, rede convolucional que já apresentou resultados promissores na segmentação de imagens biomédicas (RONNEBERGER; FISCHER; BROX, 2015), de forma a tornar mais robusta às variações de tonalidade do reflexo.

Tabela 2 – Sumarização das abordagens de segmentação do reflexo retiniano.

<b>Trabalho</b>	<b>Abordagem</b>
Kumar et al. (2009b)	- Limiarização de imagem
Yoo e Park (2009)	- Crescimento de região
Ali, Khan e Kim (2009)	- Localização da íris via maximização de medidas de correlação
Lepisto, Launiainen e Kunttu (2009)	- Cálculo de <i>redness</i> e análise das formas candidatas
<b>Neste trabalho</b>	- U-Net

Na literatura, existem trabalhos que propõem métodos para a identificação de algumas patologias na visão baseando-se na extração e análise de características em fotografias do olho.

Em Rivas-Perea et al. (2014) é proposto o diagnóstico da leucocoria, caracterizada pelo reflexo esbranquiçado, em fotografias capturadas de forma espontânea em momentos recreativos. A proposta utiliza a Transformada Discreta do Cosseno, Análise de Componentes Principais para extrair características das imagens, e também a teoria de lógica *fuzzy* para combinar os classificadores Máquina de Vetores de Suporte (SVM), Redes Neurais Artificiais e Análise de Discriminantes Lineares, alcançando 92% de acurácia, com 89% de taxa de verdadeiros positivos.

O diagnóstico da leucocoria também é realizado por Henning et al. (2014). A base de imagens utilizada consiste também de imagens espontâneas de crianças com retinoblastoma, além de imagens retiradas de uma rede social. Os autores categorizaram as imagens em três tipos: com leucocoria, sem leucocoria e pseudo-leucocoria, totalizando

832 imagens. O método proposto utiliza de redes neurais convolucionais e alcança uma taxa de 96,8% para os verdadeiros positivos.

No trabalho de Prashasthi et al. (2017) a extração de características cromáticas da imagem externa do olho compõe um método que auxilia a identificação das seguintes patologias oculares: catarata, conjutivite, icterícia, hemorragia e paralisia. A extração de características consiste em converter as imagens do espaço RGB para HSV e analisar a porcentagem da cor característica para cada patologia ocular. A base de imagens é composta por 150 exemplares, dos quais 77% foram utilizados para treino e 23% para teste. A proposta obteve acurácia geral de 94,28% utilizando o classificador *Adaboost*.

O presente trabalho, diferente desses supracitados, propõe um método para a detecção da presença de patologias na visão em imagens do Teste de Brückner (Capítulo 4). O método segue um fluxo de execução que possui as etapas de localização de olhos e segmentação do reflexo retiniano e classificação em saudável ou patológico.

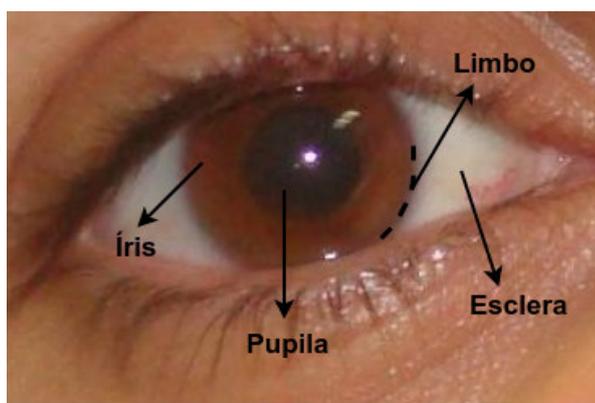
### 3 FUNDAMENTOS TEÓRICOS

Este capítulo apresenta a fundamentação teórica utilizada no desenvolvimento deste trabalho, importante para a compreensão do contexto ao qual ele está inserido, bem como das técnicas adotadas para alcançar os objetivos. A seguir, aborda-se sobre aspectos estruturais do olho humano, contextualizando o teste de Brückner. Apresentam-se os conceitos e técnicas de processamento de imagens digitais, bem como métodos de análise de textura e cor para extração de características, técnicas de reconhecimento de padrões e as métricas de avaliação dos resultados.

#### 3.1 O Olho Humano

A visão é um dos cinco sentidos os quais provêm aos seres vivos a capacidade de percepção do meio em que se encontram. O olho é o órgão responsável pela visão humana. Caracterizado pelo formato globular, esse órgão está localizado em uma cavidade óssea na face e suas estruturas podem ser divididas em três camadas: externa, composta pela esclera, córnea e limbo; medial, onde está a íris, pupila, coróide e corpo ciliar; e interna, composta pela retina (JOGI, 2009). A Figura 1 apresenta as principais componentes da camada externa do olho, cujo conhecimento prévio é importante para a compreensão deste trabalho.

Figura 1 – Estrutura externa do olho humano.



Fonte: Elaborada pelo Autor.

A percepção de mundo por meio da visão se dá pela seguinte forma: o olho capta a luz refletida pelos elementos dispostos no meio através da córnea. A luz entra em contato com a íris e penetra o globo ocular de forma controlada por uma abertura chamada pupila.

Em seguida, atinge o cristalino, que atua como uma lente responsável pela focalização. Por fim, os raios luminosos convergem para a retina, onde células fotossensíveis enviarão um impulso para o cérebro através do nervo óptico. Alterações em uma ou mais fases desse processo significam o comprometimento da saúde ocular.

### 3.2 Teste de Brückner

O Teste de Brückner é um importante exame oftalmológico através do qual é possível detectar patologias na visão precocemente. Simples e indolor, trata-se de uma forma de triagem visual de baixo custo e de relevante contribuição para a identificação prematura de problemas oculares, incluindo a catarata congênita, erros refrativos e retinoblastoma (SUN et al., 2016). Também é popularmente conhecido como teste do olhinho e exame do reflexo vermelho.

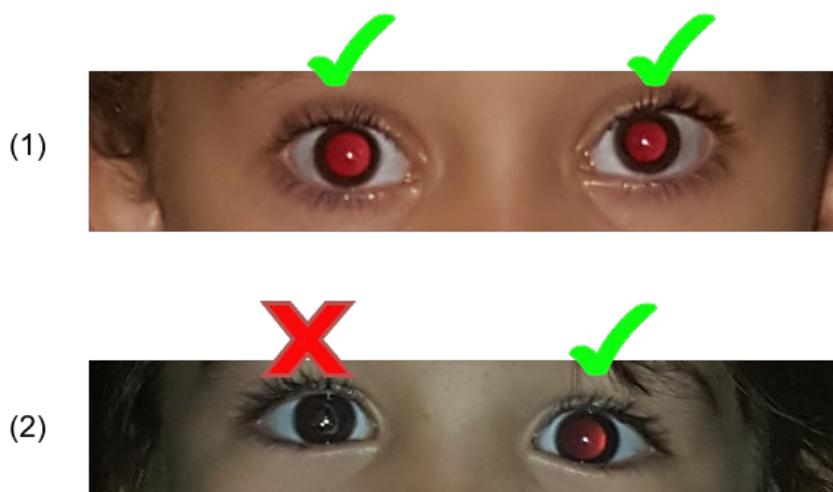
O exame consiste na avaliação da qualidade dos meios transparentes do olho por meio da observação do reflexo retiniano (JALIS, 2015). O procedimento caracteriza-se pelo direcionamento de luz branca nos olhos do paciente a partir de uma fonte luminosa. Ao entrar em contato com a parte interna do olho, altamente irrigada por vasos sanguíneos, essa luz é refletida na cor vermelha. A obtenção de um reflexo com essas características mostra que as principais estruturas internas do olho estão transparentes, permitindo que a luz atinja a retina normalmente. Isso caracteriza o reflexo como sendo de boa qualidade. É importante ressaltar que o fenômeno do reflexo retiniano é comum em fotografias com *flash*.

No entanto, existem casos em que a qualidade do reflexo aparenta-se comprometida, pois, a luz é refletida com outra tonalidade ou é totalmente absorvida. Esses aspectos indicam que algo impede a passagem normal da luz pela camada interna, corroborando portanto uma possível presença de patologia. A partir disso, iniciam-se os devidos procedimentos para o diagnóstico e tratamento.

Na Figura 2 é possível visualizar e compreender a diferença entre os aspectos dos reflexos retinianos apresentados após a incidência de luz branca nos olhos de um paciente. Em (1) observa-se reflexos retinianos em olhos saudáveis. A cor avermelhada indica o contato da luz com a retina. Em (2), porém, percebe-se que o olho direito apresenta outro comportamento. Logo, é possível que a saúde ocular deste paciente esteja comprometida.

Segundo Aguiar (2010), em relação às cores, o fundo de olho normalmente varia

Figura 2 – Resposta do Teste de Brückner para dois pacientes diferentes.



Fonte: Elaborada pelo Autor.

de vermelho-claro a vermelho-escuro, assumindo tons castanhos ou ainda mais escuros. Então o reflexo pode assumir tais colorações e indicar condições saudáveis. Tons que fogem desses padrões tendem a indicar patologias. Porém, o exame requer cautela para que se encontrem sinais de opacidade ainda que o reflexo aparente condições normais.

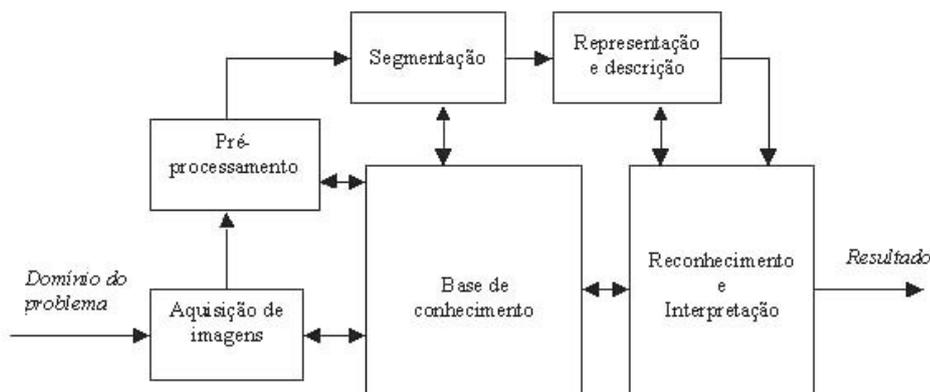
### 3.3 Processamento de Imagens Digitais

Processamento de imagens é definido, em linhas gerais, como um conjunto de técnicas computacionais que transformam uma imagem de entrada em uma saída desejada (GONZALEZ; WOODS, 2010). Essas técnicas possibilitam a extração e identificação de informações nas imagens, melhorando a qualidade visual de aspectos estruturais. Segundo Pedrini e Schwartz (2007), isso facilita a percepção humana e a interpretação automática por meio de máquinas

Existem muitos algoritmos com determinadas finalidades em processamento de imagens. Então, as metodologias se diferenciam ao passo em que cada uma utiliza técnicas para propósitos específicos. Não obstante, geralmente obedecem a sequência de passos apresentada na Figura 3.

Gonzalez e Woods (2010) apontam que as etapas fundamentais para resolução de problemas de processamento de imagem são: aquisição das imagens digitais, pré-processamento, segmentação, representação, descrição, reconhecimento e interpretação. Cada etapa gera um resultado, o qual é utilizado como entrada para o passo subsequente.

Figura 3 – Etapas fundamentais em processamento digital de imagens.



Fonte: Adaptado de (GONZALEZ; WOODS, 2010).

No entanto, uma metodologia pode seguir somente um subconjunto destas etapas.

Esta seção apresenta as principais técnicas de processamento de imagens utilizadas no desenvolvimento deste trabalho, aplicadas em tarefas de melhoramento, representação e análise.

### 3.3.1 Modelos de Representação de Cores

Em processamento digital de imagens, as cores são poderosos descritores das propriedades de um objeto (MARQUES FILHO; VIEIRA NETO, 1999). Considerá-las nesse processo pode tornar tarefas como identificação e segmentação mais simplificadas.

Para que objetos coloridos sejam vistos, é necessário que o olho seja atingido pela luz refletida por eles. Por exemplo, se um objeto parece azul à luz do dia é porque reflete somente a parte azul da luz recebida para os nossos olhos, enquanto que o restante do espectro é absorvido.

As cores podem ser diferenciadas entre si pelas seguintes características: brilho, matiz e saturação. O brilho representa a intensidade luminosa da radiação. A Matiz associa-se à pureza da cor, isto é, o comprimento de onda predominante ainda que haja combinação de várias ondas visíveis. Por fim, a saturação expressa o grau de mistura do matiz original com a luz branca. Cores como o verde e o verde claro, por exemplo, têm o mesmo valor de matiz, mas apresentam diferentes graus de saturação.

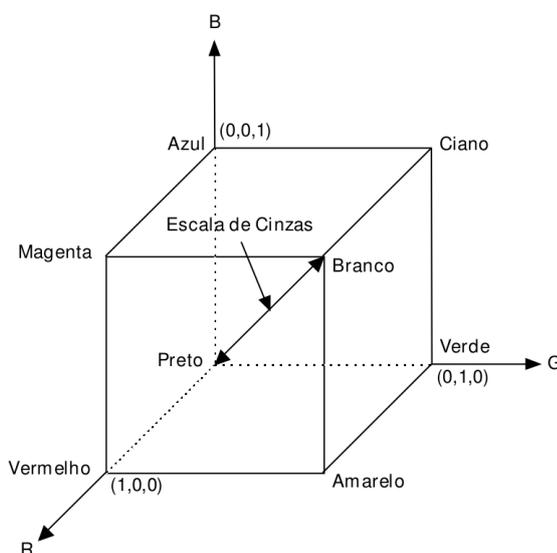
O objetivo dos modelos é permitir a especificação de cores em um formato padronizado. Um modelo de cores é, portanto, uma representação tridimensional na qual

cada cor é representada por um ponto no sistema de coordenadas 3-D. Por isso, também são chamados de espaços de cores ou sistemas de cores.

Como dito anteriormente, cores são poderosos descritores. Logo, a representação das imagens em espaços de cores diferentes pode trazer à tona características latentes, as quais podem auxiliar o reconhecimento de padrões, bem como demais tarefas de processamento.

Um dos modelos mais conhecidos é o RGB (*red, green, blue*). Baseado em um sistema de coordenadas cartesianas, sua principal abstração é em forma de cubo, conforme ilustrado pela Figura 4. Três vértices representam as cores primárias; outros três as cores secundárias; o vértice junto à origem corresponde o preto e o seu antípoda, à cor branca. Neste modelo, a escala de cinza se estende através da diagonal do cubo que sai do preto (origem) em direção ao branco. Os valores de R, G e B são representados numa escala de 0 a 255. Há também representações normalizadas entre 0 e 1. O modelo RGB é o mais utilizado por câmeras e monitores de vídeo.

Figura 4 – Espaço de cores RGB

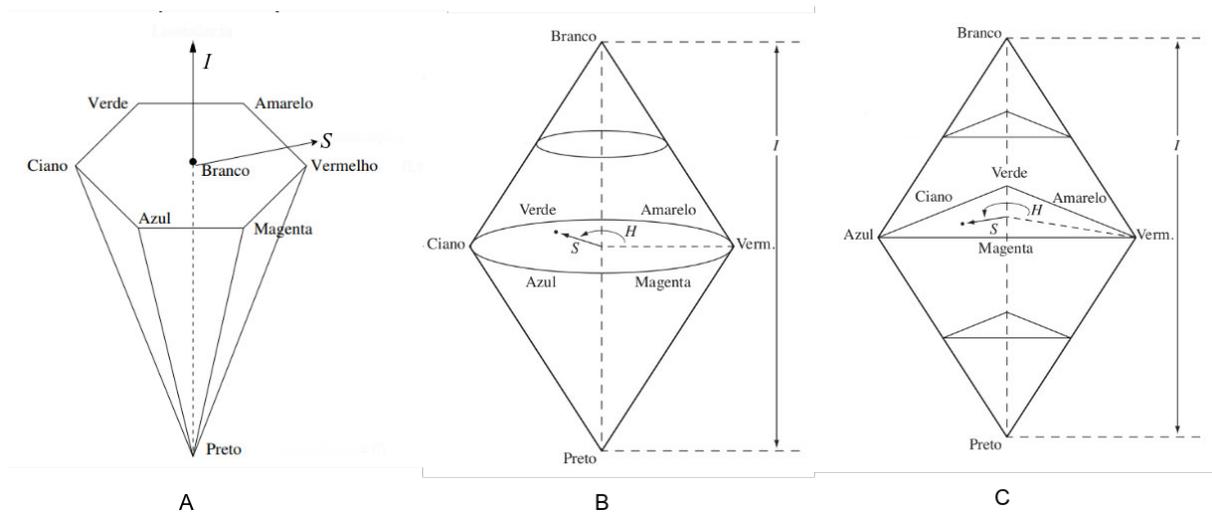


Fonte: Adaptado de (MARQUES FILHO; VIEIRA NETO, 1999).

Outro espaço de cores bastante conhecido é o HSV. Nele, é possível trabalhar separadamente com as componentes matiz (*Hue*), saturação (*Saturation*) e o valor do brilho da cor determinada por essas duas (*Value*). Esse espaço de cores traz uma percepção análoga à forma como o ser humano observa as cores, visto que a informação referente à cor (matiz e saturação) é separada da componente que define sua intensidade. Segundo

Gonzalez e Woods (2010), essas características fazem desse modelo uma ferramenta ideal para o desenvolvimento de algoritmos de processamento de imagens com base em descrições de cores que são naturais e intuitivas para os seres humanos, que, afinal, são os desenvolvedores e usuários desses algoritmos. Esse modelo possui as variantes HSI (*Hue, Saturation, Intensity*) e HLS (*Hue, Saturation, Lightness*). Ambos têm suas interpretações geométricas ilustradas pela Figura 5.

Figura 5 – Interpretação geométrica dos modelos HSV (A), HLS (B) e HSI (C).



Fonte: Adaptada de (GONZALEZ; WOODS, 2010).

O modelo HSV (Figura 5.A) pode ser visto como uma pirâmide hexagonal derivada do cubo RGB. As diferentes matizes são representadas na parte base hexagonal da pirâmide, onde cada vértice dessa base corresponde a uma cor primária RGB ou complementar (ciano, magenta, amarelo). Saturação é medida ao longo do eixo horizontal e Valor, pelo eixo vertical. Ambos apresentam valores entre 0 e 1. No caso da componente Valor, 0 corresponde à cor preta e 1 às intensidades máximas de cada cor. A conversão do espaço RGB para o HSV é feita por meio das equações a seguir:

$$H = \begin{cases} 60 \frac{(G-B)}{M-m} & , se \ M = R \\ 60 \frac{(B-R)}{M-m} + 120 & , se \ M = G \\ 60 \frac{(R-G)}{M-m} + 240 & , se \ M = B \end{cases} \quad (3.1a)$$

$$S = \begin{cases} \frac{(M-m)}{M} & , se \quad M \neq 0 \\ 0 & , caso contrário \end{cases} \quad (3.1b)$$

$$V = M \quad (3.1c)$$

onde  $m = \min(R, G, B)$  e  $M = \max(R, G, B)$ .

O espaço HLS (Figura 5.B), entretanto, assemelha-se geometricamente a dois cones de bases coincidentes. As matizes são pontos na borda da base circular; saturação varia conforme a distância ao eixo do cone e o brilho, no eixo comum aos cones. A diferença de interpretação, neste caso, ocasiona-se visto que no HLS as matizes puras são encontradas no plano em que a saturação assume o valor 0,5. Converte-se do RGB para o HLS com o uso das seguintes equações:

$$H = \begin{cases} 60 \frac{(G-B)}{M-m} & , se \quad M = R \\ 60 \frac{(B-R)}{M-m} + 120 & , se \quad M = G \\ 60 \frac{(R-G)}{M-m} + 240 & , se \quad M = B \end{cases} \quad (3.2a)$$

$$S = \begin{cases} 0 & , se \quad M = m \\ \frac{(M-m)}{M+m} & , se \quad 0 \leq L \leq 0,5 \\ \frac{(M-m)}{2-(M+m)} & , se \quad L > 0,5 \end{cases} \quad (3.2b)$$

$$L = \frac{M + m}{2} \quad (3.2c)$$

Por fim, o HSI (Figura 5.C) comporta-se, geometricamente, como uma pirâmide de 3 lados onde qualquer ponto da superfície representa uma cor puramente saturada. Por conseguinte, quanto mais distante uma cor estiver do centro do triângulo, maior será a sua saturação. O componente de intensidade no modelo HSI é medido de acordo com uma linha perpendicular ao triângulo, a qual cruza o seu centro. As equações abaixo expressam

a conversão do modelo RGB para o HSI.

$$H = \arccos \left( \frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)+(G-B)}} \right) \quad (3.3a)$$

$$S = 1 - \frac{3 * \min(R, G, B)}{R + G + B} \quad (3.3b)$$

$$I = \frac{R + G + B}{3} \quad (3.3c)$$

O modelo YCbCr consiste de 3 componentes: Y (Luminância), Cb e Cr (informações de cor). O componente Cb é dado pela diferença entre a cor azul e um certo valor de referência. O componente Cr, por fim, corresponde à diferença entre a cor vermelha e um referencial. A conversão do espaço de cores RGB para YCbCr é dada pela Equação 3.4.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,169 & -0,331 & 0,5 \\ 0,5 & -0,419 & -0,081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.4)$$

Outro modelo de cores conhecido é chamado XYZ. Nesse modelo, entende-se que as quantidades de vermelho, verde e azul necessárias para construir qualquer nova cor em específico são expressas por seus coeficientes tricromáticos X, Y e Z como mostrado pelas equações a seguir.

$$X = \frac{X}{X + Y + Z} \quad (3.5a)$$

$$Y = \frac{Y}{X + Y + Z} \quad (3.5b)$$

$$Z = \frac{Z}{X + Y + Z} \quad (3.5c)$$

A relação entre esse sistema de cores e o RGB é dada pela Equação 3.6.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0,490 & 0,310 & 0,200 \\ 0,177 & 0,813 & 0,010 \\ 0,000 & 0,010 & 0,990 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.6)$$

A partir do modelo XYZ, derivam os espaços de cores L\*a\*b e L\*U\*V. Em ambos, a componente L\* corresponde à luminância, enquanto que a crominância é dada pelas duas últimas. Diferem, todavia, no cálculo de suas respectivas crominâncias, como mostrado em 3.7.

$$L^* = \begin{cases} 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} & , se \quad \frac{Y}{Y_n} > 0,008856 \\ 903,3\left(\frac{Y}{Y_n}\right) & , se \quad \frac{Y}{Y_n} \leq 0,008856 \end{cases} \quad (3.7a)$$

$$u^* = 13L^*(u' - u'_n) \quad (3.7b)$$

$$v^* = 13L^*(v' - v'_n) \quad (3.7c)$$

$$a^* = 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \quad (3.7d)$$

$$b^* = 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right] \quad (3.7e)$$

$$u' = \frac{4X}{X + 15Y + 3Z} \quad (3.7f)$$

$$v' = \frac{9Y}{X + 15Y + 3Z} \quad (3.7g)$$

$$f(t) = \begin{cases} t^{\frac{1}{3}} & , se \quad t > 0,008856 \\ 7,787t + \frac{16}{116} & , caso \text{ contrário} \end{cases} \quad (3.7h)$$

onde  $X_n, Y_n e Z_n$  são os valores da iluminação de referência (fonte luminosa).

A partir das características apresentadas de cada espaço de cor, corrobora-se que o processamento de imagens em demais sistemas traz novas formas de analisar as características do objeto de estudo, colaborando, portanto, com o desenvolvimento de métodos mais eficazes.

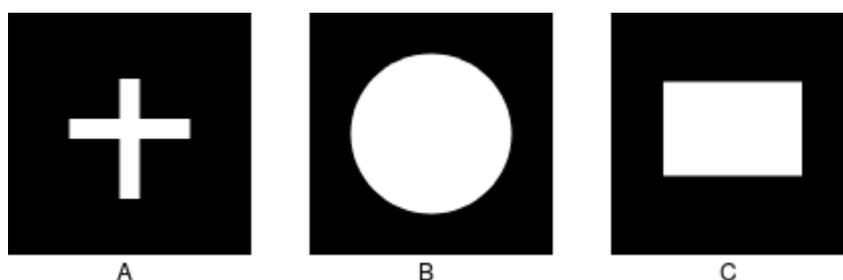
### 3.3.2 Morfologia Matemática

Esta seção apresenta os principais conceitos e operações no campo da Morfologia Matemática, essenciais para o entendimento de algumas etapas que compõem o método desenvolvido e apresentado neste trabalho.

A expressão morfologia diz respeito ao estudo da forma, das aparências externas de algo. Os estudos de Serra (1986) apresentam a morfologia matemática como o estudo da estrutura geométrica das entidades presentes em uma imagem. Estes conceitos têm aplicação em várias tarefas e com diferentes objetivos, tais como detecção de bordas, melhoramento, segmentação, entre outros.

A morfologia matemática tem como princípio básico a teoria de conjuntos. Consiste na extração de informações geométricas e topológicas de um conjunto desconhecido (a imagem) pela transformação através de outro conjunto completamente definido, chamado elemento estruturante. Logo, os resultados obtidos para cada transformação aplicada estão diretamente ligados a ele. Em nível de imagens, o elemento estruturante é uma entidade que pode assumir vários formatos, cuja escolha depende do processamento a ser realizado. Alguns podem ser visualizados pela Figura 6.

Figura 6 – Exemplos de Elementos Estruturantes.



Fonte: Elaborada pelo autor.

Os conceitos de morfologia matemática podem ser utilizados em imagens binárias ou em nível de cinza para diversos fins. Entretanto, essa seção discorre sobre as principais operações morfológicas utilizadas no desenvolvimento do método proposto neste trabalho, aplicados estritamente em imagens binárias.

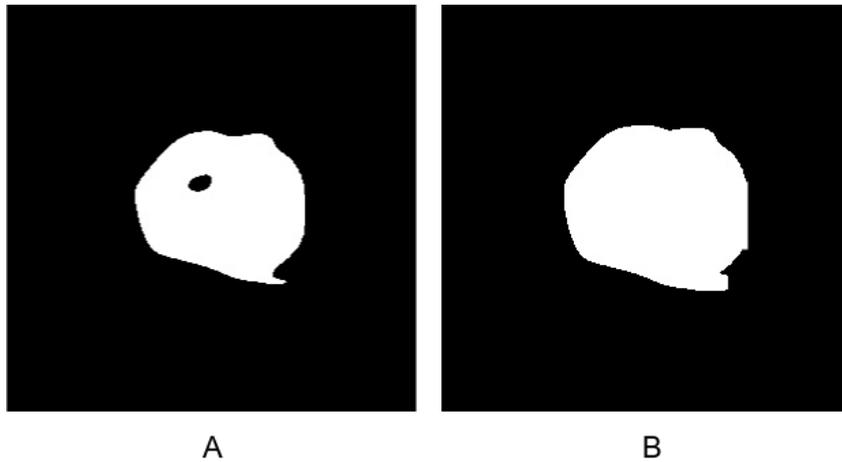
As duas operações básicas do processamento morfológico são a dilatação e a erosão. O entendimento delas facilita a compreensão das demais operações a serem explanadas posteriormente.

Sejam  $C$  e  $D$  conjuntos definidos no espaço  $Z^2$ . A dilatação de  $C$  por  $D$  é definida como:

$$C \oplus D = \{x | (\hat{D})_x \cap C \neq \emptyset\} \quad (3.8)$$

$D$  é o elemento estruturante. Esse processo consiste em rotacionar  $D$  em torno da origem para que se obtenha a sua reflexão e, assim, deslocá-lo de  $x$ . Portanto, a dilatação consiste em todos os  $x$  deslocamentos para os quais a interseção entre a reflexão de  $D$  e  $C$  inclui pelo menos um elemento diferente de zero (GONZALEZ; WOODS, 2010). O resultado da dilatação, em outras palavras, é o inchamento da imagem com o uso do elemento estruturante. Isso pode ser visualizado na Figura 7.

Figura 7 – Exemplo da operação de dilatação sobre uma imagem (A) resultando em (B). O elemento estruturante é retangular e tem dimensões 10 x 10.



Fonte: Elaborada pelo autor.

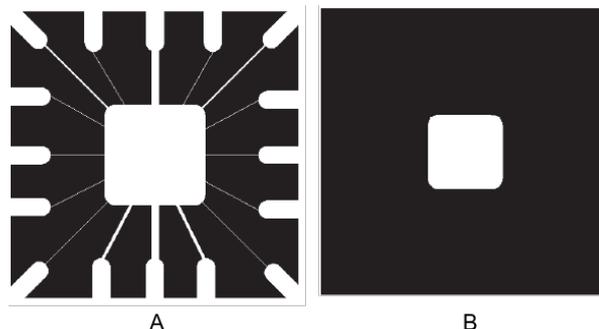
A outra operação morfológica a ser explanada é chamada de erosão. Sejam  $C$  e  $D$  conjuntos definidos no espaço  $Z^2$ , a erosão de  $C$  por  $D$  é definida como:

$$C \ominus D = \{x | (D)_x \subseteq C\} \quad (3.9)$$

Sendo  $D$  o elemento estruturante, a erosão de  $C$  por  $D$  resulta no conjunto de pontos  $x$  tais que  $D$ , transladado de  $x$ , está contido em  $C$ . Em nível de imagem, a erosão provoca o efeito homônimo nas entidades presentes, ou seja, diferente do inchaço provocado pela dilatação, ela ocasiona na diminuição do tamanho das estruturas, como ilustrado pela Figura 8.

É importante ressaltar que as operações de dilatação e erosão, quando aplicadas mais de uma vez na mesma imagem, têm seus efeitos ampliados. Destaca-se também que

Figura 8 – Exemplo da operação de erosão sobre uma imagem (A) resultando em (B). O elemento estruturante é um quadrado de dimensões 45 x 45.



Fonte: Adaptada de (GONZALEZ; WOODS, 2010).

os efeitos produzidos por essas depende diretamente da definição do elemento estruturante.

Os conceitos de dilatação e erosão são bastante importantes no que concerne ao entendimento das operações de abertura e fechamento apresentadas a seguir. Como visto anteriormente, a dilatação provoca a expansão da imagem e a erosão, por sua vez, ocasiona no encolhimento. As operações apresentadas nesta Seção podem ser compreendidas como a aplicação em conjunto daquelas supracitadas.

A abertura de um conjunto  $C$  com o elemento estruturante  $D$  é definida como:

$$C \circ D = (C \ominus D) \oplus D \quad (3.10)$$

Portanto, pode-se dizer que a abertura consiste em uma erosão seguida de dilatação. Já o fechamento de um conjunto  $C$  com o elemento estruturante  $D$  é definido como:

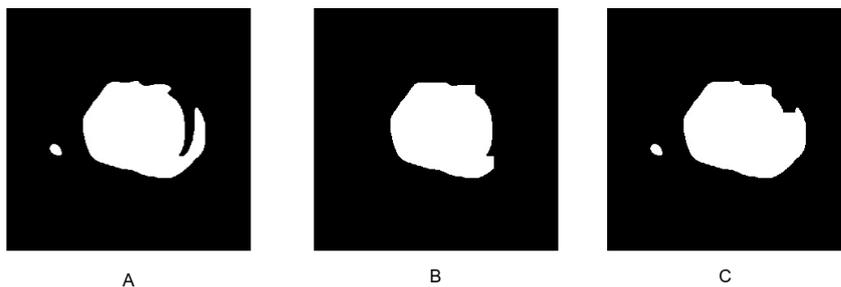
$$C \circ D = (C \oplus D) \ominus D \quad (3.11)$$

Desta feita, diz-se que o fechamento consiste em uma dilatação seguida de erosão.

Os efeitos ocasionados pela abertura, em geral, são a suavização de contornos, e separação de estruturas ligadas por finas linhas ou istmos. O fechamento, por sua vez, conecta estruturas separadas por pequenas quebras, eliminando também pequenos orifícios. A Figura 9 mostra alguns resultados para essas operações utilizando-se o mesmo elemento estruturante.

Destaca-se que a abertura e o fechamento satisfazem uma propriedade chamada idempotência (GONZALEZ; WOODS, 2010). Significa que não produzem efeitos acumulativos na imagem, diferentemente da dilatação e da erosão. Logo, a aplicação sucessiva

Figura 9 – Operações aplicadas sobre a imagem (A). Abertura resulta em (B). Fechamento, em (C). O elemento estruturante é um quadrado de dimensões 15 x 15.



Fonte: Elaborada pelo autor.

de abertura ou fechamento em uma mesma imagem não irá alterá-la, a menos que haja outros processamentos entre as suas utilizações. Novamente, convém observar que os resultados produzidos por essas técnicas dependem do elemento estruturante utilizado.

### 3.4 Descritores de Características

#### 3.4.1 Baseados em Textura

É comum que as imagens contenham regiões caracterizadas não só por uma única intensidade ou cor, mas por mudanças nos valores dessas propriedades. Essa variação de intensidade entre *pixels* vizinhos ou regiões vizinhas é chamada de textura. Sendo a imagem interpretada como uma superfície, os valores assumidos pelos *pixels* são analogamente vistos como elevações. Portanto, sob esta ótica, a textura mensura a rugosidade superficial (RUSS, 2006).

Há três abordagens de análise de textura utilizadas em processamento de imagem, que são categorizadas como: métodos estatísticos, estruturais e espectrais (GONZALEZ; WOODS, 2010). As abordagens estatísticas avaliam as texturas como suave, rugosa, entre outras. As técnicas estruturais, por sua vez, trabalham com arranjos de primitivas de imagens, como a descrição de textura baseada em linhas paralelas espaçadas regularmente. E as técnicas espectrais estão baseadas em propriedades do espectro de Fourier.

Neste trabalho, utiliza-se a abordagem estatística para análise das características de textura das imagens de interesse. Essa abordagem define a textura por meio da extração de um conjunto de medidas que descrevem as imagens através de regras estatísticas, as quais administram tanto a distribuição quanto a relação entre os diferentes níveis de intensidade. Para a análise de textura, este trabalho utiliza duas técnicas: matrizes de

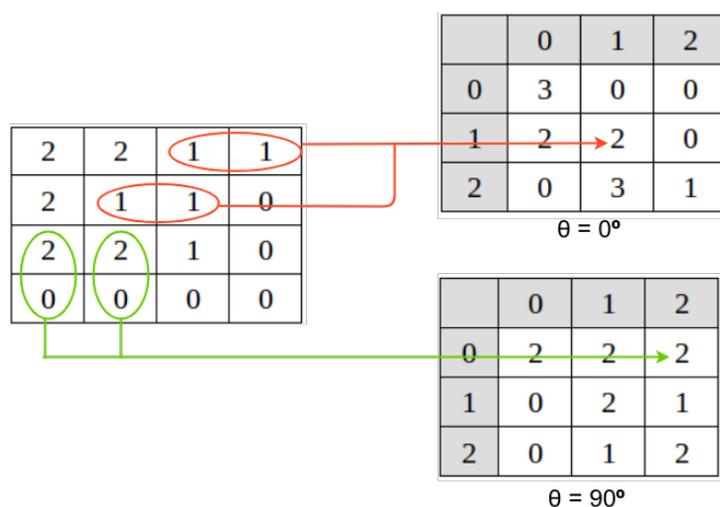
coocorrência em nível de cinza e em nível de cores.

### 3.4.1.1 Matrizes de Coocorrência de Níveis de Cinza

Uma forma de descrever a textura consiste em considerar tanto a distribuição de intensidades, como também as posições relativas dos *pixels* em uma imagem. Dessa forma, considera-se um operador  $P$  que define a posição um *pixel* em relação ao outro. Considera-se também uma imagem  $f$  com  $L$  níveis de intensidade possíveis. Diz-se, portanto, que  $M$  é uma matriz na qual o elemento  $m_{ij}$  é a quantidade de vezes em que a intensidade  $i$  ocorre próxima da intensidade  $j$  em  $f$ , na posição definida por  $P$ , sendo que  $i, j \leq L$ . A essa matriz, dá-se o nome de matriz de coocorrência em nível de cinza (GLCM - *Gray-Level Co-occurrence Matrix*).

Para uma imagem, várias matrizes de coocorrência podem ser extraídas dependendo da configuração do operador  $P$ . Como dito, esse operador define a posição relativa dos *pixels*. Para isso, considera-se tanto a direção quanto na distância entre os *pixels* verificados. No exemplo ilustrado pela Figura 10, tem-se algumas configurações definidas e GLCM geradas para uma imagem com intensidades  $L = 3$ .

Figura 10 – Matrizes de coocorrência em nível de cinza geradas para uma imagem.



Fonte: Elaborada pelo autor.

Observa-se na Figura 10 que para cada direção  $\theta$  de checagem foi gerada uma GLCM. Em ambas, o critério de distância entre pixels é  $d=1$ . Mas este parâmetro também é passível de alterações. Logo, a utilização dessa técnica provê múltiplas representações de textura e, por conseguinte, mais insumos para a análise estatística.

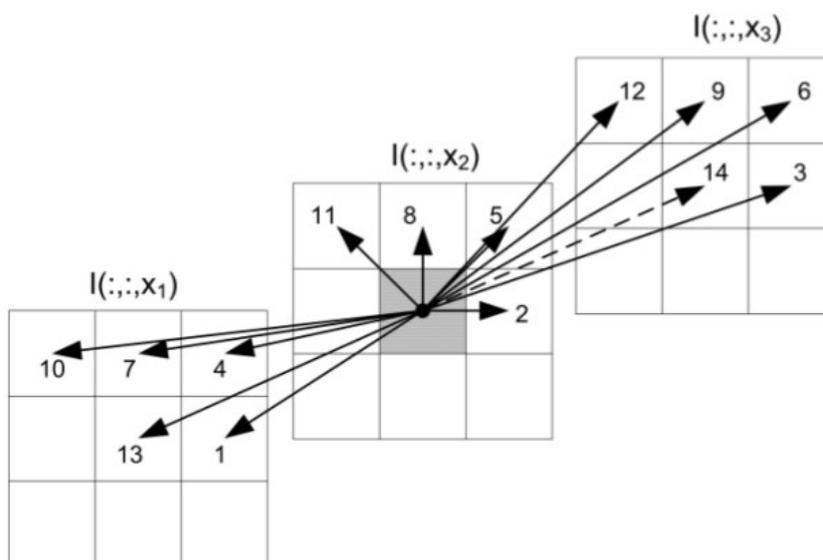
No presente trabalho, as GLCM são calculadas para cada componente em vários espaços de cores distintos.

### 3.4.1.2 Matrizes de Coocorrência de níveis de Cores

As matrizes de coocorrência em nível de cinza têm vasta aplicação em análise de textura, garantindo também resultados promissores em alguns campos. Baseado nisso, Benco et al. (2014) propuseram uma nova técnica chamada CLCM (*Colour-Level Co-occurrence Matrix*), em tradução livre, matrizes de coocorrência em níveis de cores.

A principal diferença está no relacionamento entre *pixels* considerado para cálculo das coocorrências. Na técnica GLCM, como visto na seção anterior, define-se a direção e a distância entre *pixels* e com isso extraem-se matrizes de uma mesma imagem em nível de cinza. As CLCM são, entretanto, calculadas levando-se em consideração as componentes da imagem colorida e, por conseguinte, a posição relativa dos *pixels* nas três imagens. Este processo pode ser visualizado na Figura 11.

Figura 11 – Matrizes de coocorrência em nível de colorido geradas para uma imagem colorida.



Fonte: Adaptado de (BENCO et al., 2014).

Em imagens coloridas, um *pixel* em uma posição  $(k,l)$  é representado por três valores  $I(k,l, X_1)$ ,  $I(k,l, X_2)$ ,  $I(k,l, X_3)$ , sendo  $I$  a intensidade. Tais valores compõem a representação tridimensional do *pixel*. Sendo assim, para uma distância  $d = 1$  e ângulos  $\theta = 0^\circ, 45^\circ, 90^\circ e 135^\circ$ , cria-se um cubo que representa a vizinhança do *pixel* de referência nas

mesmas posições, porém em componentes distintas. Nesta técnica, leva-se em consideração as 13 direções em que serão analisadas as coocorrências. Desconsidera-se a decima quarta devido a redundância de informações (BENCO et al., 2014). Assim como nas GLCMs, para cada direção é gerada uma matriz. Logo, para cada imagem, no mínimo 13 matrizes serão geradas.

Ressalta-se que, para que se obtenha mais informações a respeito do relacionamento entre as componentes, é válida a permutação entre os canais. Assim, muda-se a componente de referência, gerando novos conjuntos de 13 matrizes.

### 3.4.2 Baseados em cores

Outra abordagem empregada para a extração de características de imagens utilizada no desenvolvimento deste trabalho é a análise de cores. A cor é uma informação de extrema importância em imagens. É, portanto, um poderoso descritor. As técnicas de análise de cores são bastante utilizadas em sistemas de recuperação de imagens do tipo CBIR - (*Color-Based Image Retrieval*) por serem invariantes à escala, rotação e translação (SALMI; BOUCHEHAM, 2014).

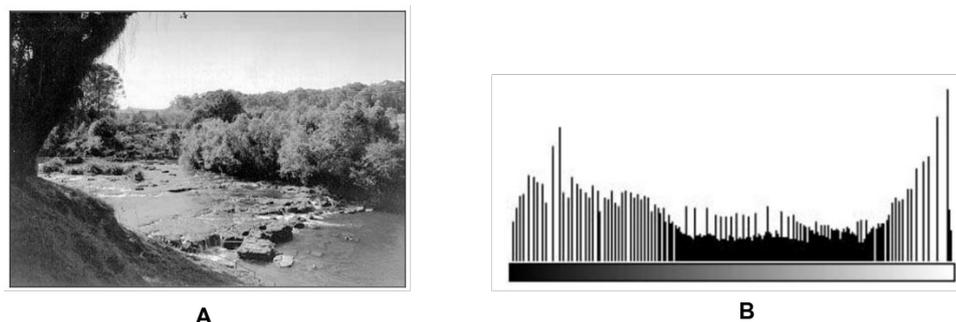
Neste trabalho, as técnicas para a descrição das características de cor utilizadas são baseadas no histograma de cores e suas variantes. Os conceitos e demais propriedades a respeito dessa técnica serão discutidos ao longo desta seção.

#### 3.4.2.1 Histograma de cores

Gonzalez e Woods (2010) conceituam o histograma de uma imagem com níveis de intensidade no intervalo  $[0, L - 1]$  como uma função discreta  $h(r_k) = n_k$ , onde  $r_k$  é o  $k$ -ésimo valor de intensidade e  $n_k$  é o número de *pixels* da imagem com intensidade  $r_k$ . Em outras palavras, um histograma de uma imagem é um conjunto de números indicando o percentual de *pixels* naquela imagem que apresentam uma determinada intensidade. Estes valores são normalmente representados por um gráfico de barras onde, para cada intensidade, é fornecida a quantidade (ou o percentual) de *pixels* correspondentes na imagem (MARQUES FILHO; VIEIRA NETO, 1999). A Figura 12 ilustra o histograma de uma imagem em nível de cinza.

Partindo das definições supracitadas, percebe-se que o histograma pode fornecer informações estatísticas úteis para o processamento de imagens. Neste trabalho, os

Figura 12 – Representação de uma imagem em nível de cinza (A) pelo seu histograma (B).



Fonte: Adaptado de (MARQUES FILHO; VIEIRA NETO, 1999).

histogramas de cores são utilizados para a descrição das características de cor apresentadas pelas imagens do Teste de Brückner. Esta é mais uma forma de avaliar o comportamento do reflexo retiniano que, como citado em seções anteriores, tem sua qualidade verificada pela análise da cor que apresenta.

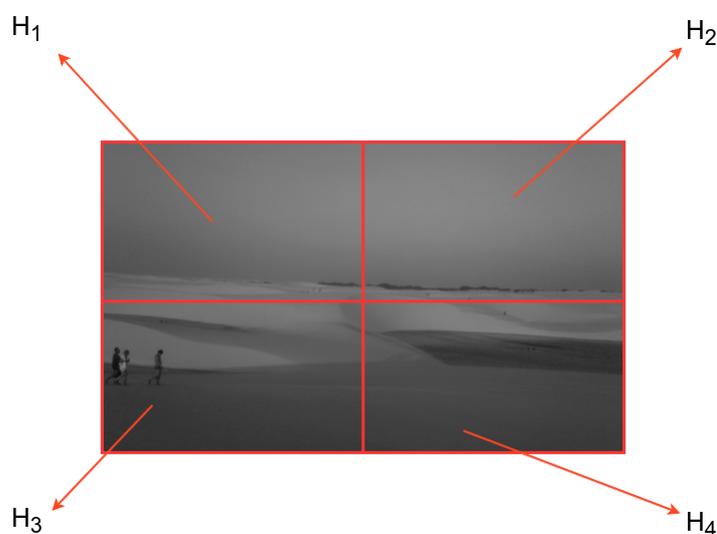
Os histogramas de cores são assim chamados por causa da sua aplicação. Isto é, dada uma imagem representada em um sistema de cores, será computado um histograma para cada uma de suas componentes para fins de quantificação de suas intensidades correspondentes.

Essa técnica pode ser aplicada em dois níveis de processamento: global e local. Em suma, histogramas globais de cores (HGC) são calculados levando-se em consideração toda a imagem, de forma análoga ao ilustrado pela Figura 12. Já para o cálculo dos histogramas locais de cores (HLC), deve-se dividir a imagem em regiões seguindo alguma estratégia, sendo que cada região produzirá seu próprio histograma. Esse passo é ilustrado pela Figura 13.

HGCs são rápidos e fáceis de calcular. Também são invariantes a rotação e translação. Não obstante, não fornecem informações sobre a distribuição espacial das cores (WAN; JIN; YUE, 2011). Para incluir essas informações na análise, os histogramas locais de cores são uma opção melhor, dada a condição de serem calculados por região da imagem. Sendo assim, se a estratégia de divisão da imagem fornece  $N$  sub-regiões, serão produzidos  $N$  histogramas. HLCs superam as dificuldades dos HGCs, porém tornam-se mais sensíveis à rotação e translação (SALMI; BOUCHEHAM, 2014).

O processo de representação das características com HGC irá produzir um histograma para cada componente, os quais podem ser concatenados para que se tenha

Figura 13 – Divisão de uma imagem em quatro blocos. Calcula-se um histograma para cada bloco.



Fonte: Elaborada pelo autor.

uma informação completa a cerca da imagem. O mesmo se dá com HLCs, mas com um passo anterior que é a divisão em regiões. Neste caso, a concatenação de histogramas produz uma representação maior que o HGC. Histogramas de cores podem ser interpretados diretamente como os vetores de características de uma imagem. Mas, assim como as matrizes de coocorrência, podem servir apenas como base para uma análise estatística mais robusta, com a extração de métricas capazes de melhorar a descrição fornecida por eles.

### 3.4.3 Abordagem Estatística

A descrição de características tanto de textura quanto de cores consistem na criação de representações das imagens contendo o tipo de informação requerida. A etapa de análise estatística, portanto, transforma essa informação em vetores de característica por meio da utilização de meios matemáticos, objetivando, por meio dessa nova representação, descrever os padrões de forma mais simplificada. Estes artifícios matemáticos são apresentados nesta seção.

Para a extração de características das matrizes de coocorrência do tipo GLCM e CLCM, são utilizados os descritores de Haralick (HARALICK; SHANMUGAM et al., 1973). Essa técnica tem sido amplamente utilizada em aplicações de análise de imagens, especialmente na área da biomedicina (ZAYED; ELNEMR, 2015).

São utilizadas, ao todo, 13 métricas. Para o entendimento dos cálculos, as seguintes definições são necessárias:  $L$  é o número máximo de intensidade das imagens, logo, as dimensões das matrizes são  $L \times L$ ;  $p(i,j)$  é o valor presente na matriz nos índices  $i$  e  $j$ .

O primeiro descritor a ser apresentado é o segundo momento angular (SMA), que mede a uniformidade local dos níveis de cinza. A energia é outra métrica utilizada, e deriva do SMA. Seus cálculos são realizados, respectivamente, como mostrado pelas equações 3.12 e 3.13.

$$SMA = \sum_i \sum_j p(i, j)^2 \quad (3.12)$$

$$Energia = \sqrt{SMA} \quad (3.13)$$

O contraste é a medida das variações dos níveis de cinza entre o pixel de referência e seus vizinhos. A correlação, por sua vez, analisa a dependência linear entre as intensidades na matriz de coocorrência.

$$Contraste = \sum_i \sum_j (i - j)^2 p(i, j) \quad (3.14)$$

$$Correlação = \sum_i \sum_j p(i, j) \frac{(i - \mu_x)(j - \mu_y)}{\sigma_x \sigma_y} \quad (3.15)$$

onde  $\mu_x, \mu_y, \sigma_x, \sigma_y$  são médias e desvios padrões calculados por:

$$\mu_x = \sum_i \sum_j i p(i, j) \quad (3.16)$$

$$\mu_y = \sum_i \sum_j j p(i, j) \quad (3.17)$$

$$\sigma_x = \sqrt{\sum_i \sum_j (i - \mu_x)^2 p(i, j)} \quad (3.18)$$

$$\sigma_y = \sqrt{\sum_i \sum_j (j - \mu_y)^2 p(i, j)} \quad (3.19)$$

Outras medidas são a homogeneidade, que calcula a autocorrelação espacial (Equação 3.20); a variância e a soma das médias, que são calculadas como mostram as equações 3.21 e 3.22. A entropia, que mede a aleatoriedade das entradas da matriz, é dada pela Equação 3.23.

$$\text{Homogeneidade} = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i, j) \quad (3.20)$$

$$\text{Var} = \sum_i \sum_j (i - \mu)^2 p(i, j) \quad (3.21)$$

$$SM = \sum_{i=2}^{2L} i P_{x+y}(i) \quad (3.22)$$

$$\text{Entropia} = - \sum_i \sum_j p(i, j) \log(p(i, j)) \quad (3.23)$$

onde,

$$p_{x+y}(k) = \sum_{i=1}^L \sum_{j=1}^L p(i, j), \quad i + j = k, \quad e \quad k = 2, 3 \dots 2L \quad (3.24)$$

$$p_{x-y}(k) = \sum_{i=1}^L \sum_{j=1}^L p(i, j), \quad |i - j| = k, \quad e \quad k = 0, 1 \dots L - 1 \quad (3.25)$$

sendo  $x$  e  $y$ , respectivamente, linha e coluna da matriz.

Também utiliza-se as entropias da soma (ES) e da diferença (ED), variâncias da soma (VS) e da diferença (VD), e probabilidade máxima, respectivamente apresentadas pelas equações a seguir:

$$ES = - \sum_{i=2}^{2L} p_{x+y}(i) (\log(p_{x+y}(i))) \quad (3.26)$$

$$ED = - \sum_{i=0}^{L-1} p_{x-y}(i) (\log(p_{x-y}(i))) \quad (3.27)$$

$$VS = \sum_{i=2}^{2L} (i - ES)^2 p_{x+y}(i) \quad (3.28)$$

$$VD = \text{variância de } p_{x-y} \quad (3.29)$$

$$\text{Prob.Max} = \max p_{i,j} \quad (3.30)$$

Para a análise dos histogramas de cores globais e locais, são utilizadas métricas similares, porém como esta representação é unidimensional, os cálculos precisam ser mapeados para atender essa característica.

Para a extração de características dos histogramas, utiliza-se a média, mediana e moda, que são medidas amplamente conhecidas no ramo da estatística. Também é calculada a entropia como já mostrado pela Equação 3.23. As outras medidas são: amplitude, coeficiente de variação e o quadrado médio de diferenças sucessivas.

A amplitude corresponde à diferença entre o maior e o menor valor do histo-

grama. O coeficiente de variação (CV) e o quadrado médio de diferenças sucessivas são apresentados (MSSD) pelas equações 3.31 e 3.32.

$$CV = \frac{1}{M} V^{0,5} \quad (3.31)$$

$$MSSD = \sum_{i=0}^L \frac{(p(i+1)-p(i))^2}{2(L-1)} \quad (3.32)$$

### 3.5 Redes Neurais

Redes Neurais Artificiais (RNA) é um modelo matemático representado por um conjunto de unidades computacionais interligadas, que realizam processamento simples, cuja funcionalidade é baseada no funcionamento dos neurônios. Esses modelos são amplamente utilizados em reconhecimento de padrões.

A unidade básica de processamento de uma RNA é chamada de neurônio artificial. Ele representa uma função matemática em que, para um dado conjunto de entradas, é fornecida uma saída, como expresso pela Equação 3.33.

$$f(x) = \sigma \left( \sum_{i=1}^N x_i w_i + b \right) \quad (3.33)$$

onde  $x_i$  é a entrada  $i$ ,  $w_i$  é o peso associado a entrada  $i$ ,  $b$  é o termo *bias* e  $\sigma$  é a função de ativação.

Existem as funções de ativação lineares e não-lineares. Em relação a esta última categoria, citam-se como exemplos a função sigmoide, tangente hiperbólica e as recentemente propostas *Rectified Linear Units* (ReLU) e a *Leaky ReLU*, cuja velocidade de convergência em relação às supracitadas é superior (MAAS; HANNUN; NG, 2013). Essas funções são expressas respectivamente pelas equações 3.34 e 3.35.

$$f(x) = \max(0, x) \quad (3.34)$$

$$f(x) = \max(0.1, x) \quad (3.35)$$

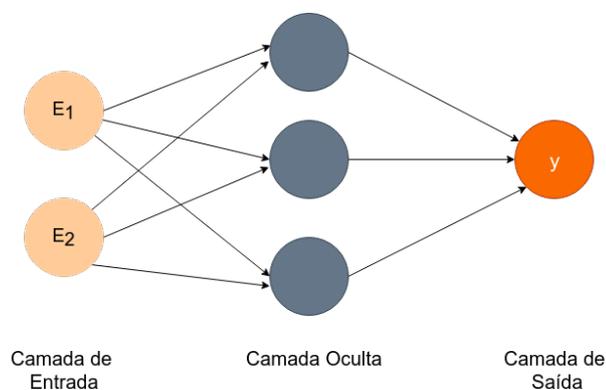
#### 3.5.1 Redes *Multilayer Perceptron*

*Multilayer Perceptron* (MLP) é um tipo de RNA composta por múltiplas camadas de neurônios, todas completamente conectadas à sua subsequente. Os neurônios, por sua vez, implementam as funções de ativação determinadas. Essas redes, de acordo com Hornik,

Stinchcombe e White (1989), são capazes de aproximar qualquer função mensurável para qualquer grau de precisão desejado. Consiste em uma técnica de aprendizado supervisionado.

Em linhas gerais, uma MLP é caracterizada por conter uma camada inicial, que recebe os dados de entrada do modelo, uma ou mais camadas intermediárias, chamadas de camadas ocultas ou escondidas e a camada de saída. Essas redes são enquadradas na categoria *feedforward*, ou seja, não possuem ligações retroalimentativas. Logo, a saída de uma camada nunca será utilizada como entrada dessa mesma ou de antecessoras. A Figura 14 apresenta a arquitetura básica de uma MLP.

Figura 14 – Rede *Multilayer Perceptron*.



Fonte: Elaborada pelo autor.

Para o treinamento de uma MLP, uma função de erro é definida a fim de calcular o erro das predições realizadas com base na saída esperada para os dados de entrada fornecidos. O treinamento objetiva minimizar o erro total da rede. Para isso, as redes MLP utilizam um algoritmo nomeado *Backpropagation*. Esse algoritmo é caracterizado por dois passos distintos. Primeiro, dados são apresentados à camada de entrada da rede. Eles seguem através da rede, camada por camada, sendo transformados por meio dos cálculos realizados até que, na camada de saída, seja produzida uma resposta. Em seguida, a saída obtida é comparada à saída desejada, obtendo-se um valor de erro. Esse erro é propagado partindo da camada de saída até a camada de entrada, modificando, dessa forma, os pesos das conexões nas camadas ocultas conforme a continuidade desse processo.

As etapas do algoritmo *backpropagation* são apresentadas a seguir:

1. Inicializar os valores dos pesos sinápticos de cada neurônio com valores aleatórios.

2. Apresentar as entradas da rede em uma representação vetorial  $e_1, e_2, \dots, e_n$  de características. E informar os vetores  $d_1, d_2, \dots, d_n$  de saídas desejadas.
3. Calcular as saídas da rede de acordo com a Equação 3.33.
4. Reajustar os pesos a partir da camada de saída em direção à camada de entrada, de acordo com a Equação 3.36:

$$w_{ij}(t+1) = w_{ij}t + \eta \delta_j x_i \quad (3.36)$$

onde  $w_{ij}$  é o peso do neurônio  $j$  em uma iteração  $t$ ,  $x_i$  corresponde a um neurônio da saída ou da entrada,  $\eta$  é a taxa de aprendizagem e  $\delta_j$  um termo de erro para o neurônio  $j$ , calculado como segue:

$$\delta_j = \begin{cases} y_j(1 - y_j)(d_j - y_j), & \text{se } j \text{ é um neurônio de saída} \\ x_j(1 - x_j) \sum_k \delta_k w_{jk}, & \text{se } j \text{ é um neurônio oculto} \end{cases} \quad (3.37)$$

5. Retorna para o passo 2.

É importante ressaltar que a taxa de aprendizado ( $\eta$ ) implica fortemente no desempenho do aprendizado. Uma taxa grande tende a produzir grandes oscilações na curva de aprendizado da rede. Taxas muito pequenas tendenciam os resultados a mínimos locais, retardando o aprendizado. Com a aplicação desse algoritmo, MLPs tornam-se capazes de generalizar o conhecimento aprendido por meio de exemplos a um conjunto de dados desconhecido (MAIA, 2018).

### 3.6 Aprendizagem Profunda

Aprendizado profundo é uma sub-área do aprendizado de máquina a qual representa uma tentativa de aprendizado de abstrações de alto nível em um conjunto de dados. Essa área tem sido amplamente estudada em visão computacional e, conseqüentemente, muitas propostas têm sido desenvolvidas (REDMON et al., 2015; SZEGEDY; IOFFE; VANHOUCHE, 2016).

O conceito de aprendizado profundo se origina no estudo das redes neurais artificiais. Como visto na Seção 3.5.1, RNAs são constituídas por unidades chamadas de

neurônios, os quais implementam funções de ativação e, durante o treinamento, os pesos das conexões são atualizados via *backpropagation*. Dependendo do problema, uma rede pode precisar de muitas camadas para que se alcance resultados satisfatórios, fator que retarda o seu treinamento e a torna mais suscetível ao *overfitting* (LIU et al., 2017).

Os algoritmos de aprendizagem profunda também se apresentam como arquiteturas hierárquicas contendo múltiplas camadas, cada qual contendo sua unidade de processamento não-linear. Diferente das RNAs tradicionais, essas técnicas permitem a extração automática de características a partir do conjunto de treinamento, sem a necessidade de uma série de técnicas de processamento de imagens e reconhecimento de padrões (HUA et al., 2015). Desta feita, a intervenção humana no processo é bastante diminuta.

Neste trabalho, dar-se-á foco para duas abordagens de aprendizado profundo: redes neurais convolucionais (CNN - *Convolutional Neural Networks*) e redes totalmente conectadas (FCN - *Fully Connected Networks*). Essas técnicas são utilizadas, pelo método proposto, nas etapas de detecção dos olhos (Seção 4.2.1) e segmentação do reflexo retiniano (Seção 4.2.2).

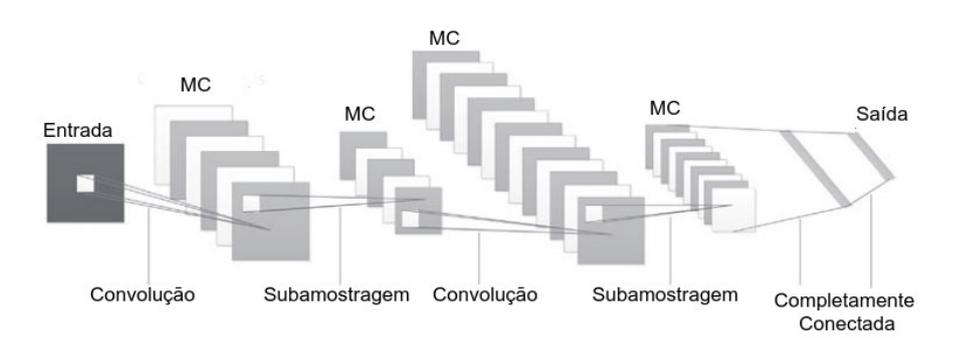
### 3.6.1 Redes Neurais Convolucionais

As redes neurais convolucionais (CNN - *Convolutional Neural Networks*) são modelos biologicamente inspirados capazes de construir um aprendizado hierárquico de características (LECUN et al., 1998). CNNs são um tipo de arquitetura profunda discriminativa (AREL; ROSE; KARNOWSKI, 2010), mostrando desempenho satisfatório no processamento de dados bidimensionais, como imagens e vídeos.

Goodfellow, Bengio e Courville (2016) enumeram três conceitos chave para o entendimento das CNNs: interação esparsa, compartilhamento de parâmetros e representação equivariante. A interação esparsa por meio de *kernels* de dimensões pequenas aplicados sobre toda a imagem de entrada numa operação chamada de convolução, em substituição às multiplicações de matriz inerentes às RNAs tradicionais. O compartilhamento de parâmetros é uma habilidade que permite o aprendizado de somente parte do total de parâmetros de uma CNN, permitindo que a outra parte permaneça "congelada", flexibilizando o treinamento. Essa característica possibilita a identificação de outra propriedade chamada equivariância, na qual, dadas as mudanças no conjunto de entrada, a saída mudará da mesma forma.

A CNN geralmente utiliza em sua arquitetura três tipos de camadas: convolução, subamostragem e completamente conectada (GUO et al., 2016). As camadas de convolução e subamostragem compõem as camadas iniciais da rede, e são implicitamente responsáveis pela extração de características. A camada completamente conectada, por fim, localiza-se no final da rede, e é responsável pela classificação dos padrões. A Figura 15 ilustra uma arquitetura básica de CNN.

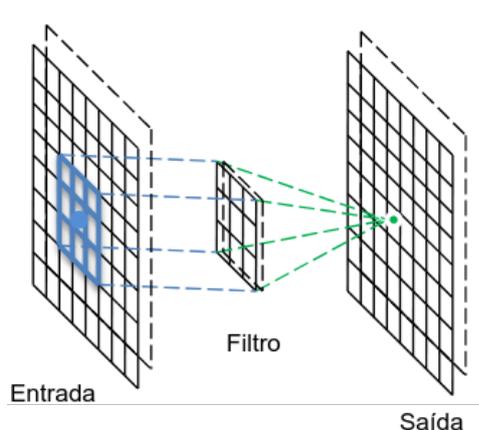
Figura 15 – Arquitetura básica de uma CNN contendo convoluções, subamostragens, mapas de características (MC) e a camada completamente conectada.



Fonte: Adaptado de (LIU et al., 2017).

A camada de convolução é composta por filtros treináveis aplicados sobre a imagem de entrada, gerando, portanto, vários mapas de características (GUO et al., 2016). Ao fim do treinamento, cada filtro será responsável pela detecção de uma característica específica da imagem, ou de parte dela. A Figura 16 ilustra esse processo.

Figura 16 – Ilustração da camada de convolução.

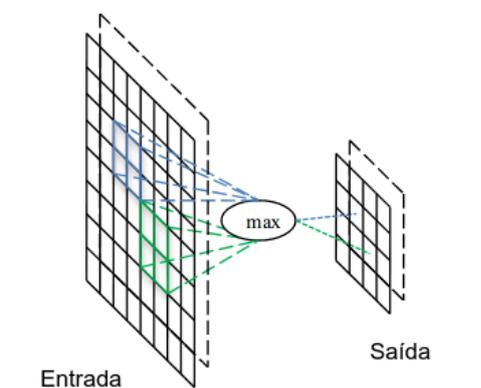


Fonte: Adaptado de (GUO et al., 2016).

As camadas de subamostragem são subsequentes às de convolução, e podem ser utilizadas para reduzir as dimensões dos mapas de características. Com funcionamento

similar às camadas de convolução, filtros são aplicados sobre os mapas, extraindo valores (amostras) dessas sub-regiões pela utilização de operações como média aritmética (*Average Pooling*) ou valor máximo (*Max Pooling*). Este último, por exemplo, consiste na extração do valor máximo em uma janela de convolução, como apresentado através da Figura 17.

Figura 17 – Camada de subamostragem com a aplicação da operação *Max Pooling*.



Fonte: Adaptado de (GUO et al., 2016).

Por fim, a camada completamente conectada consiste de uma estrutura similar às MLPs, em que os neurônios de entrada recebem os *pixels* dos mapas de característica gerados pela camada anterior, e geram a classificação dos padrões extraídos.

Neste trabalho, são utilizadas duas arquiteturas baseadas nos conceitos de redes neurais convolucionais: YOLO e U-Net. A primeira está presente na etapa de localização dos olhos (Seção 4.2.1). E a segunda é usada no processo de segmentação automática do reflexo retiniano (Seção 4.2.2).

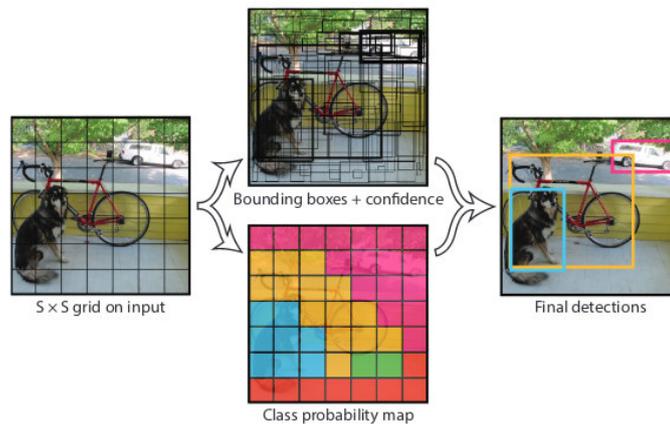
### 3.6.2 YOLO - *You Only Look Once*

A YOLO é uma arquitetura de rede neural elaborada especialmente para a detecção de objetos. Essa tarefa consiste em determinar as localizações de cada objeto presente na imagem. Proposta por Redmon et al. (2015), a YOLO trata a detecção de objetos como um problema de regressão, cujo resultado é a delimitação de *bounding boxes* baseada na probabilidade de determinado objeto pertencer a uma classe em específico. A YOLO é conhecida por apresentar resultados bastante satisfatórios em cenários complexos (KRAWCZYK; STARZYRNSKI, 2018) e em tempo real (YANG; JIACHUN, 2018; LAN et al., 2018).

A YOLO utiliza características de toda a imagem para prever as *bounding boxes*. A ideia consiste em dividir a imagem em grade com  $S \times S$  células. Cada célula realiza predição de  $B$  *bounding boxes* e fornece uma taxa de confiança quanto à esta predição. Em relação às *bounding boxes*, elas representam cinco valores:  $x, y, w, h$  correspondendo, respectivamente, às coordenadas  $x, y$  da sua origem, largura e comprimento. O quinto valor é a medida *intersection over union* (IoU) calculada entre a predição e conjunto *ground truth*.

A taxa de confiança fornecida pela célula corresponde à probabilidade do objeto estar contido naquela região, considerando-se também se essa região possui as dimensões adequadas. Essa medida é calculada por  $Pr(Object) * IoU_{predict}^{truth}$ .

Figura 18 – Divisão da imagem em grade para a predição de *bounding boxes*.



Fonte: Adaptado de (REDMON et al., 2015).

A probabilidade condicional de cada classe é dada por  $Pr(Class_i|Object)$ . Um conjunto de probabilidades condicionais são calculadas por célula, a despeito da quantidade de *boundibg boxes*. Por fim, a taxa de confiança para uma classe específica é calculada pela multiplicação entre as probabilidades condicionais de classe e de cada *bounding box*, como expresso pela expressão 3.38. Essa taxa indica a probabilidade de uma classe ocorrer em uma região da imagem, bem como o quão acurada é a *bounding box* gerada (REDMON et al., 2015). A Figura 18 ilustra como a YOLO realiza a predição de objetos.

$$Pr(Class_i|Object) * Pr(Object) * IoU_{predict}^{truth} = Pr(Class_i) * IoU_{predict}^{truth} \quad (3.38)$$

Neste trabalho, é utilizada a terceira versão da YOLO, chamada YOLOv3 (REDMON, 2018). Um conceito característico desta rede é o de âncoras, os quais estão diretamente ligados à predição. Âncoras são *bounding boxes* previamente definidas para cada célula, e serão, durante o treinamento, ajustadas para obter a melhor concordância com o *ground truth*. Esse mecanismo desacopla os mecanismos de predição de classes a partir de localizações espaciais (REDMON; FARHADI, 2017). Posto isso, a saída da rede deve conter 4 valores para cada *bounding box*  $t_x, t_y, t_w, t_h$ . Sabendo-se que a célula é uma região transladada da origem da imagem por  $(c_x, c_y)$  e que os âncoras possuem dimensões  $p_w, p_h$ , as predições correspondem a:

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$

onde  $b_x, b_y, b_w, b_h$  são as coordenadas x,y do centro, largura e comprimento da *bounding box* predita.  $\sigma$  corresponde à função sigmoide, forçando a normalização da saída entre 0 e 1.

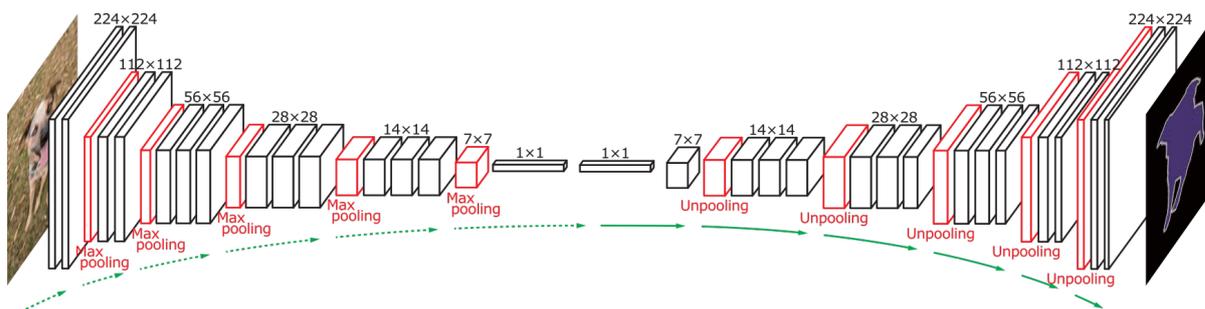
Neste trabalho, é utilizada a arquitetura YOLOv3 como proposta em Redmon (2018). Mais detalhes dessa arquitetura serão apresentados na Seção 4.2.1.

### 3.6.3 U-Net: Rede Totalmente Convolutacional

As CNNs são utilizadas tipicamente em tarefas de classificação. Dada uma imagem de entrada, as camadas de convolução e subamostragem irão gerar mapas de características que, hierarquicamente, representarão os padrões identificados e servirão de insumo ao processo de classificação realizado pela camada completamente conectada.

Para a tarefa de segmentação, no entanto, são necessárias informações de localização, ou seja, indicar quais *pixels* compõem ou não o objeto de interesse na imagem. Para essa finalidade, Long, Shelhamer e Darrell (2015) desenvolveram o conceito de Redes totalmente convolucionais (FCN - *Fully Connected Networks*), que são arquiteturas projetadas para a segmentação de imagens. A Figura 19 ilustra uma arquitetura básica de FCN.

Figura 19 – Ilustração da arquitetura de uma rede completamente convolucional.



Fonte: Adaptada de (NOH; HONG; HAN, 2015).

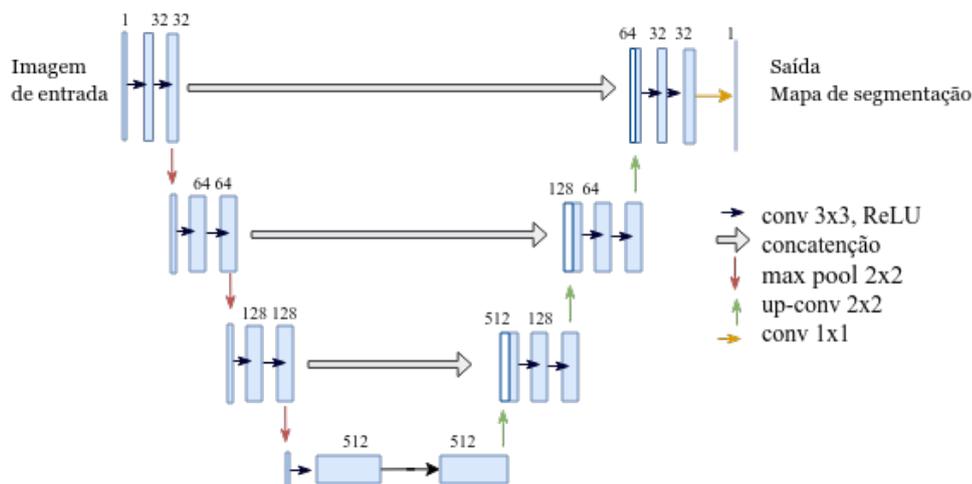
Como visto na Figura 19, as FCNs são compostas por dois estágios: contração e ampliação. No primeiro, os mapas de características são gerados igualmente às CNNs por meio das camadas de convolução e subamostragem. Remove-se a camada completamente conectada, portanto, os mapas gerados servem como entrada para o estágio de ampliação. Nessa fase, novos mapas de características são produzidos através de novas camadas convolucionais e ampliados pelos operadores de *upsampling* (ou *unpooling*). Na última camada, os mapas gerados são submetidos a uma função de ativação que realizará a classificação pixel a pixel, produzindo, desta feita, a máscara de segmentação.

Neste trabalho, é utilizada uma arquitetura FCN chamada U-Net. Proposta por Ronneberger, Fischer e Brox (2015), diferencia-se das FCNs comuns porque combina os mapas de característica da fase de contração com seu correspondente simétrico na etapa de ampliação. Assim, a U-Net possui forma similar à letra "u", como pode ser visto pela Figura 20.

A etapa de contração (descida do "U") contém blocos convolucionais compostos por duas camadas de convolução com filtros de dimensões 3 x 3 e função de ativação ReLU. Cada bloco é sucedido de uma camada de subamostragem (*Max Pooling*) com janelamento 2 x 2. Ressalta-se que, a partir do segundo bloco, são gerados o dobro de mapas de característica do bloco anterior. Após o último bloco de convolução, no nível mais profundo da rede, não haverá camada completamente conectada. Logo, os mapas gerados nesse nível serão a entrada da etapa de ampliação.

A fase de ampliação (subida do "U") contém blocos convolucionais similares aos da fase de contração. Eles são precedidos da operação *upsampling*, a qual amplia o tamanho das saídas produzidas pelo bloco que o antecede. Nessa fase, concatena-se os mapas de características dos blocos simétricos. O último bloco se diferencia dos demais

Figura 20 – Ilustração da arquitetura de uma rede U-Net.



Fonte: Adaptada de (RONNEBERGER; FISCHER; BROX, 2015).

pois é composto por três camadas convolucionais. As duas primeiras, contendo filtros de dimensão 3 x 3 e função de ativação ReLU. A última, entretanto, possui filtros de tamanho 1x1 e função de ativação sigmoide. Essa função atua como um classificador pixel-a-pixel, gerando valores entre 0 e 1 os quais indicam, respectivamente, fundo e região de interesse.

A U-Net têm apresentado resultados promissores em segmentação de imagens médicas (RONNEBERGER; FISCHER; BROX, 2015). No presente trabalho, sua efetividade é verificada na etapa de segmentação do reflexo retiniano.

### 3.7 Máquina de vetores de suporte (SVM)

Esta seção apresenta o método Máquina de Vetores de Suporte que é a técnica de aprendizado de máquina (AM) utilizada nos experimentos deste trabalho na etapa de classificação (Seção 4.2.4).

Máquina de Vetores de Suporte é um método de aprendizagem supervisionada introduzido por Cortes e Vapnik (1995) usado para estimar uma função que classifique dados em duas classes. O conceito básico por trás das SVMs compreende a construção de um hiperplano como superfície de decisão de forma que seja máxima a margem de separação entre as classes. O objetivo do treinamento por meio das SVMs é a obtenção de hiperplanos que dividam a amostra de tal maneira que sejam otimizados os limites de generalização (ALMEIDA et al., 2009).

As SVMs são sistemas de aprendizagem caracterizados pela utilização de um

espaço de hipóteses de funções lineares em um espaço de características de alta dimensionalidade. São treinadas por algoritmos fortemente influenciados pela teoria da otimização e de aprendizagem estatística.

Na metodologia das SVMs, uma variável de predição é chamada de atributo. E este atributo, quando aplicado na construção de hiperplanos, é chamado de característica. O conjunto de características selecionadas para descrever o objeto que se deseja classificar é chamado de vetor. Os vetores que se encontram mais próximos dos hiperplanos construídos para separar as classes são chamados de vetores de suporte.

As SVMs possuem diferentes *kernels* que são utilizados na resolução de problemas de espaços não-lineares. Os mais utilizados são os tipos Linear, Polinomial (que manipula uma função polinomial cujo grau pode ser definido durante os treinamentos), Sigmoidal (permite que a SVM se comporte de maneira similar à rede MLP<sup>1</sup>), e Gaussiano (a SVM se comporta como uma rede RBF<sup>2</sup>). As quatro funções básicas de *kernel* são:

- Linear:  $K(x, y) = x^T y$
- Polinomial:  $K(x, y) = (\gamma x^T y + r)^d, \gamma > 0$
- Sigmoidal:  $K(x, y) = \tanh(\gamma x^T y + r)$
- Função de base Radial (RBF):  $K(x, y) = e^{-\gamma \|x-y\|^2}$

Definições diferentes do *kernel*, bem como dos demais parâmetros, causam alterações nos resultados fornecidos por uma SVM. Nos experimentos deste trabalho, foi utilizado o *kernel* RBF, com parâmetros C (penalidade para classificações incorretas nos dados de treinamento) e  $\gamma$  positivos.

### 3.8 Métricas de Avaliação

A avaliação do desempenho de um método objetiva promover a validação dos resultados obtidos. Esta seção apresenta as métricas utilizadas para quantificar o desempenho do método proposto.

Para avaliar a similaridade entre os mapas de segmentação gerados e as anotações manuais presentes no conjunto de dados, são utilizadas as seguintes métricas: Coeficiente Dice e Índice de Jaccard. O coeficiente de similaridade Dice é o mais utilizado para a

<sup>1</sup> Rede perceptron de múltiplas camadas (*Multi-layer Perceptron*)

<sup>2</sup> Funções de base radial (*Radial Basis Function*)

validação de segmentação em imagens médicas (TAHA; HANBURY, 2015). Ele é definido conforme Equação 3.39, onde X e Y representam as máscaras comparadas para cada imagem de entrada, no caso, a anotação manual e o mapa de segmentação gerado pelo processo de segmentação automática.

$$DICE = \frac{2 * |X \cap Y|}{|X| + |Y|} \quad (3.39)$$

O índice de Jaccard é dado pela razão entre a interseção dos conjuntos X e Y e a união desses (Equação 3.40). Esse índice também é conhecido como Interseção sobre a União (IoU, sigla em inglês).

$$IoU = \frac{|X \cap Y|}{|X \cup Y|} \quad (3.40)$$

A avaliação dos resultados fornecidos pelo classificador para o método proposto consiste em validar a discriminação entre os casos nos quais os olhos estão saudáveis ou possuem alguma patologia. Considera-se, portanto, quatro situações possíveis:

- Verdadeiro Positivo (VP): Há patologia no olho e a classificação foi correta.
- Falso Positivo (FP): Há patologia, mas a classificação foi incorreta.
- Verdadeiro Negativo (VN): Não há patologia no olho e a classificação foi correta.
- Falso Negativo (FN): Não há patologia, mas a classificação foi incorreta.

Neste trabalho, são utilizadas as medidas de Especificidade (E), Sensibilidade (S), Acurácia (A), Precisão (Pr) e o índice *Kappa* por serem métodos estatísticos comumente utilizados para validar resultados em problemas de classificação.

A sensibilidade (Equação 3.41) é a proporção de imagens corretamente classificadas como patológicas. A especificidade (Equação 3.42) indica a probabilidade de as imagens serem classificadas corretamente como saudáveis.

$$S = \frac{VP}{VP + FN} \quad (3.41)$$

$$E = \frac{VN}{VN + FP} \quad (3.42)$$

A acurácia (Equação 3.43) é a proporção total das amostras corretamente classificadas.

$$A = \frac{VP + VN}{VP + FP + FN + VN} \quad (3.43)$$

A precisão (Equação 3.44) relaciona verdadeiros positivos e falsos positivos de acordo com a Equação 3.44.

$$P = \frac{VP}{VP + FP} \quad (3.44)$$

O índice Kappa mede o relacionamento entre concordância e causalidade, bem como o desacordo esperado, indicando o quão legítimas são as interpretações dos resultados (ROSENFELD; FITZPATRICK-LINS, 1986). O valor de *Kappa* é definido como:

$$K = \frac{A - P_c}{1 - P_c} \quad (3.45)$$

onde *A* é a acurácia (Equação 3.43) e  $P_c$  é a probabilidade de concordância devido ao acaso, a qual pode ser expressa por:

$$P_c = \frac{TP + FN}{D} \times \frac{TP + FP}{D} + \frac{TN + FP}{D} \times \frac{TN + FN}{D} \quad (3.46)$$

onde *D* é o número total de exemplos do conjunto de teste. A Tabela 3 mostra como os valores de índice Kappa são categorizados, tal qual estabelecido por Landis e Koch (1977).

Tabela 3 – Precisão da classificação, conforme o índice Kappa.

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

Por fim, o desempenho do método proposto também é avaliado por meio da curva ROC (*Receiver Operating Characteristic*). Esta análise é bastante utilizada em tomadas de decisões médicas, com uso também amplo no campo do aprendizado de máquina (FAWCETT, 2006). A principal informação extraída de uma curva ROC é a área sob a curva (*Area Under Curve* - AUC), cujos valores encontram-se entre 0 e 1. Portanto, quanto mais próximo de 1 (equivalente a 100%), melhor é o desempenho do classificador.

### 3.9 Considerações Finais

Neste capítulo, foram apresentados os conceitos e demais fundamentos teóricos imprescindíveis para a compreensão do método proposto e de sua aplicação. Abordaram-se tópicos como o teste de Brückner e sua potencial utilidade para a detecção de patologias oculares e triagem; técnicas de processamento e análise de imagens digitais; aprendizado profundo utilizando CNN e FCN; reconhecimento de padrões utilizando SVM e métricas de avaliação.

No próximo capítulo, serão explanados os materiais utilizados e o método proposto por este trabalho, cujo objetivo é a detecção da presença de patologias oculares em imagens do Teste de Brückner.

## 4 MATERIAIS E MÉTODOS

Este capítulo descreve os procedimentos realizados para a detecção automática da presença de patologia na visão baseada em imagens do Teste de Brückner. É importante ressaltar que não faz parte desse método o diagnóstico de patologias, limitando-o somente à classificação dos casos em saudáveis ou com patologia.

Primeiro, descreve-se o conjunto de imagens utilizado para validar a proposta. Em seguida, descreve-se cada etapa que constitui esta metodologia. E, por fim, apresenta-se as informações a respeito do hardware e softwares utilizados no desenvolvimento do método proposto pelo presente trabalho.

### 4.1 Base de Imagens

As imagens utilizadas no presente estudo foram adquiridas de pacientes em uma clínica oftalmologia em São Luís-MA. A base de imagens tem atualização constante à medida em que novos pacientes aceitam colaborar com esta pesquisa. Alguns exemplos podem ser vistos na Figura 21. Ressalta-se que esse estudo recebeu a aprovação do comitê de ética para o uso de imagens humanas. Os pacientes que se voluntariaram a participar do experimento assinaram um termo de consentimento.

Figura 21 – Exemplos da base de imagens.



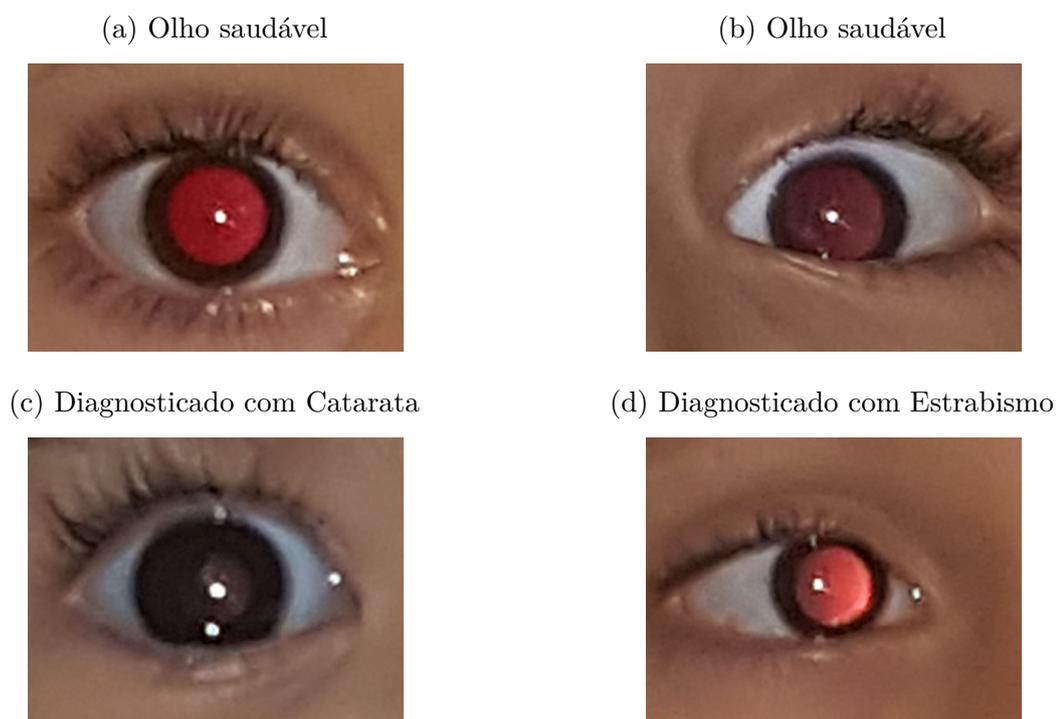
Para a aquisição das imagens, os pacientes devem estar com a pupila dilatada e sem óculos, pois este é o procedimento padrão do Teste de Brückner (JALIS, 2015). As imagens são capturadas por meio de um *smartphone* com a função *flash* ativada, caracterizando um processo não invasivo e análogo ao exame realizado com o oftalmoscópio. Captura-se,

desta feita, um imagem a certa distância do paciente de forma que sua face esteja incluída e o reflexo produzido pela retina seja visualizado.

A base de pacientes utilizada para esse experimento conta com 59 imagens de faces, em RGB, divididas em duas classes: pacientes saudáveis (37 imagens) e pacientes com patologia (22 imagens).

As Figuras 22(a) e (b) apresentam exemplos de olhos saudáveis. O reflexo retiniano em pacientes saudáveis apresenta a coloração avermelhada e visualmente uniforme, mas pode assumir tons mais pálidos em indivíduos afrodescendentes (CAGLIARI et al., 2016). Nos casos com patologia, há uma variação na coloração, pois, cada alteração visual acarreta em tonalidades diferentes de reflexo, como mostrado pelas Figuras 22(c) e (d).

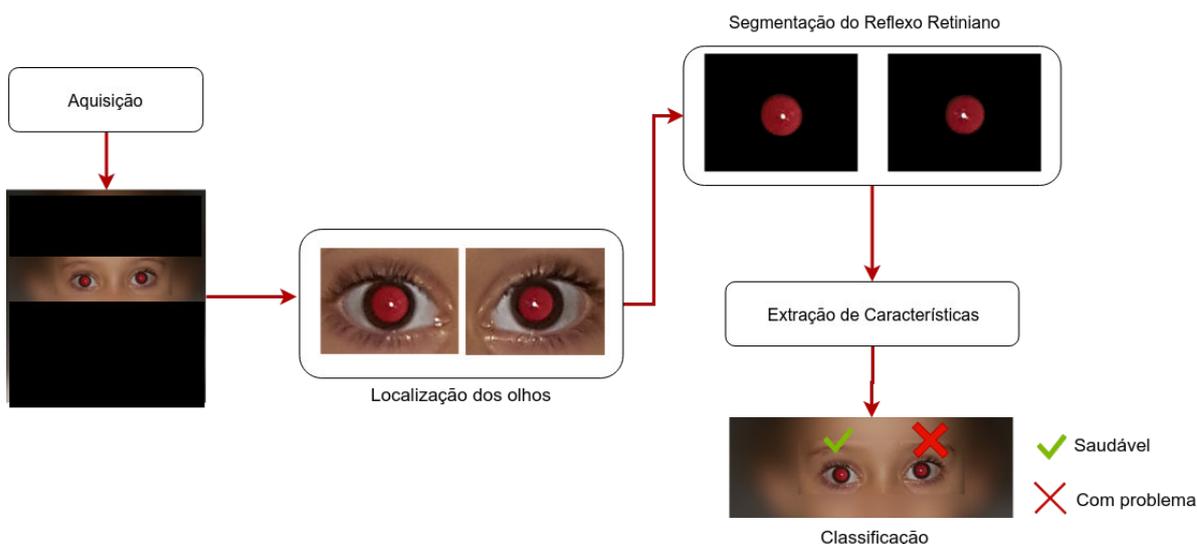
Figura 22 – Exemplos de olhos de pacientes do conjunto de imagens. Diagnóstico realizado pelo especialista.



## 4.2 Método Proposto

O método proposto é composto pelas seguintes etapas: (1) aquisição das imagens, (2) localização dos olhos, (3) segmentação da região do reflexo retiniano, (4) extração de características e (5) classificação. A Figura 23 apresenta o fluxo de execução de cada etapa.

Figura 23 – Etapas da metodologia.



Fonte: Elaborado pelo autor

O método inicia com a captura de uma fotografia da face do paciente. A partir dessa imagem, os olhos serão localizados na etapa de detecção de olhos. Em seguida, ocorre a etapa de segmentação, na qual serão geradas novas imagens contendo somente a região do reflexo retiniano. Dos resultados da segmentação, são extraídas as características as quais são passadas para o classificador. Esse produzirá o resultado final, classificando se o olho é saudável ou tem patologia. As seções seguintes apresentam detalhadamente cada etapa do método proposto.

#### 4.2.1 Localização dos olhos

Partindo-se da captura da imagem, inicia-se o processo de localização dos olhos do paciente. Essa fase é importante, pois, limita o escopo da imagem à região de cada olho separadamente, eliminando, dessa forma, demais detalhes da cena que são indesejados. Esta seção apresenta o processo de localização de olhos que, no método proposto, é realizado com a utilização da rede neural convolucional YOLO (Seção 3.6.2).

Proposta por Redmon et al. (2015), a rede YOLO (*You Only Look Once*) é conhecida pela sua utilização em tarefas de detecção de objetos em tempo real, apresentando resultados bastante satisfatórios em ambientes irrestritos, ou seja, com variações de iluminação ou pose (YANG; JIACHUN, 2018; LAN et al., 2018; KRAWCZYK; STARZYRNSKI, 2018). No caso do ambiente de aplicação do método proposto, a utilização de uma técnica robusta busca garantir que, na imagem do paciente, os olhos serão localizados não obstante

as condições variantes. Essa robustez também permite mais flexibilidade na captura das imagens, podendo o especialista aproximar-se ou distanciar-se do paciente a fim de obter imagens mais adequadas.

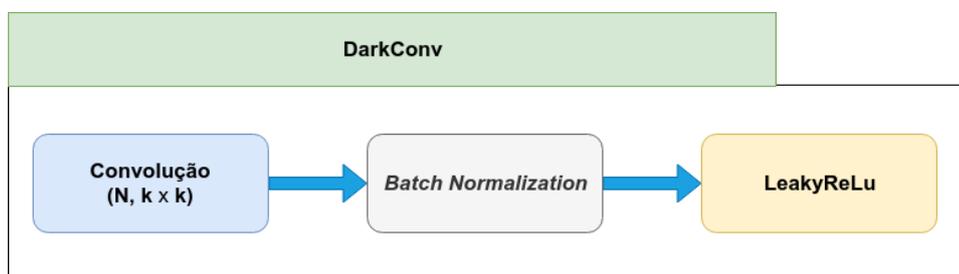
Outro fator que contribuiu para a escolha da YOLO é que esta rede apresenta bom desempenho na detecção de objetos de tamanho relevante numa cena (REDMON et al., 2015). No caso das imagens adquiridas, os olhos ocupam uma parte relevante da face dos pacientes. Parte-se, então, da hipótese de que a localização de olhos por meio dessa rede obtém resultados significativos, dada a área ocupada pelos olhos na cena. Mais detalhes sobre a arquitetura YOLO utilizada e os procedimentos de treino, validação e teste serão discutidos a seguir.

#### 4.2.1.1 Arquitetura da rede YOLO

A rede convolucional utilizada como localizador de olhos no método proposto é a terceira versão da YOLO, proposta por Redmon (2018). Chamada YOLOv3, a arquitetura contém blocos de convolução e blocos residuais. A seguir, será apresentado como os blocos estão organizados e compõem, dessa forma, a YOLOv3.

O primeiro bloco de convolução é chamado de *DarkConv*. Neste bloco, que é ilustrado pela Figura 24, é feita uma convolução em 2D com  $N$  filtros e *kernel* de tamanho  $k \times k$ . Em seguida, há uma camada de *Batch Normalization*, seguida pela função de ativação *LeakyReLU* (Leaky Rectified Linear Unit), apresentada pela Equação 4.1.

Figura 24 – *DarkConv*. Bloco de convoluções utilizado na rede, contendo  $N$  filtros e *kernel* de tamanho  $k$ .

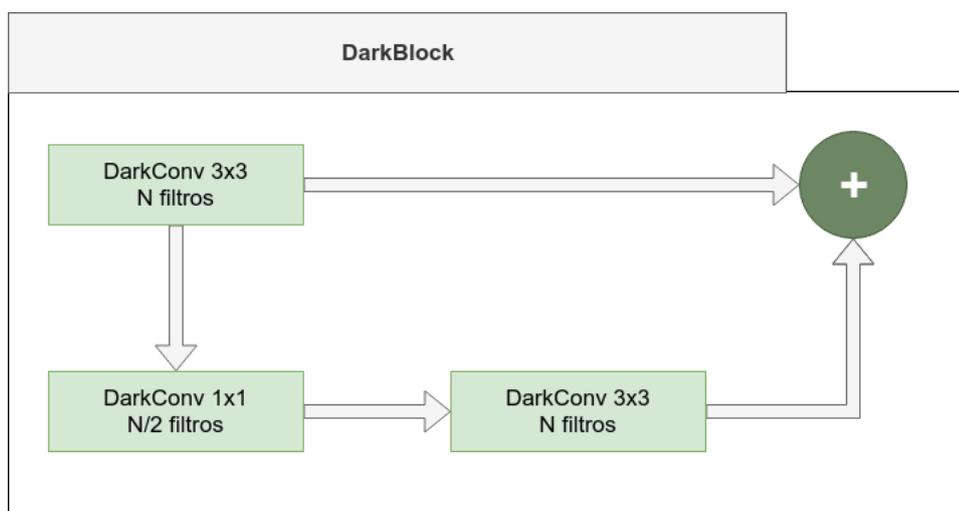


Fonte: Elaborado pelo autor

$$\phi = \begin{cases} x, & \text{se } x > 0 \\ 0.1x, & \text{caso contrário} \end{cases} \quad (4.1)$$

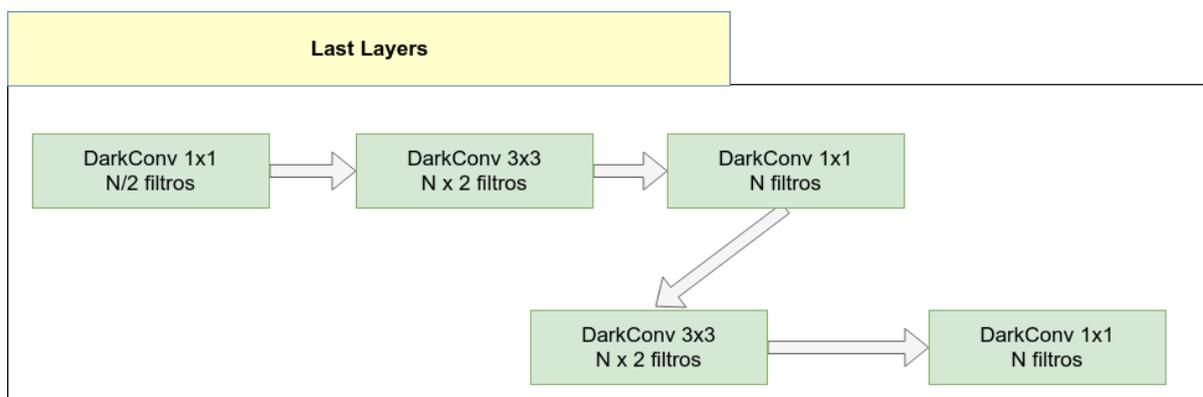
O segundo bloco que compõe a YOLOv3 é chamado de *DarkBlock* (Figura 25). É similar ao bloco residual proposto por Szegedy, Ioffe e Vanhoucke (2016). O bloco recebe como entrada um bloco *DarkConv* de *kernel*  $3 \times 3$  ( $N$  filtros, *stride* 2), cujo resultado é duplicado. Uma cópia serve como entrada para um *DarkConv*, com  $N/2$  filtros e *kernel*  $1 \times 1$ , cuja saída é fornecida para outro *DarkConv*, de  $N$  filtros e *kernel*  $3 \times 3$ . A saída desse último bloco é somada com a outra cópia do resultado do primeiro *DarkConv*.

Figura 25 – *DarkBlock: Darknet Residual Block*. Bloco residual utilizado na rede, contendo  $N$  filtros e *kernel* de tamanho  $k$ .



Fonte: Elaborado pelo autor

Figura 26 – Bloco contendo as últimas camadas da rede, contendo  $N$  filtros e *kernel* de tamanho  $k$ .



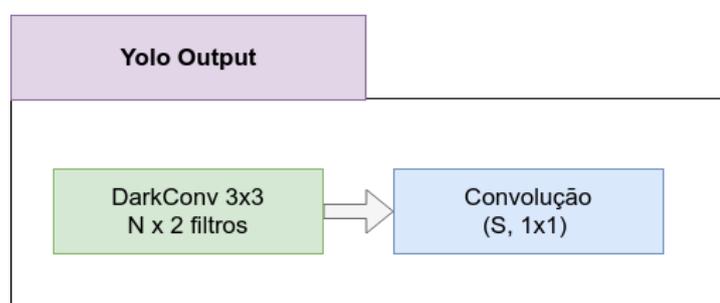
Fonte: Elaborado pelo autor

O terceiro bloco corresponde às camadas finais da rede, predecessoras da camada de saída. Chamado de *LastLayers*, esse bloco é composto por 5 blocos *DarkConv* configu-

rados, alternadamente, contendo  $N$  filtros e kernel  $1 \times 1$  e  $N \times 2$  filtros e kernel  $3 \times 3$ , como mostrado na Figura 26.

O bloco final da YOLOv3 é chamado de *YOLO Output* (Figura 27). Tem como entrada a saída do bloco *LastLayers* e é responsável por prover a saída da rede, consistindo de um bloco *DarkConv* de kernel  $3 \times 3$  e  $N \times 2$  filtros seguido de uma camada convolucional 2D convencional sem função de ativação definida. Essa indefinição procede de forma análoga à função de ativação linear. Destaca-se que essa última camada é responsável pelo processo de regressão inerente da rede. É composta por uma quantidade  $S$  de filtros e kernel  $1 \times 1$ . O número  $S$  é definido como a multiplicação do número de âncoras pela soma da quantidade de classes mais 5. Logo  $S = 9 \times (1 + 5)$ .

Figura 27 – Ilustração da saída da rede utilizada.



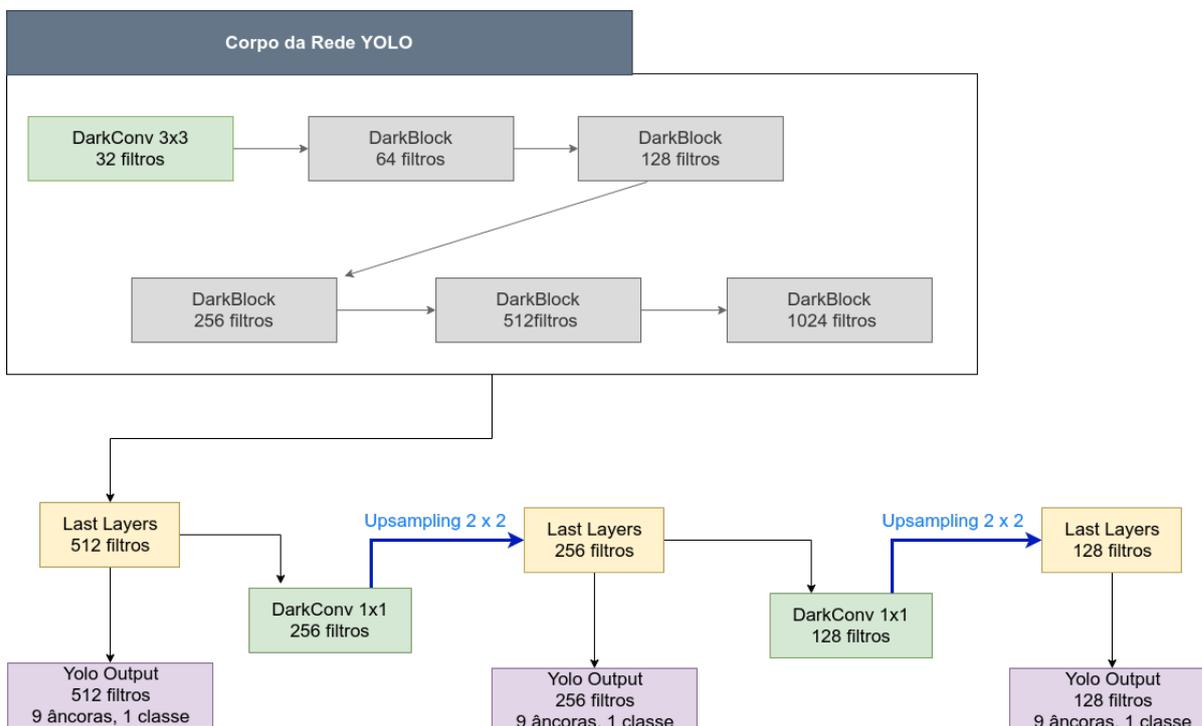
Fonte: Elaborado pelo autor

Após a definição dos blocos que compõem a YOLOv3, é possível compreender a arquitetura da rede de um modo mais completo. A rede, portanto, é composta por 24 blocos de convolução, totalizando 47 camadas convolucionais. Somando-se a primeira saída da rede, totaliza-se 53 camadas. O resultado da rede é predito em 3 escalas diferentes, logo, são fornecidas 3 saídas, ilustradas pela Figura 28.

Na primeira saída, o resultado oriundo da rede passa por um bloco *LastLayers*, cuja saída é a entrada de um bloco *YOLO Output* com 512 filtros. Na segunda saída, um bloco *LastLayers* de 256 filtros recebe como entrada o resultado do bloco *LastLayers* da primeira saída, o qual é processado e passado como entrada para um *YOLO Output* de 256 filtros. Por fim, a terceira saída atua de forma análoga à segunda, entretanto, seus blocos *LastLayers* e *YOLO Output* contêm a metade do número de filtros da saída predecessora.

Definida a arquitetura da rede YOLOv3, o próximo passo consiste no treinamento e teste desta de forma que ela possa ser utilizada, em definitivo, no processo de localização

Figura 28 – Arquitetura YOLOV3 completa utilizada para a tarefa de localização dos olhos.



Fonte: Elaborado pelo autor

dos olhos.

#### 4.2.1.2 Treinamento, Validação e Teste

Esta seção apresenta detalhes do processo de treinamento e teste da YOLOV3 utilizada na etapa de localização da região dos olhos no desenvolvimento do método proposto para a detecção de alterações visuais em imagens de Teste de Brückner.

Para as etapas de treinamento e validação, utilizou-se a base de imagens coletadas para o desenvolvimento do trabalho de Almeida et al. (2012). Os autores cederam esta base, que é composta por imagens de pacientes obtidas em consultório oftalmológico seguindo um protocolo de aquisição similar ao utilizado neste trabalho. Contém um total de 45 pacientes, sendo que cada qual possui 5 ou 6 imagens as quais diferem-se pela direção do olhar do paciente (frente, esquerda, direita, etc). Totaliza-se 243 imagens de rosto, tendo, portanto, 486 olhos. A resolução dessas imagens é de 1572x2048 *pixels*.

Para o treinamento de CNNs, é recomendada a utilização de um grande conjunto de imagens. Assim, a base em questão foi ampliada por meio dos processos de espelhamento e rotação em 90, 180 e 270 graus, totalizando, enfim, 1215 imagens. Dessas, 90% foram

utilizadas para treino e 10% para validação. As regiões de olhos são delimitadas por *bounding boxes* em cada uma das imagens. As posições  $x,y$  de início e  $x+w, y+h$  de término das marcações (sendo  $w$  e  $h$  a largura e comprimento, respectivamente) são escritas em um arquivo de texto cujo caminho é passado como argumento para a execução do *script* Python que contém a implementação da rede.

É importante ressaltar que os conjuntos treino e validação possuem pacientes diferentes de forma que este último conjunto não induza o aprendizado da rede por compartilhar exemplares. O teste é realizado com a base de imagens do Teste de Brückner. As regiões de olhos localizadas pela rede são, portanto, utilizadas como entrada para o próximo estágio do método proposto: a segmentação do reflexo retiniano.

#### 4.2.2 Segmentação do Reflexo Retiniano utilizando U-Net

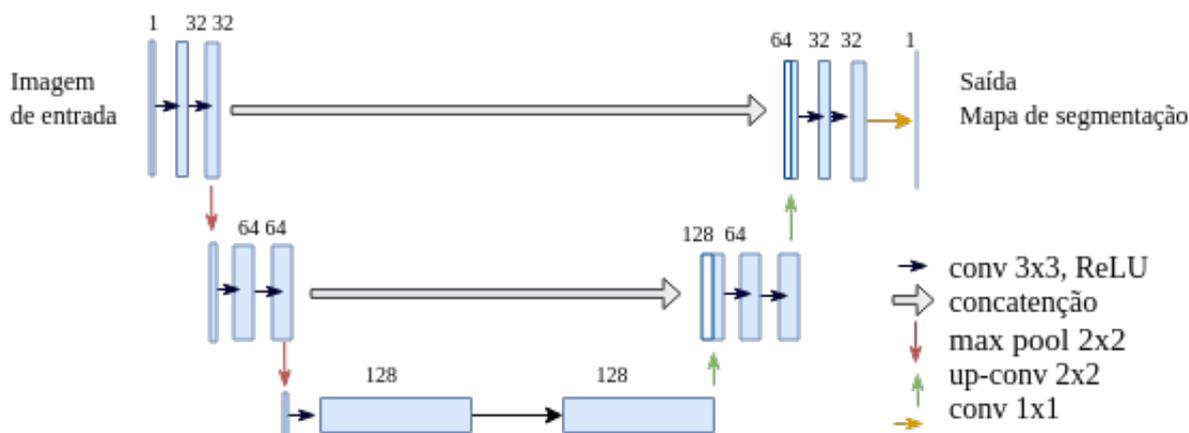
O método proposto utiliza a rede completamente convolucional U-Net (Seção 3.6.3) para a tarefa de segmentação da região do reflexo retiniano nas imagens do exame de Brückner. A rede é treinada a fim de gerar as máscaras das regiões de interesse (ROI) para cada imagem de entrada. Esta seção apresenta mais detalhes sobre a rede convolucional utilizada, bem como a estratégia de segmentação proposta.

Neste trabalho, é utilizada uma arquitetura de rede completamente convolucional (*Fully Convolutional Network* - FCN) chamada U-net. A principal estratégia que diferencia essa arquitetura de outras FCNs é a combinação dos mapas de característica da fase de contração com seu correspondente simétrico na etapa de ampliação. Isso permite a propagação de informações contextuais para os mapas de alta resolução (RONNEBERGER; FISCHER; BROX, 2015). Dessa forma, a U-net pode ser representada com o formato da letra "U", como mostra a Figura 29.

Na Figura, as caixas azuis correspondem aos mapas de características. O número de canais é indicado sobre cada caixa. As caixas brancas representam os mapas de característica concatenados. Setas coloridas indicam as operações realizadas.

A arquitetura proposta tem 14 camadas. Na etapa de contração, cada nível contém uma sequência de duas camadas convolucionais de *kernel*  $3 \times 3$ , que geram a mesma quantidade de mapas de características e aplicam a função de ativação ReLU, seguidas do operador max pooling. No primeiro nível, cada camada convolucional gera 32 mapas. Nos próximos, a quantidade de mapas geradas é o dobro daquela estabelecida

Figura 29 – Arquitetura U-Net utilizada neste trabalho.



Fonte: Elaborada pelo autor.

para o nível anterior. No último nível da fase de contração, cada camada convolucional gera 128 mapas de características.

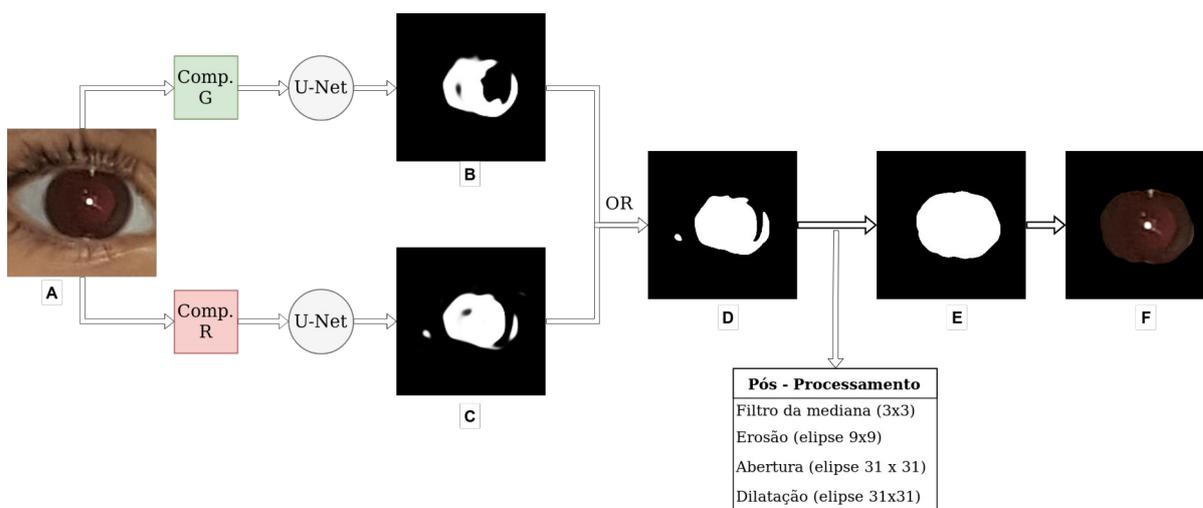
A U-Net não possui camada completamente conectada. Dessa forma, a saída da última camada convolucional da fase de contração é a entrada para a etapa de ampliação. A entrada para cada nível dessa etapa é submetida à operação *up-convolution*, responsável pelo aumento da resolução dos mapas de características. Em cada nível, há duas camadas convolucionais que geram a metade da quantidade de mapas gerados no nível anterior. Nessa etapa, os mapas de característica na entrada de cada nível são combinados com seus correspondentes simétricos da fase de contração. Isso permite que a rede utilize características originais em conjunto com aquelas obtidas após convoluções. O último nível possui três camadas convolucionais. As duas primeiras geram mapas a partir de *kernels* 3 x 3 e ativação ReLU, enquanto que a última utiliza *kernels* 1 x 1 e ativação Sigmoide, cuja saída tende a 1 se aquele pixel faz parte da ROI ou 0, caso contrário.

#### 4.2.2.1 Estratégia de Segmentação

Esta seção apresenta detalhadamente a estratégia de segmentação do reflexo retiniano utilizando a rede U-Net. A Figura 30 mostra os passos executados para a extração da região de interesse a partir das imagens obtidas pela etapa de localização de olhos (seção 4.2.1).

A imagem RGB (Figura 30.A) da região do olho tem suas componente R e G

Figura 30 – Etapas para a segmentação da região do reflexo retiniano.



Fonte: Elaborado pelo autor

extraídas. Cada uma delas é passada como entrada para sua respectiva rede U-Net, as quais produzirão máscaras de segmentação (Figuras 30.B e 30.C). As máscaras resultantes das entradas R e G são combinadas através do operador binário OR. A estratégia de combiná-las surge porque, a partir de análises empíricas, observou-se que o mapas de segmentação gerados para as componentes se complementam (Figura 30.D). Em seguida, a nova máscara binária é submetida a uma etapa de pós-processamento a fim de uniformizar sua principal região em detrimento de possíveis ruídos (Figura 30.E). Esta etapa é caracterizada pela aplicação das seguintes operações:

1. Filtro da mediana, *kernel* 3x3
2. Erosão, com elemento estruturante 9x9
3. Abertura morfológica, com elemento estruturante 31 x 31
4. Dilatação, com elemento estruturante 31x31

Os elementos estruturantes citados são do tipo elíptico e tiveram suas dimensões definidas após vários testes. Depois desse processo, uma máscara definitiva é obtida e enfim aplicada sobre a imagem de entrada, segmentando, dessa forma, a região de interesse (Figura 30.F).

#### 4.2.2.2 Treinamento da U-Net

A partir da base de imagens principal, com 59 imagens (Seção 4.1), foi criado um novo conjunto contendo somente as regiões de olhos recortadas manualmente, totalizando 118 imagens. Em cada imagem desse novo conjunto, foram aplicadas operações de translação em 4 direções diferentes e aleatórias, tomando-se cuidado para não ultrapassar os limites da imagem de modo que se perdesse a visibilidade da região do reflexo. Assim, o conjunto tornou-se mais variado e próximo da situação real, em que o método localizador de olhos não garante que eles estejam sempre centralizados na imagem da região detectada. Esse novo conjunto de imagens ficou com 472 exemplares. E, a partir dele, foram criados mais dois, de mesma quantidade, um contendo somente imagens da componente R e o outro, da componente G.

Duas redes U-Net com arquitetura e configurações de treinamento iguais foram treinadas, uma para o grupo de imagens R e a outra para aquele com exemplos da componente G. Esse experimento também foi realizado com imagens da componente B, mas as máscaras geradas assemelhavam-se às produzidas para o canal G. Da mesma forma, foram realizados experimentos com as imagens originais em RGB, no entanto, os resultados para a segmentação não foram satisfatórios. Assim, com base nos testes executados, optou-se por manter a estratégia de segmentação somente nos canais R e G.

#### 4.2.3 Extração de características

O processo de extração de características das imagens do reflexo retiniano para a detecção de alterações visuais é discorrido nesta seção. Para atingir os objetivos do método proposto, foram realizados estudos e análises do reflexo baseando-se em dois tipos de características: textura e cor.

Nessas abordagens, a imagem precisa ser representada de outras formas, de modo que possam ser realizadas diferentes análises dos dados a fim de detectar os padrões característicos para cada classe. Neste estudo, a imagem é representada por matrizes de coocorrência para a análise das características de textura. E para a extração de características de textura, representa-se a imagem por meio de histogramas.

O pré-processamento ao qual as regiões do reflexo são submetidas é a conversão para outros espaços de cores além do RGB. Dessa forma, busca-se padrões latentes à imagem que podem ser identificados pela descrição das características fornecidas por

espaços de cores diferentes.

As subseções a seguir explicam a respeito das técnicas de extração de características utilizadas pelo presente estudo.

#### 4.2.3.1 Extração de Características de Textura

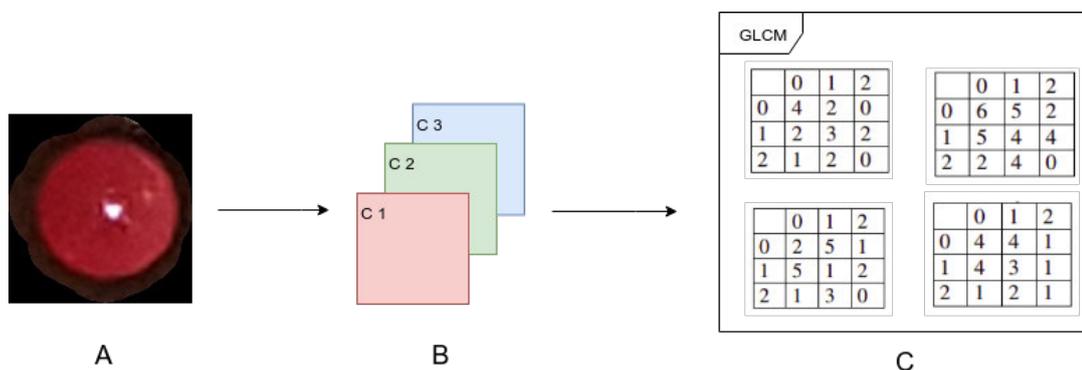
Neste primeiro momento, far-se-á uma explicação sobre o processo de extração de característica de textura das regiões do reflexo retiniano oriundas do processo de segmentação automática. Para isso, representa-se as imagens por meio das matrizes de coocorrência (seção 3.4.1), técnica pela qual analisa-se a frequência de aparições de uma intensidade de pixel dada a ocorrência de outra em uma certa distância e determinada direção. Neste trabalho, foram consideradas duas abordagens de representação: matrizes de coocorrência em nível de cinza (GLCM) e em nível de cores (CLCM). A principal diferença entre essas abordagens é o método para o cálculo das frequências, visto que GLCMs são calculadas individualmente para cada componente no espaço de cores; e as CLCMs são geradas baseadas na relação entre as componentes.

A seguir, descreve-se com mais detalhes os processos de extração de características de textura.

##### 4.2.3.1.1 GLCM

Para cada componente do espaço de cores para o qual a região de interesse é convertida, são geradas quatro matrizes de coocorrência. Esse processo pode ser melhor compreendido com a visualização da Figura 31.

Figura 31 – Método GLCM. A imagem do reflexo (A) é decomposta em canais (B) e, para cada canal se obtêm as matrizes (C).



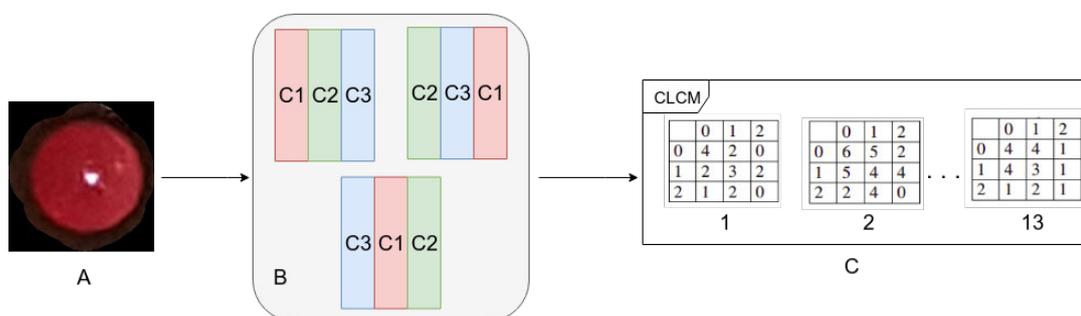
Fonte: Elaborado pelo autor

O cálculo das matrizes leva em consideração dois parâmetros principais: distância e direção. A configuração deles abre espaço para criar novas representações de uma mesma imagem, e, assim, diferentes análises. Neste experimento, considera-se a distância  $d=1$  e quatro direções ( $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ ,  $135^\circ$ ). Dessa forma, uma imagem é capaz de gerar 12 matrizes de coocorrência (3 componentes \* 4 GLCMs).

#### 4.2.3.1.2 CLCM

O cálculo das matrizes CLCM leva em consideração a relação entre os *pixels* das três componentes da imagem. O processo pode ser resumido como visto na Figura 32.

Figura 32 – Método CLCM. A imagem do reflexo (A) é decomposta em canais (B) e, para cada combinação de canais se obtêm as 13 matrizes (C).



Fonte: Elaborado pelo autor

Os parâmetros a serem configurados são a distância e a componente de referência para o cálculo das coocorrências em 13 direções como visto na Figura 32. Neste experimento, é utilizada a distância  $d=1$ , e três combinações de componentes, isto é, alterações na componente de referência. Desta feita, uma imagem gera 39 matrizes CLCM (3 combinações de componentes \* 13 CLCMs).

#### 4.2.3.1.3 Descrição e vetores de características

Após o cálculo das matrizes de coocorrência (GLCMs e CLCMs), é preciso representá-las por meio de vetores de características. Essa representação matemática é necessária para o funcionamento dos algoritmos de aprendizado de máquina utilizados em etapas posteriores a esta. Neste experimento, calculam-se os 13 descritores de Haralick (HARALICK; SHANMUGAM et al., 1973) sobre as matrizes de coocorrência geradas pelos métodos GLCM e CLCM.

No método GLCM, uma imagem gera 12 matrizes. Cada uma delas é descrita por 13 valores correspondentes às métricas de Haralick, os quais são concatenados, totalizando, portanto, uma instância com 156 ( $12 * 13$ ) características.

Para o método CLCM, uma imagem produz 39 matrizes, também descritas por 13 métricas. A concatenação desses valores calculados para cada matriz formam uma única instância de 507 ( $39 * 13$ ) características.

#### 4.2.3.2 Extração de características de cor

Esta seção aborda o processo de extração de características de cores das imagens resultantes do processo de segmentação automática, as quais contém a região do reflexo retiniano a ser analisada. As técnicas utilizadas consistem na representação as imagens por meio de histogramas, mecanismos pelos quais se permite a análise das frequências das intensidades de *pixels* em uma imagem. Neste trabalho, utiliza-se as técnicas histograma de cores global e local.

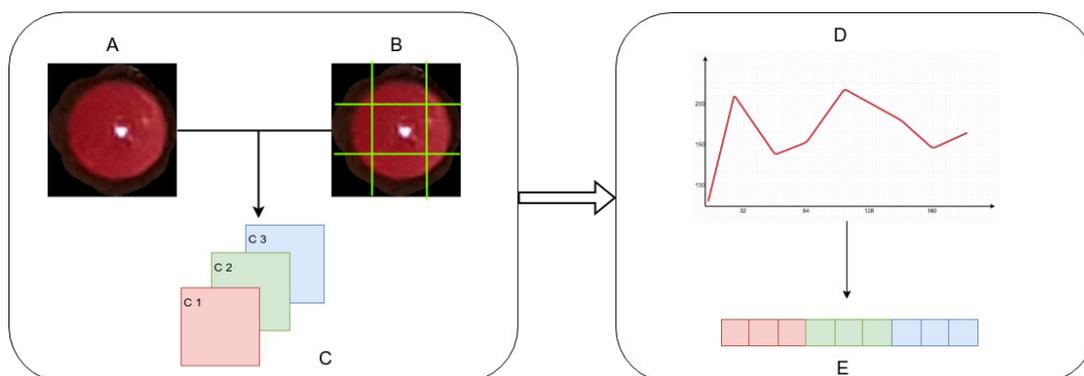
##### 4.2.3.2.1 Histogramas de cores global e local

Os histogramas globais consistem no cálculo das frequências de intensidades para os *pixels* de cada canal da imagem. Dada uma imagem em um certo espaço de cores, calcula-se um histograma individual para cada componente, cujo processo é representado pelo fluxo seguido na Figura 33A . Ao fim, a concatenação desses histogramas representará as características da imagem. É importante ressaltar que cada histograma contém 256 valores, correspondentes às intensidades de 0 a 255 que cada *pixel* pode assumir. Logo, cada imagem, devido a concatenação, é representada por 768 características.

O processo de representação por histogramas locais consiste, primeiramente, na divisão da imagem em uma quantidade  $n$  de regiões. Para cada região serão extraídos histogramas de suas componentes, de forma análoga ao método global. Busca-se, com uma análise local, identificar mais detalhes das imagens que podem ser, de certa forma, desconsiderados com a análise global. Neste trabalho, utiliza-se o método de grade para a divisão de regiões. A Figura 33B também apresenta o fluxo de execução dessa etapa.

Para os experimentos, as imagens foram divididas em grades 3x3 e 5x5. Para cada região, são extraídos histogramas de cada componente (Figura 33D). Por fim, todos os histogramas são concatenados (Figura 33E). Imagens divididas em grades 3x3 geram

Figura 33 – Calculando histogramas globais e locais.



Fonte: Elaborado pelo autor

6912 ( $3 \times 3 \times 3 \times 256$ ) características; enquanto que, com a divisão  $5 \times 5$ , são gerados 19200 ( $5 \times 5 \times 3 \times 256$ ) características.

#### 4.2.3.2.2 Extração de métricas do histograma

Uma outra abordagem que envolve a extração de características de cor é o cálculo de algumas métricas com base nos dados fornecidos pelo histograma. Algumas dessas métricas também são calculadas sobre as matrizes de coocorrência explanadas na Seção 4.2.3.1, entretanto, nesta etapa, levando em consideração a unidimensionalidade dos histogramas de cores, tanto em nível global quanto local.

As métricas calculadas são a entropia, variância, amplitude, mediana, média, moda, coeficiente de variação e o quadrado médio de diferenças sucessivas. Para mais detalhes sobre essas medidas, recomenda-se verificar a Seção 3.4.3 do Capítulo 3.

Cada histograma passa, portanto, a ser representado pelas 8 métricas supracitadas. Dessa forma, no caso dos histogramas globais, as imagens são representadas por 24 características (3 componentes \* 8 métricas). Para os histogramas locais, obtêm-se instâncias com 216 e 600 características, respectivamente para as divisões em grades  $3 \times 3$  e  $5 \times 5$ .

#### 4.2.4 Treinamento e Classificação utilizando SVM

A etapa final consiste na classificação do reflexo retiniano segmentado nas categorias saudável ou com patologia utilizando as técnicas de reconhecimento de padrões

descritas na Seção 3.7, de acordo com as propostas de extração de características de cor e textura (Seção 4.2.3).

Nesta etapa, utiliza-se a biblioteca LIBSVM (CHANG; LIN, 2011) em conjunto com o software WEKA (HALL et al., 2009), o qual auxilia os usuários no uso do SVM de forma mais facilitada. Esse classificador é utilizado com o núcleo de base radial (RBF). Ao utilizar esse núcleo, requer-se a configuração de dois parâmetros:  $C$  e  $\gamma$ . Não se sabe *a priori* quais  $C$  e  $\gamma$  são melhores para o problema. Desta feita, deve-se identifica-los para que o classificador possa prever dados desconhecidos com precisão. Para determinar os valores destes parâmetros, utiliza-se a técnica de *grid search* existente na biblioteca LIBSVM, que consiste em uma busca exaustiva no espaço de parâmetros. Esta busca é realizada sobre os dados de treinamento.

Entretanto, antes de realizar o procedimento de classificação, as características passam por um processo de normalização para uma faixa de valores comuns entre 0 e 1. Este mecanismo, definido pela Equação 4.2, padroniza a distribuição de valores das variáveis, as quais podem assumir diferentes domínios. E também facilita a convergência do classificador na etapa de treinamento.

$$Norm = \frac{num - min}{max - min} \quad (4.2)$$

Sabe-se que é comum a separação dos dados em conjuntos de treinamento e teste, esta última desconhecida para o classificador, de forma que a sua predição reflita o desempenho da classificação de outro conjunto de dados desconhecido. Neste trabalho, utiliza-se o método de validação cruzada *K-fold* para a separação desses conjuntos.

No método *K-fold*, primeiro divide-se o conjunto de treinamento em  $k$  subconjuntos de tamanhos iguais. Em seguida, treina-se o classificador com  $k - 1$  subconjuntos e então testa-se o subconjunto remanescente. Dessa forma, cada exemplo do conjunto de treinamento inteiro é predito uma vez que a precisão da validação cruzada é a porcentagem de dados que são classificados corretamente (CHANG; LIN, 2011). No presente trabalho, utiliza-se  $k = 10$ , logo, os dados de treinamento são organizados em 10 grupos. Cada experimento foi repetido 5 vezes com as instâncias organizadas de maneira aleatória.

Para mensurar o desempenho do método proposto e validar seus resultados foram utilizadas as técnicas de análise de sensibilidade, especificidade, acurácia, índice Kappa e área sob a curva ROC, apresentados na Seção 3.8. Essas são as métricas de desempenho

tipicamente aplicadas a análise de imagens médicas (BLAND, 2010). A classificação é encerrada com o treinamento e teste do SVM gerando a categorização das características extraídas do reflexo em saudável ou patológico.

### 4.3 Hardware e Softwares Utilizados

Os experimentos contaram com o suporte de ferramentas responsáveis por facilitar a aplicação dos métodos dentro da proposta deste trabalho. Estas ferramentas consistem em bibliotecas e aplicações implementadas nas linguagens de programação Java<sup>1</sup> e Python<sup>2</sup>. Essas linguagens orientadas a objetos são conhecidas por serem gratuitas e por oferecerem muitos recursos, entre eles, portabilidade, facilidade de programação, confiabilidade e documentação disponível.

As redes convolucionais utilizadas foram implementadas em Python com o uso da biblioteca de aprendizado profundo Keras(CHOLLET et al., 2015). Para o treinamento das redes convolucionais, tanto para a etapa de detecção de olhos, quanto para a segmentação do reflexo retiniano, foi utilizada a GPU NVIDIA 1060Ti.

Os experimentos de classificação foram realizados no software WEKA (versão 3.9) (HALL et al., 2009). Esta ferramenta, codificada em linguagem Java, também foi utilizada para a visualização dos resultados da classificação com o SVM. O classificador SVM foi obtido por meio da biblioteca LibSVM (versão 3.22) (CHANG; LIN, 2011). Todas essas bibliotecas estão disponíveis gratuitamente na internet.

O computador utilizado para a implementação e testes da metodologia proposta conta com a seguinte configuração: processador Intel Core i7-7700, 16GB de memória RAM e 1TB de HD.

### 4.4 Considerações Finais

Este capítulo apresentou o método desenvolvido para a detecção da presença de patologia na visão baseada em imagens do Teste de Brückner. Apresentou-se a base de imagens coletada para este fim. Explanou-se também sobre cada etapa do método desenvolvido, iniciando pela localização dos olhos dos pacientes, a qual utiliza a rede neural convolucional YOLOv3 para encontrar essas regiões de interesse nas imagens. Depois,

---

<sup>1</sup> <https://www.java.com>

<sup>2</sup> <https://www.python.org>

a segmentação do reflexo retiniano utilizando U-Net, etapa importante para o recorte dessa região que é o objeto de análise do Teste de Brückner. Seguindo para a extração de características, em que são utilizadas duas abordagens (textura e cor) para representá-las. Finalizando o método, discorreu-se sobre a categorização dos dados representados através do classificador SVM. Por fim, o capítulo apresenta hardware e softwares utilizados.

O capítulo subsequente trata dos resultados alcançados e sua discussão, trazendo também mais detalhes dos experimentos.

## 5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os resultados obtidos pelo método proposto para a detecção da presença de patologia na visão em imagens do teste de Brückner. São discutidos os resultados para as etapas de localização dos olhos, segmentação do reflexo retiniano e classificação do reflexo utilizando as características de textura e cor. Os experimentos foram realizados com a base de imagens coletada como explicitado na seção 4.1.

### 5.1 Localização dos Olhos

A etapa de localização de olhos conta com a utilização da rede YOLOv3. Para o treinamento dessa arquitetura, utilizou-se os pesos pré-treinados com a base *ImageNet* como uma forma de acelerar o processo de convergência da rede, e também utilizar este aprendizado *a priori* como sendo a classe "não-olho". Desta feita, foram experimentadas duas abordagens de treinamento para a rede YOLOv3. Em ambas, o treinamento foi realizado em 100 épocas. A primeira delas consiste em manter fixos os pesos de todas as camadas entre o primeiro *DarkBlock* e o bloco *YOLO Output* durante as primeiras 50 épocas, sendo "descongelados" no restante de treinamento. Na segunda abordagem, entretanto, não houve congelamento de pesos.

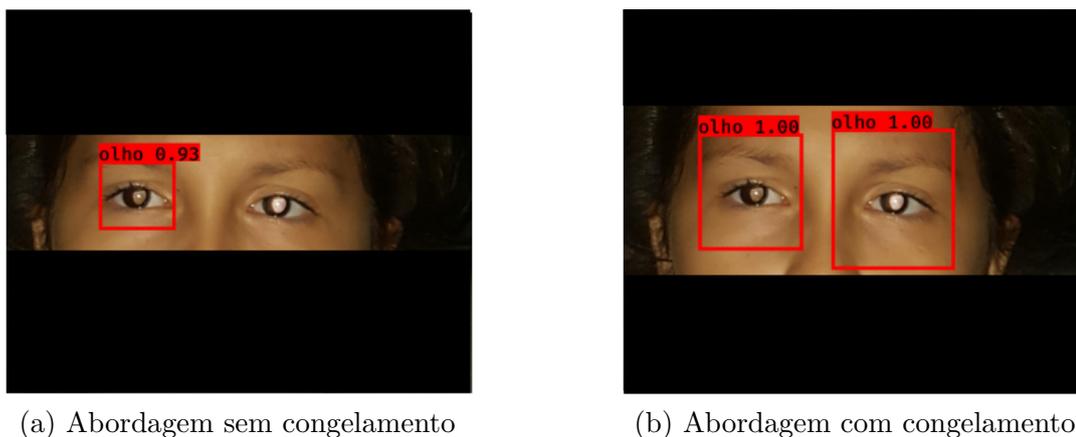
Como dito na seção 4.2.1, a base para esse experimento contém 1235 imagens. Dessas, 90% são utilizadas para treino e 10% para validação. Os testes foram realizados com a base de imagens do teste de Brückner, que possui 59 imagens de pacientes. Os resultados para a fase de teste podem ser visualizados na Tabela 4.

Tabela 4 – Resultado da etapa de localização de olhos utilizando YOLOv3.

Abordagem	VP	FP	Precisão
Com congelamento	118/118	5	95,93%
Sem congelamento	113/118	5	95,76%

VP e FP correspondem, respectivamente aos verdadeiros e falsos positivos. Em outras palavras, os verdadeiros positivos são as regiões de olho decerto localizadas. Os falso positivos são regiões classificadas incorretamente como "olho". Percebe-se pela tabela que os métodos apresentam resultados muito próximos de precisão, visto que geraram o mesmo número de falsos positivos. No entanto, a abordagem com congelamento conseguiu localizar

Figura 34 – Resultados do método localizador de olhos utilizando a rede YOLOv3.



os olhos corretamente em 100% das imagens. Enquanto que o método sem congelamento apresentou taxa de 95,76%. A Figura 34 apresenta os resultados da localização gerados pelas abordagens para uma mesma imagem.

Analisando a Figura 35, percebe-se que os falso positivos encontrados apresentam características semelhantes às do olho. Em ambas, o objeto circular está centralizado em uma região de cor branca, de forma similar à relação entre íris e esclera.

Figura 35 – Exemplos de falsos positivos gerados pelo localizador de olhos.



Pela análise das métricas, verifica-se que a abordagem de treinamento com congelamento de pesos produziu os melhores resultados. Os olhos detectados são, portanto, passados para a próxima etapa do método.

## 5.2 Segmentação do Reflexo Retiniano

Após a localização dos olhos, o método parte para a etapa de segmentação do reflexo retiniano utilizando a rede U-Net.

Para o treinamento da U-Net, as bases de imagens foram divididas nos conjuntos treino, validação e teste como segue: 80% para treino, 20% para teste, 20% do conjunto

de treino é usado para validação. Os resultados obtidos para as métricas de avaliação de segmentação utilizadas podem ser visualizados na Tabela 5.

Tabela 5 – Resultados obtidos pelas redes U-Net para a segmentação do reflexo retiniano.

	U-Net G	U-Net R
Coefficiente Dice (%)	87,45	87,60
Índice de Jaccard (%)	77,74	77,95

As medidas de similaridades são obtidas comparando-se as máscaras geradas com aquelas anotadas manualmente. Em ambas as métricas, o melhor resultado é aquele mais próximo de 100%. A partir desses resultados, é possível inferir que as máscaras geradas contém, no geral, uma grande quantidade de *pixels* que se sobrepõem à anotação manual.

Também foram realizados experimentos com outras duas abordagens. A primeira é proposta em Kumar et al. (2009b) e segmenta o reflexo baseando-se na localização da mancha brilhosa. Os testes utilizando esse método alcançaram coeficiente Dice de 62,46% e 53,45% para o índice de Jaccard. O outro método experimentado baseia-se em grafos e *superpixels* e faz uso do algoritmo *Spectral Clustering* para segmentação. Os testes realizados com esse método obtiveram coeficiente Dice de 69% e 54,08% para o índice de Jaccard. A Figura 36 apresenta um comparativo entre as segmentações geradas pelo método proposto neste trabalho e as demais abordagens experimentadas.

A partir dessa comparação, observa-se que elas corroboram com os resultados apresentados na Tabela 5. A estratégia que combina as máscaras geradas pela U-Net G e R são as que mais se aproximam das anotações manuais (*ground-truth*).

### 5.3 Classificação

O experimento conta com 118 imagens segmentadas do reflexo retiniano, sendo 88 para a classe saudável e 30 para a classe problema (com patologia). As imagens são representadas pelas características de textura e cores. Esta seção apresenta os resultados obtidos para cada abordagem de extração de características.

Os testes foram realizados com o classificador SVM configurado com o *kernel* RBF. Os parâmetros  $C$  e  $\gamma$  foram estimados por meio do método *Grid Search*. Os experimentos foram realizados seguindo o método *k-fold*, com  $k = 10$  *folds*, sendo executados por cinco vezes. Para cada execução, extraíram-se as métricas de avaliação acurácia (A), desvio padrão da acurácia (Desv), sensibilidade (S), especificidade (E), índice Kappa e a área

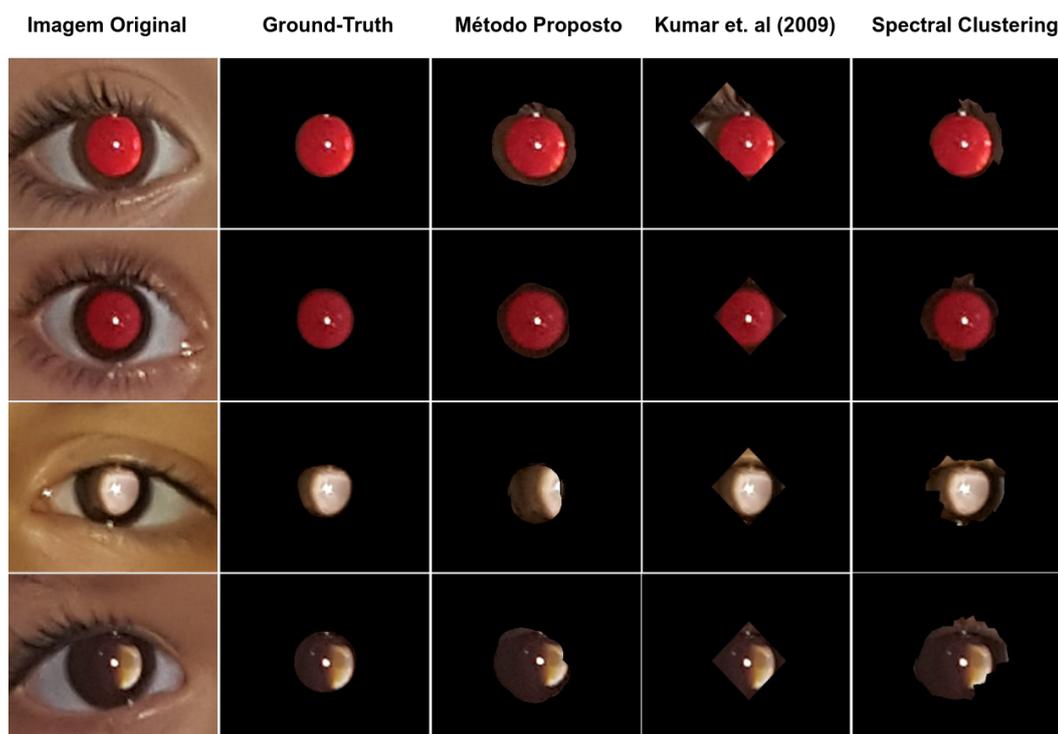


Figura 36 – Comparativo entre o método de segmentação proposto e outras duas abordagens.

sob a curva ROC (Seção 3.8). Os resultados apresentados a seguir, portanto, são a média dos valores obtidos para cada métrica.

### 5.3.1 Avaliação das características de textura

As características de textura das imagens são representadas pela análise estatística das matrizes de coocorrência dos tipos CLCM e GLCM (Seção 4.2.3.1). Os resultados obtidos pelo classificador SVM para essas abordagens são apresentados pelas Tabelas 6 e 7.

Tabela 6 – Resultados da classificação para a abordagem CLCM.

Espaço de cores	A (%)	Desv.	E (%)	S (%)	Kappa	ROC
HLS	88,64	1,75	94,24	69,32	0,68	0,82
HSV	89,66	1,10	95,46	72,64	0,71	0,84
LAB	89,15	1,73	96,38	68,00	0,69	0,82
LUV	88,30	1,83	95,94	65,98	0,66	0,80
<b>RGB</b>	<b>90,33</b>	<b>0,46</b>	<b>97,04</b>	<b>70,68</b>	<b>0,72</b>	<b>0,83</b>
XYZ	88,64	1,28	94,78	70,66	0,68	0,82
<b>YCrCb</b>	90,00	1,10	95,24	74,68	0,72	0,84

Para a abordagem CLCM, o classificador alcança 90,33% de acurácia nas imagens

Tabela 7 – Resultados da classificação para a abordagem GLCM.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	89,83	1,98	95,02	74,68	0,72	0,84
HSV	90,67	2,47	95,68	76,00	0,74	0,85
LAB	90,50	1,51	96,12	74,00	0,73	0,85
<b>LUV</b>	<b>91,86</b>	<b>1,53</b>	<b>97,94</b>	<b>74,02</b>	<b>0,76</b>	<b>0,86</b>
RGB	90,33	1,28	95,70	74,66	0,73	0,85
<b>XYZ</b>	91,31	0,81	96,30	<b>76,65</b>	0,76	0,86
YCrCb	91,10	0,48	96,87	74,17	0,75	0,85

em RGB, apresentando um desvio padrão baixo de 0,46, indicando uma baixa variação da acurácia durante as execuções do experimento. Essa abordagem alcança um valor de 70,68% de sensibilidade, que é a taxa de acerto para a classe problema. Já os resultados obtidos com as matrizes GLCM superiores, visto que o SVM alcançou acurácia de 91,96% e uma sensibilidade de 74,02% no espaço de cores LUV.

Os resultados de especificidade obtidos para ambas as abordagens têm valores pouco acima de 97%, indicando uma alto poder discriminatório para a classe saudável. Isso é compreensível, visto que a maior quantidade de exemplares saudáveis no conjunto de imagens pertence a essa classe. Na Tabela 7, destaca-se a mais alta sensibilidade para o teste realizado com as imagens no espaço XYZ. Esse resultado, apesar de não ser o maior em acurácia, também se apresenta como promissor, visto que os valores de especificidade também são elevados, e o desvio padrão da acurácia mantém-se baixo.

A partir dos resultados expostos nas tabelas, percebe-se que o método GLCM é o mais indicado, em comparação com o CLCM. Ressalta-se, ainda, que ambos produzem índice Kappa superior a 0,70, encaixando-se na categoria "muito bom".

### 5.3.2 Avaliação das características de cores

Os métodos de análise de cores se baseiam nas informações fornecidas pelos histogramas das imagens. As abordagens utilizadas são o histograma de cores em nível global (HCG) e local (HCL), e as variantes MHCG e MHCL em que são extraídas métricas desses histogramas (Seção 4.2.3.2).

Os resultados obtidos pelo SVM para a abordagem HCG podem ser vistos pela Tabela 8. O melhor resultado para a técnica HCG foi obtido com as imagens no espaço de cores LAB, com acurácia de 90,5% e desvio padrão de 1,39, o qual indica uma variação pequena para aquela métrica.

Tabela 8 – Resultados da classificação para a abordagem HCG.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	89,15	0,70	92,04	80,66	0,71	0,86
HSV	89,49	1,53	95,46	72,00	0,70	0,83
<b>LAB</b>	<b>90,50</b>	<b>1,39</b>	<b>92,26</b>	<b>85,36</b>	<b>0,75</b>	<b>0,88</b>
LUV	87,79	1,75	91,16	78,00	0,68	0,84
RGB	88,81	1,51	93,16	76,00	0,70	0,84
XYZ	88,47	2,20	92,28	77,32	0,69	0,84
YCrCb	88,81	1,51	90,90	82,68	0,71	0,86

Em comparação com os métodos de textura, observa-se aqui o aumento na sensibilidade, indicando que esse método fornece características mais discriminatórias para a classe problema. Não obstante, a especificidade apresenta uma drástica diminuição, indicando um aumento de falsos negativos para a classe saudável. Em termos de triagem, isso não representa uma gravidade, pois entende-se que é melhor que um paciente saudável seja encaminhado para uma consulta com especialista que classificar como saudável um paciente com problema.

A classificação com características obtidas pela extração de métricas do histograma global gerou os resultados apresentados pela Tabela 9. Com essa abordagem, observa-se o aumento na acurácia e na especificidade, sendo que o melhor resultado foi alcançado com o espaço de cores HSV.

Tabela 9 – Resultados da classificação para a abordagem MHCG.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	89,15	1,25	95,94	69,34	0,69	0,82
<b>HSV</b>	<b>92,20</b>	<b>1,39</b>	<b>96,82</b>	<b>78,68</b>	<b>0,78</b>	<b>0,87</b>
LAB	86,77	1,41	94,32	64,68	0,62	0,79
LUV	85,08	0,96	92,26	63,98	0,58	0,78
RGB	89,32	0,75	93,40	77,34	0,71	0,85
XYZ	86,61	2,19	92,46	69,34	0,63	0,80
YCrCb	87,79	1,53	92,92	72,66	0,67	0,82

A acurácia aumenta em decorrência da elevação na especificidade. No entanto, há uma diminuição na sensibilidade, indicando que, com essa abordagem, o classificador se torna menos apto para detectar olhos com patologia.

Os experimentos com as características descritas por histogramas locais (HCL e MHCL) possuem duas vertentes. Uma, com os histogramas obtidos pelas imagens divididas em *grid* 3x3 e a outra, com o *grid* 5x5.

A Tabela 10 mostra os resultados para o HCL com *grid* 3x3. Para essa abordagem, o classificador SVM obteve o melhor desempenho geral com imagens no espaço de cores XYZ.

Tabela 10 – Resultados da classificação para a abordagem HCL *grid* 3x3.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	88,81	1,25	89,98	85,34	0,71	0,87
HSV	90,00	0,37	92,24	83,30	0,74	0,87
LAB	89,32	1,13	90,64	85,34	0,73	0,88
LUV	88,13	0,84	89,32	84,66	0,70	0,87
RGB	91,01	1,28	92,94	85,34	0,76	0,89
<b>XYZ</b>	<b>91,10</b>	<b>1,46</b>	<b>92,90</b>	<b>85,82</b>	<b>0,77</b>	<b>0,89</b>
<b>YCrCb</b>	88,55	1,09	88,07	<b>90,00</b>	0,72	0,89

Com o HCL 3x3, houve ganho de sensibilidade, em relação ao histograma global, tratando-se portanto de uma proposta promissora, visto a importância da identificação da classe problema no contexto de aplicação do exame de Brückner. Destaca-se que, apesar de alcançar uma acurácia menor, o experimento com imagens em YCrCb obteve sensibilidade 90%.

Também foram feitos testes com as características obtidas via análise estatística do HCL 3x3. Os resultados para essa abordagem estão expostos na Tabela 11.

Tabela 11 – Resultados da classificação para a abordagem MHCL *grid* 3x3.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	92,71	0,75	97,48	78,68	0,79	0,88
HSV	90,84	1,83	96,82	73,34	0,74	0,85
LAB	91,18	2,44	95,94	77,32	0,75	0,86
<b>LUV</b>	<b>92,88</b>	<b>1,13</b>	<b>97,06</b>	<b>80,68</b>	<b>0,80</b>	<b>0,88</b>
RGB	90,00	1,39	95,70	73,32	0,72	0,84
<b>XYZ</b>	91,52	0,69	95,20	<b>80,82</b>	0,77	0,88
YCrCb	89,83	2,29	95,75	72,50	0,71	0,84

Observa-se que com o MHCL 3x3 a maior sensibilidade continua sendo obtida em imagens no espaço de cores XYZ. Entretanto, em uma análise geral, os melhores resultados obtidos pelo SVM são com o experimento realizado no espaço LUV. Novamente, assim como visto com o método MHCG, a análise estatística dos histogramas gera modelos de classificação com especificidade mais elevada. Neste caso, 97,06%. Mas, como também ocorre com o MHCG, a sensibilidade diminui. Porém o MHCL 3x3 ainda mantém essa medida em níveis acima de 80%. A acurácia média obtida apresenta desvio padrão baixo,

indicando consistência nos resultados. Este é o primeiro modelo experimentado a obter índice Kappa de 0,8, enquadrando-se na categoria excelente.

Como dito na Seção 4.2.3.2, a divisão em mais *grids* busca por padrões latentes que podem ser identificados em nível local. Portanto, também foi experimentada classificação com características extraídas pelo método HCL com *grid* 5x5, cujos resultados são apresentados pela Tabela 12.

Tabela 12 – Resultados da classificação para a abordagem HCL *grid* 5x5.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
<b>HLS</b>	90,67	0,84	90,90	<b>90,00</b>	0,76	0,89
HSV	89,66	1,10	90,22	88,02	0,74	0,89
LAB	89,83	1,98	90,90	86,70	0,74	0,88
LUV	85,93	1,28	85,70	86,70	0,66	0,86
<b>RGB</b>	<b>91,35</b>	<b>0,92</b>	<b>92,02</b>	<b>89,34</b>	<b>0,78</b>	<b>0,90</b>
XYZ	90,67	1,19	91,17	89,17	0,76	0,90
YCrCb	84,32	2,44	82,37	90,00	0,63	0,86

Os níveis de acurácia ainda se mantêm na faixa entre 91% e 92%. No entanto, percebe-se para essa abordagem, que o SVM alcançou em seu melhor resultado um certo equilíbrio entre as medidas de especificidade (92,02%) e sensibilidade (89,34%). Ressalta-se que o desvio padrão da acurácia média é pequeno, indicando consistência, e que a área sob a curva ROC é muito próxima de 1, corroborando o bom desempenho do SVM para este tipo de característica.

De forma análoga aos demais experimentos, também foi analisado o desempenho do SVM para a classificação das características obtidas via análise estatística do HCL 5x5. A Tabela 13 apresenta os resultados alcançados.

Tabela 13 – Resultados da classificação para a abordagem MHCL *grid* 5x5.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	91,86	1,28	98,42	72,66	0,76	0,85
HSV	91,35	1,25	97,26	74,00	0,75	0,85
LAB	94,23	1,83	98,20	82,66	0,84	0,90
LUV	93,38	1,93	97,06	82,66	0,82	0,89
RGB	92,71	1,75	97,50	78,64	0,79	0,88
XYZ	90,46	1,44	95,45	75,85	0,73	0,85
<b>YCrCb</b>	<b>94,27</b>	<b>1,06</b>	<b>97,75</b>	<b>84,17</b>	<b>0,84</b>	<b>0,90</b>

Com as características das imagens no espaço de cores YCrCb, o SVM alcançou acurácia de 94,27%, apresentando um desvio padrão baixo. Corroborando análises

anteriores, os métodos de extração de características baseados em análise estatística do histograma contribuíram para que o classificador gerasse maiores valores de especificidade. A sensibilidade obtida é de 84,17%. Esse é considerada a abordagem mais promissora dentre as analisadas até então, obtendo área sob a curva ROC de 0,90 e índice Kappa de 0,84. Ao considerar esses valores, pode-se dizer que o desempenho do classificador é considerado excelente.

### 5.3.3 Combinação de características de textura e cor

De forma a avaliar o poder discriminatório do classificador, também foram realizados experimentos com características combinadas de textura e cor. Para isso, vetores concatenados foram gerados, combinando cada abordagem de análise de textura com a melhor avaliada entre as representações de características de cor; no caso, a análise estatística do histograma local em *grids* 5x5 (MHCL 5x5).

Os resultados obtidos para as abordagens de combinação CLCM e MHCL 5x5 e GLCM + MHCL 5x5 são apresentados, respectivamente, pelas Tabelas 14 e 15.

Tabela 14 – Resultados da classificação para a abordagem combinada CLCM + MHCL *grid* 5x5.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
<b>HLS</b>	<b>95,25</b>	<b>0,46</b>	<b>98,90</b>	<b>84,66</b>	<b>0,86</b>	<b>0,91</b>
HSV	93,72	0,46	98,90	78,68	0,82	0,88
<b>LAB</b>	94,74	0,37	97,96	<b>85,32</b>	0,85	0,91
LUV	93,38	1,83	97,74	80,68	0,81	0,89
RGB	94,06	0,84	97,96	82,64	0,83	0,90
XYZ	93,22	1,19	97,17	81,65	0,81	0,89
YCrCb	94,27	0,81	96,87	86,70	0,84	0,91

Tabela 15 – Resultados da classificação para a abordagem combinada GLCM + MHCL *grid* 5x5.

<b>Espaço de cores</b>	<b>A (%)</b>	<b>Desv.</b>	<b>E (%)</b>	<b>S (%)</b>	<b>Kappa</b>	<b>ROC</b>
HLS	93,89	0,70	98,90	79,34	0,82	0,89
HSV	93,72	1,53	98,66	79,34	0,82	0,88
LAB	93,05	3,24	97,74	79,34	0,80	0,88
LUV	94,06	1,46	97,28	84,68	0,83	0,90
RGB	94,40	0,46	98,66	81,98	0,84	0,90
XYZ	93,22	0,37	97,72	80,00	0,81	0,88
<b>YCrCb</b>	<b>95,12</b>	<b>1,10</b>	<b>98,02</b>	<b>86,70</b>	<b>0,86</b>	<b>0,92</b>

Como visto nas tabelas, as abordagens de combinação conduziram o classificador a obter valores ainda maiores de acurácia. A abordagem CLCM + MHCL 5x5 se destaca apresentando um desvio padrão consideravelmente baixo (0,46), bem como especificidade de quase 99%. A sensibilidade, no entanto, não apresentou ganhos significativos, apresentando 84,66%. Vale destacar que, no experimento com o espaço LAB, essa métrica alcançou 85,32%. Mas em linhas gerais, o experimento com maior destaque é o realizado com o espaço HLS.

Quanto a abordagem GLCM + MHCL 5x5, os testes com características extraídas no espaço YCrCb se destacam, gerando acurácia média de 95,12 e sensibilidade de 86,70%.

Ambas as abordagens de combinação se mostram promissoras, pois, de acordo com os valores de índice Kappa e área sob ROC, o classificador obteve excelente desempenho. Por fim, a Tabela 16 apresenta um resumo de todos os resultados alcançados na etapa de classificação. A Tabela organiza os resultados por acurácia, do menor ao maior desempenho.

Tabela 16 – Resumo dos resultados da etapa de classificação.

Técnica	Espaço de cores	A (%)	Desv.	E (%)	S (%)	Kappa	ROC
CLCM	RGB	90,33	0,46	97,04	70,68	0,72	0,83
HGC	LAB	90,50	1,39	92,26	85,36	0,75	0,88
HLC 3x3	XYZ	91,10	1,46	92,90	85,82	0,77	0,89
HLC 5x5	RGB	91,35	0,92	92,02	89,34	0,78	0,90
GLCM	LUV	91,86	1,53	97,94	74,02	0,76	0,86
MHGC	HSV	92,20	1,39	96,82	78,68	0,78	0,87
MHLC 3x3	LUV	92,88	1,13	97,06	80,68	0,80	0,88
MHLC 5x5	YCrCb	94,27	1,06	97,75	84,17	0,84	0,90
GLCM + MHLC 5x5	YCrCb	95,12	1,10	98,02	86,70	0,86	0,92
CLCM + MHLC 5x5	HLS	95,25	0,46	98,90	84,66	0,86	0,91

De uma maneira geral, observa-se que os resultados apresentam especificidade maior que a sensibilidade. Isso decorre do desbalanceamento da base de imagens, a qual possui mais olhos rotulados como saudáveis, classe alvo da especificidade. No entanto, as abordagens analisadas alcançaram resultados promissores mesmo com a limitação imposta pela classe problema, principalmente nos experimentos envolvendo características de cores.

Por fim, ressalta-se que durante o desenvolvimento desta proposta foram realizados testes com outros descritores de textura, como *Local Binary Patterns*, função K de Ripley (RIPLEY, 1977) e índices de diversidade filogenética (CARVALHO FILHO et al., 2017). Para análise de cores, realizaram-se testes com vetores de cores coerentes (PASS; ZABIH; MILLER, 1997). Entretanto, os resultados obtidos com essas técnicas não foram

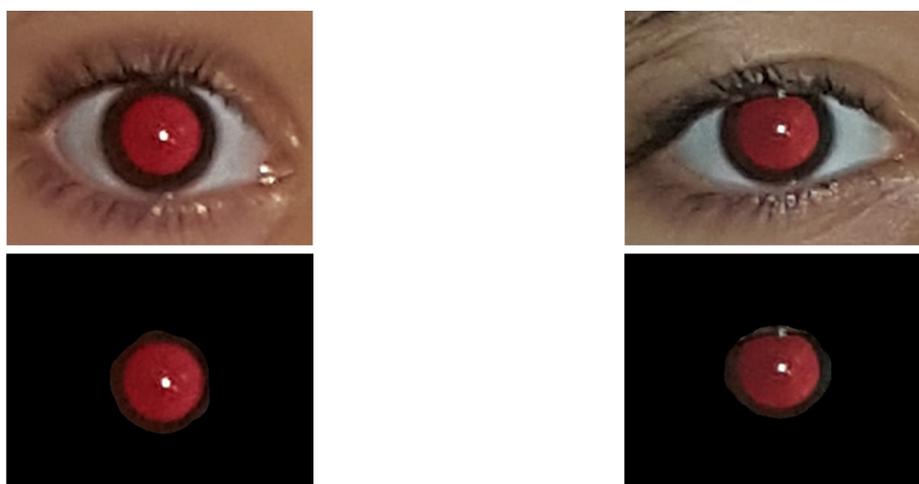
satisfatórios. Por esse motivo e também para que o texto não se estendesse em demasia, optou-se por não apresentá-los.

### 5.3.4 Estudos de caso

Nesta seção são apresentados alguns exemplos de imagens e seus resultados de classificação, fazendo-se também um estudo comparativo entre cada abordagem de representação de características.

A Figura 37 apresenta casos de olhos saudáveis classificados corretamente a partir das características de textura e cor.

Figura 37 – Olhos saudáveis classificados corretamente para todos os métodos de representação de características.

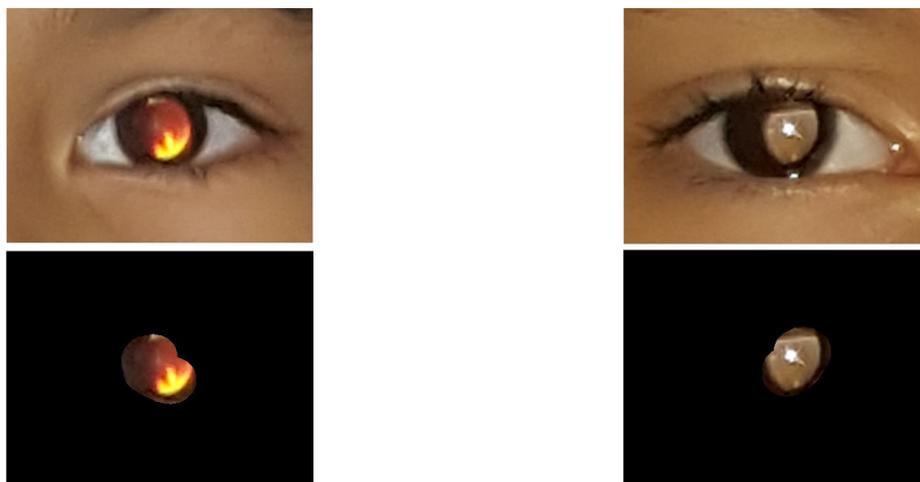


Verifica-se que, em imagens de olhos saudáveis, a cor avermelhada do reflexo é mais uniforme e intensa. E isso impacta numa representação de características mais precisa para essa classe, como corroborado pelos valores altos de especificidade apresentados pelos experimentos. Quanto aos olhos com patologia, a coloração do reflexo pode variar de acordo com cada patologia. A Figura 38 apresenta classificações corretas para esses casos.

Observa-se que a presença de manchas ou o aspecto esbranquiçado do reflexo apontam para a presença de patologias, impactando positivamente no processo de classificação. Nesses casos, considera-se que as representações das características tanto de textura quanto de cor apresentaram um poder discriminatório promissor.

Entretanto, como visto anteriormente, as abordagens de representação por análise de textura levaram o SVM a obter resultados baixos de sensibilidade em comparação com as representações de cores. Um exemplo a ser destacado é mostrado na Figura 39(a).

Figura 38 – Olhos com patologia classificados corretamente para todos os métodos de representação de características.

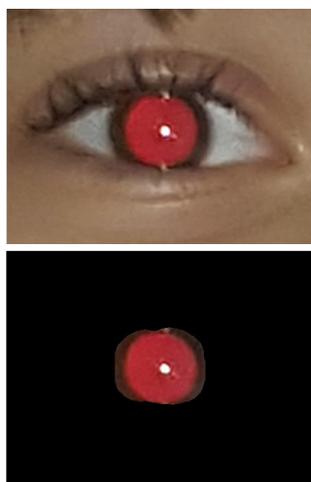


Fonte: Elaborada pelo autor.

Neste caso, o olho foi diagnosticado com patologia pelo especialista. Os resultados da classificação com o SVM divergem para as abordagens exploradas.

Figura 39 – Reflexos retinianos semelhantes entre classes.

(a) Olho com patologia.



(b) Olho saudável.



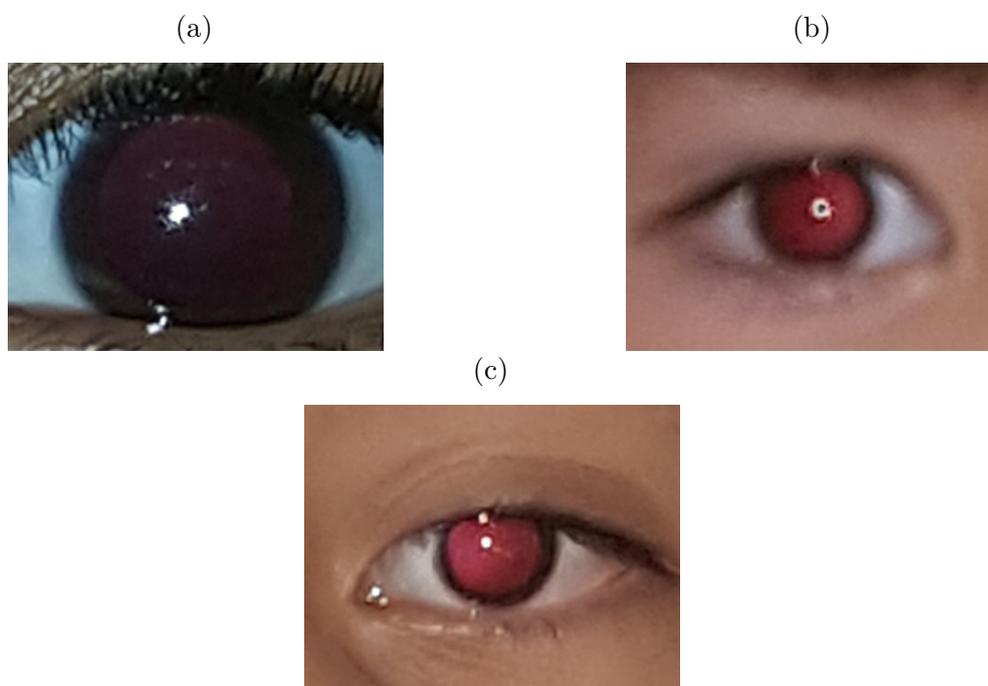
Fonte: Elaborada pelo autor.

Neste caso em específico, o paciente sofre de ametropia, o que indica erros refrativos. Comparando-se as Figuras 39(a) e 39(b), observa-se que o reflexo apresenta um comportamento similar aos casos saudáveis. Os descritores de textura não geraram características discriminatórias o suficiente para que o SVM classificasse corretamente o caso (a). Contudo, esse exemplar foi classificado de forma correta pela abordagem CLCM

+ MHLC 5x5. Isso indica que as características geradas a partir de uma análise local, combinadas com as de textura, foram discriminatórias o suficiente para que o classificador pudesse reconhecer os padrões latentes a essa categoria de imagem.

Outros casos falhos de classificação são apresentados na Figura 40. Na Figura 40(a), o olho é saudável, mas apresenta muito pouco reflexo. Isso acarretou na detecção de uma possível alteração visual visto que a ausência ou baixa intensidade do reflexo pode caracterizar uma patologia. Nos casos apresentados pelas Figuras 40(b) e (c), os reflexos têm comportamento similar ao esperado para olhos saudáveis, embora seus diagnósticos indiquem, respectivamente, microesotropia e ametropia.

Figura 40 – Casos falhos de classificação.



Fonte: Elaborada pelo autor.

Em uma análise geral, observa-se que os métodos de análise de cores geraram as características que melhor contribuíram para o bom desempenho do SVM quanto à sensibilidade. Essa observação inspirou a implementação das abordagens híbridas combinando o MHLC 5x5 às técnicas GLCM e CLCM. Essas combinações produziram, portanto, resultados promissores para o método proposto. E isso é corroborado pelos valores obtidos de ROC e índice Kappa, que enquadram o método proposto na categoria excelente.

## 6 CONCLUSÃO

O Teste de Brückner é um exame oftalmológico simples e indolor que pode ser realizado em indivíduos de qualquer idade. Também conhecido como exame do reflexo vermelho, é capaz de detectar alterações visuais como erros refrativos, catarata e retinoblastoma de forma prematura, de forma que, uma vez detectada uma patologia, o paciente pode ter seu tratamento iniciado com celeridade.

Neste estudo foi proposto um método computacional para a detecção automática da presença de patologias na visão em imagens do Teste de Brückner. Para isso, coletou-se imagens de pacientes submetidos ao exame. Cada olho foi rotulado pelo especialista como integrante de uma das seguintes categorias: saudável ou com patologia.

O método consiste de 4 etapas principais. A localização dos olhos é a primeira. Para essa tarefa, treinou-se uma arquitetura de rede convolucional YOLOv3. A rede encontrou os olhos em todas as imagens, com precisão de 95,93% e sensibilidade 100%.

A segunda etapa do método é a segmentação do reflexo retiniano. Para essa tarefa, utiliza-se duas redes U-Net sendo uma para segmentar imagens no canal G e outra no canal R do espaço de cores RGB. As máscaras geradas pelas redes são combinadas e produzem o mapa de segmentação final. Essa abordagem obtém coeficiente Dice de 87,45% para a U-Net treinada para o canal G e 87,60% para a U-Net treinada para o canal R.

A terceira etapa é a extração de características. Neste trabalho, foram utilizadas as abordagens de matrizes de coocorrência em nível de cinza (GLCM) e colorido (CLCM) para representar as características de textura; e os histogramas de cores em nível global e local como descritores de cor. Também foi proposta a combinação dessas abordagens a fim de se analisar o desempenho do classificador e avaliar qual método de extração foi o mais eficaz.

Na etapa de classificação, com o uso do SVM, o melhor resultado foi obtido com o método que combina as matrizes CLCM com as métricas do histograma de cores local *grid* 5x5, obtendo 95,25% de acurácia, 98,90% de especificidade, 84,66% de sensibilidade, 0,86 de índice Kappa e área sob a curva ROC de 0,91.

Esses resultados trazem perspectivas favoráveis ao método e corroboram a eficácia das técnicas utilizadas para cada tarefa designada. Vale ressaltar que esse método em si é uma contribuição para a literatura e que, vistos os resultados bem avaliados pelas

métricas de validação, projeta-se a possibilidade de incorporá-lo em uma ferramenta CAD.

## 6.1 Trabalhos Futuros

Não obstante, algumas tarefas podem ser realizadas para contribuir positivamente com a metodologia. Seguem sugestões para trabalhos futuros:

- Ampliar a base de imagens para que se construa um modelo de aprendizado mais robusto;
- Elaborar o estudo de uma abordagem de avaliação por pacientes analisando a similaridade entre as características dos dois olhos. Em Sun et al. (2016) esse critério também é utilizado como parte do Teste de Brückner.
- Realizar testes com outras técnicas de extração de característica de textura, como funções geo-estatísticas (ALMEIDA et al., 2012) e índices de diversidade filogenéticos (CARVALHO FILHO et al., 2017);
- Testar outras abordagens de análise de cores, como vetores de cores coerentes em nível local (PASS; ZABIH; MILLER, 1997), ou histogramas locais com outros modelos de divisão de regiões, como círculos e/ou anéis;
- Avaliar a utilização de modelos de aprendizagem profunda na etapa de classificação, como *Capsule Nets* (SABOUR; FROSST; HINTON, 2017) ou *Deep Features*;
- Classificar os tipos de patologia.

## 6.2 Produções científicas

A Tabela 17 apresenta os artigos científicos publicados que possuem relação com o método proposto pelo presente trabalho.

Tabela 17 – Artigos publicados que possuem relação com o método proposto.

Anais	Título	Qualis
ICIAR 2018	Segmentation of the Retinal Reflex in Brückner Test Images using U-Net Convolutional Network	B1
SBCAS 2018	Teste Automático de Brückner baseado em Imagens	B4

## REFERÊNCIAS

- AGUIAR, A. S. C. D. **Validação de Tecnologia para Avaliação do Teste do Reflexo Vermelho**. Dissertação (Mestrado) — Universidade Federal do Ceará, 2010.
- ALI, T.; KHAN, A.; KIM, I. Automatic detection and correction of red-eye effect. In: IEEE. **Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on**. [S.l.], 2009. p. 1–4.
- ALMEIDA, J. D.; SILVA, A. C.; PAIVA, A.; TEIXEIRA, J. A. Metodologia computacional para detecção e diagnóstico automáticos, e planejamento cirúrgico do estrabismo. In: **XXIX Congresso da Sociedade Brasileira de Computação**. [S.l.: s.n.], 2009.
- ALMEIDA, J. D. S. de; SILVA, A. C.; PAIVA, A. C. de; TEIXEIRA, J. A. M. Computational methodology for automatic detection of strabismus in digital images through hirschberg test. **Computers in Biology and Medicine**, v. 42, n. 1, p. 135 – 146, 2012. ISSN 0010-4825. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0010482511002149>>.
- AREL, I.; ROSE, D. C.; KARNOWSKI, T. P. Deep machine learning - a new frontier in artificial intelligence research [research frontier]. **IEEE Computational Intelligence Magazine**, v. 5, n. 4, p. 13–18, Nov 2010. ISSN 1556-603X.
- BENCO, M.; HUDEC, R.; KAMENCAY, P.; ZACHARIASOVA, M.; MATUSKA, S. An advanced approach to extraction of colour texture features based on glm. **International Journal of Advanced Robotic Systems**, v. 11, p. 1, 07 2014.
- BENEDICT, S. R.; KUMAR, J. S. Geometric shaped facial feature extraction for face recognition. In: IEEE. **Advances in Computer Applications (ICACA), IEEE International Conference on**. [S.l.], 2016. p. 275–278.
- BLAND, M. **An Introduction to Medical Statistics (3rd ed.)**. [S.l.]: Oup Oxford, 2010.
- CAGLIARI, P. Z.; SILVA, J. C.; VERAS, T. N.; VIEIRA, C. E. F.; BERTELLI, L. J.; RAMOS, M. C. Alterações detectadas pelo teste do reflexo vermelho. **Arquivos Catarinenses de Medicina**, v. 45, n. 3, p. 48–57, 2016.
- CARVALHO FILHO, A. O. de; SILVA, A. C.; PAIVA, A. C. de; NUNES, R. A.; GATTASS, M. Lung-nodule classification based on computed tomography using taxonomic diversity indexes and an svm. **Journal of Signal Processing Systems**, v. 87, n. 2, p. 179–196, May 2017.
- CBO. **Um olhar sobre o Brasil. Censo Oftalmológico 2014**. 2014. <[http://www.cbo.net.br/novo/publicacoes/Olhar\\_sobre\\_o\\_Brasil.pdf](http://www.cbo.net.br/novo/publicacoes/Olhar_sobre_o_Brasil.pdf)>. "Acesso em 13/12/2018".
- CHANG, C.-C.; LIN, C.-J. Libsvm: A library for support vector machines. **ACM Trans. Intell. Syst. Technol.**, ACM, New York, NY, USA, v. 2, n. 3, p. 27:1–27:27, maio 2011. ISSN 2157-6904. Disponível em: <<http://doi.acm.org/10.1145/1961189.1961199>>.
- CHOLLET, F. et al. **Keras**. [S.l.]: GitHub, 2015. <<https://github.com/keras-team/keras>>. "Acesso em 23/01/2019".

- CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, Springer, v. 20, n. 3, p. 273–297, 1995.
- FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861 – 874, 2006. ISSN 0167-8655. ROC Analysis in Pattern Recognition. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S016786550500303X>>.
- GAUBATZ, M.; ULICHNEY, R. Automatic red-eye detection and correction. In: IEEE. **Image Processing. 2002. Proceedings. 2002 International Conference on**. [S.l.], 2002. v. 1, p. I–I.
- GONZALEZ, R. C.; WOODS, R. C. **Processamento Digital de Imagens**. [S.l.]: Pearson Prentice Hall, 2010.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>.
- GUO, Y.; LIU, Y.; OERLEMANS, A.; LAO, S.; WU, S.; LEW, M. S. Deep learning for visual understanding: A review. **Neurocomputing**, v. 187, p. 27 – 48, 2016. ISSN 0925-2312. Recent Developments on Deep Big Vision.
- HALL, M.; FRANK, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; WITTEN, I. H. The weka data mining software: An update. **SIGKDD Explor. Newsl.**, ACM, New York, NY, USA, v. 11, n. 1, p. 10–18, nov. 2009. ISSN 1931-0145. Disponível em: <<http://doi.acm.org/10.1145/1656274.1656278>>.
- HARALICK, R. M.; SHANMUGAM, K. et al. Textural features for image classification. **IEEE Transactions on systems, man, and cybernetics**, Ieee, n. 6, p. 610–621, 1973.
- HENNING, R.; RIVAS-PEREA, P.; SHAW, B.; HAMERLY, G. A convolutional neural network approach for classifying leukocoria. In: IEEE. **Image Analysis and Interpretation (SSIAI), 2014 IEEE Southwest Symposium on**. [S.l.], 2014. p. 9–12.
- HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. **Neural Networks**, v. 2, n. 5, p. 359 – 366, 1989. ISSN 0893-6080.
- HUA, K.-L.; HSU, C.-H.; HIDAYATI, S. C.; CHENG, W.-H.; CHEN, Y.-J. Computer-aided classification of lung nodules on computed tomography images via deep learning technique. **OncoTargets and therapy**, Dove Press, v. 8, 2015.
- JALIS, M. Use of bruckner test for the detection of significant refractive errors in children. **Journal of Rawalpindi Medical College**, v. 19, n. 3, p. 200–203, 2015.
- JERRY, C. L. L.; EIZENMAN, M. Convolutional neural networks for eye detection in remote gaze estimation systems. In: **Proceedings of the International MultiConference of Engineers and Computer Scientists**. [S.l.: s.n.], 2008. v. 1.
- JOGI, R. **Basic Ophthalmology**. 4th. ed. New Delhi: Jypee Brother Medical Publisher ltd, 2009.

- KRAWCZYK, Z.; STARZYRNSKI, J. Bones detection in the pelvic area on the basis of yolo neural network. **2018 19th International Conference "Computational Problems of Electrical Engineering"(CPEE)**, p. 1–4, 09 2018.
- KUMAR, N.; BERG, A. C.; BELHUMEUR, P. N.; NAYAR, S. K. Attribute and simile classifiers for face verification. In: IEEE. **Computer Vision, 2009 IEEE 12th International Conference on**. [S.l.], 2009. p. 365–372.
- KUMAR, V.; BHOOSHAN, S.; SOOD, A.; SHAHI, R.; MENDIRATTA, S. A novel technique for automated red eye detection and correction. In: IEEE. **Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on**. [S.l.], 2009. p. 1–5.
- LAN, W.; DANG, J.; WANG, Y.; WANG, S. Pedestrian detection based on yolo network model. **2018 IEEE International Conference on Mechatronics and Automation (ICMA)**, p. 1547–1551, 08 2018.
- LAN, Z.; MU, Y. An efficient eye localization method based on geometrical approach and susan operator. In: IEEE. **Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on**. [S.l.], 2016. p. 899–902.
- LANDIS, J. R.; KOCH, G. G. An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers. **Biometrics**, JSTOR, p. 363–374, 1977.
- LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, Nov 1998. ISSN 0018-9219.
- LEPISTO, L.; LAUNIAINEN, A.; KUNTTU, I. Red eye detection using color and shape. In: IEEE. **Local and Non-Local Approximation in Image Processing, 2009. LNLA 2009. International Workshop on**. [S.l.], 2009. p. 153–157.
- LIU, W.; WANG, Z.; LIU, X.; ZENG, N.; LIU, Y.; ALSAADI, F. E. A survey of deep neural network architectures and their applications. **Neurocomputing**, v. 234, p. 11 – 26, 2017. ISSN 0925-2312.
- LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2015. p. 3431–3440.
- MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. In: **Proceedings of the 30th International Conference on Machine Learning**. [S.l.: s.n.], 2013. v. 30, n. 1, p. 3.
- MAIA, J. S. e Christian Cabacinha e Adriana Assis e Thiago Monteiro e Carlos Alberto Araújo Júnior e R. Redes neurais artificiais para estimar a densidade básica de madeiras do cerrado. **Pesquisa Florestal Brasileira**, v. 38, n. 0, 2018.
- MARQUES FILHO, O.; VIEIRA NETO, H. **Processamento digital de imagens**. [S.l.]: BRASPORT, 1999.

MILLER, J. M.; HALL, H. L.; GREIVENKAMP, J. E.; GUYTON, D. L. Quantification of the brückner test for strabismus. **Investigative ophthalmology & visual science**, The Association for Research in Vision and Ophthalmology, v. 36, n. 5, p. 897–905, 1995.

NOH, H.; HONG, S.; HAN, B. Learning deconvolution network for semantic segmentation. In: **Computer Vision (ICCV), 2015 IEEE International Conference on**. [S.l.: s.n.], 2015.

PASS, G.; ZABIH, R.; MILLER, J. Comparing images using color coherence vectors. In: **ACM. Proceedings of the fourth ACM international conference on Multimedia**. [S.l.], 1997. p. 65–73.

PEDRINI, H.; SCHWARTZ, W. R. **Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações**. [S.l.]: Editora Thomson Learning, 2007. 528 p. ISBN 978-85-221-0595-3.

PRASHASTHI, M.; SHRAVYA, K.; DEEPAK, A.; MULIMANI, M.; SHASHIDHAR, K. G. Image processing approach to diagnose eye diseases. In: **SPRINGER. Asian Conference on Intelligent Information and Database Systems**. [S.l.], 2017. p. 245–254.

REDMON, J. Yolov3: An incremental improvement. **Computing Research Repository (CoRR)**, 2018. Disponível em: <<http://arxiv.org/abs/1804.02767>>.

REDMON, J.; DIVVALA, S. K.; GIRSHICK, R. B.; FARHADI, A. You only look once: Unified, real-time object detection. **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, abs/1506.02640, 2015. Disponível em: <<http://arxiv.org/abs/1506.02640>>.

REDMON, J.; FARHADI, A. Yolo9000: better, faster, stronger. **2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, 2017.

RIPLEY, B. D. Modelling spatial patterns. **Journal of the Royal Statistical Society. Series B (Methodological)**, JSTOR, p. 172–212, 1977.

RIVAS-PEREA, P.; BAKER, E.; HAMERLY, G.; SHAW, B. F. Detection of leukocoria using a soft fusion of expert classifiers under non-clinical settings. **BMC ophthalmology**, BioMed Central, v. 14, n. 1, p. 110, 2014.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: **SPRINGER. International Conference on Medical Image Computing and Computer-Assisted Intervention**. [S.l.], 2015. p. 234–241.

ROSENFELD, G. H.; FITZPATRICK-LINS, K. A coefficient of agreement as a measure of thematic classification accuracy. **Photogrammetric engineering and remote sensing**, American Society for Photogrammetry and Remote Sensing, v. 52, n. 2, p. 223–227, 1986.

RUSS, J. C. **The Image Processing Handbook, Fifth Edition (Image Processing Handbook)**. Boca Raton, FL, USA: CRC Press, Inc., 2006. ISBN 0849372542.

- SABOUR, S.; FROSST, N.; HINTON, G. E. Dynamic routing between capsules. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2017. p. 3856–3866.
- SALAKHUTDINOV, R.; HINTON, G. Deep boltzmann machines. In: **Artificial Intelligence and Statistics**. [S.l.: s.n.], 2009. p. 448–455.
- SALMI, M.; BOUCHEHAM, B. Content based image retrieval based on cell color coherence vector (cell-ccv). In: IEEE. **ISKO-Maghreb: Concepts and Tools for knowledge Management (ISKO-Maghreb), 2014 4th International Symposium**. [S.l.], 2014. p. 1–5.
- SERRA, J. Introduction to mathematical morphology. **Computer Vision, Graphics, and Image Processing**, Academic Press, v. 35, n. 3, p. 283 – 305, 1986. ISSN 0734-189X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0734189X86900022>>.
- SMITH, S. M.; BRADY, J. M. Susan—a new approach to low level image processing. **International journal of computer vision**, Springer, v. 23, n. 1, p. 45–78, 1997.
- SUN, M.; MA, A.; LI, F.; CHENG, K.; ZHANG, M.; YANG, H.; NIE, W.; ZHAO, B. Sensitivity and specificity of red reflex test in newborn eye screening. **The Journal of pediatrics**, Elsevier, v. 179, p. 192–196, 2016.
- SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V. Inception-v4, inception-resnet and the impact of residual connections on learning. **Computing Research Repository (CoRR)**, 2016. Disponível em: <<http://arxiv.org/abs/1602.07261>>.
- TAHA, A. A.; HANBURY, A. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. **BMC medical imaging**, BioMed Central, v. 15, n. 1, p. 29, 2015.
- WAN, S.; JIN, P.; YUE, L. An effective image retrieval technique based on color perception. In: **2011 Sixth International Conference on Image and Graphics**. [S.l.: s.n.], 2011. p. 1017–1022.
- WU, Y.; JI, Q. Learning the deep features for eye detection in uncontrolled conditions. In: IEEE. **Pattern Recognition (ICPR), 2014 22nd International Conference on**. [S.l.], 2014. p. 455–459.
- YANG, W.; JIACHUN, Z. Real-time face detection based on yolo. **2018 IEEE International Conference on Knowledge Innovation and Invention (ICKII)**, p. 221–224, 07 2018.
- YE, L.; ZHU, M.; XIA, S.; PAN, H. Cascaded convolutional neural network for eye detection under complex scenarios. In: SPRINGER. **Chinese Conference on Biometric Recognition**. [S.l.], 2014. p. 473–480.
- YOO, S.; PARK, R.-H. Red-eye detection and correction using inpainting in digital photographs. **IEEE Transactions on Consumer Electronics**, IEEE, v. 55, n. 3, 2009.

ZAYED, N.; ELNEMR, H. A. Statistical analysis of haralick texture features to discriminate lung abnormalities. **Journal of Biomedical Imaging**, Hindawi Publishing Corp., New York, NY, United States, v. 2015, p. 12:12–12:12, jan. 2015. ISSN 1687-4188. Disponível em: <<https://doi.org/10.1155/2015/267807>>.

ZHANG, L.; LU, K.; PAN, C.; XIA, S. Eye detection for electronic map control application. In: IEEE. **Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2014 Sixth International Conference on**. [S.l.], 2014. v. 1, p. 241–244.