



UNIVERSIDADE FEDERAL DO MARANHÃO
Programa de Pós-Graduação em Ciência da Computação

POLYANA BEZERRA DA COSTA

***METODOLOGIA COMPUTACIONAL PARA AUXÍLIO NO DIAGNÓSTICO
DA PARALISIA DO SEXTO NERVO ÓPTICO ATRAVÉS DE VÍDEOS
DIGITAIS***

São Luís
2020

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

POLYANA BEZERRA DA COSTA

**METODOLOGIA COMPUTACIONAL PARA AUXÍLIO NO
DIAGNÓSTICO DA PARALISIA DO SEXTO NERVO ÓPTICO
ATRAVÉS DE VÍDEOS DIGITAIS**

SÃO LUÍS – MA

2020

POLYANA BEZERRA DA COSTA

**METODOLOGIA COMPUTACIONAL PARA AUXÍLIO NO
DIAGNÓSTICO DA PARALISIA DO SEXTO NERVO ÓPTICO
ATRAVÉS DE VÍDEOS DIGITAIS**

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

2020

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

Costa, Polyana Bezerra da.

Metodologia computacional para auxílio no diagnóstico da paralisia do sexto nervo óptico através de vídeos digitais / Polyana Bezerra da Costa. - 2020.
108 f.

Orientador(a): João Dallyson Sousa de Almeida.

Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, 2020.

1. Detecção do Centro dos Olhos. 2. Isophotes. 3. Localização dos Olhos. 4. Paralisia do Sexto Nervo. 5. YOLO. I. Almeida, João Dallyson Sousa de. II. Título.

POLYANA BEZERRA DA COSTA

**METODOLOGIA COMPUTACIONAL PARA AUXÍLIO NO
DIAGNÓSTICO DA PARALISIA DO SEXTO NERVO ÓPTICO
ATRAVÉS DE VÍDEOS DIGITAIS**

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 24 de Junho de 2020.

**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. Leandro Augusto Frata
Fernandes**
Membro da Banca Examinadora

SÃO LUÍS – MA
2020

À meus pais, meu irmão e ao tio Lulu

AGRADECIMENTOS

A Deus, por me dar forças e por me ajudar nos bons e maus momentos. À minha mãe e meu pai, por todo amor e afeto, por me motivarem e apoiarem em minhas decisões, e por perguntarem diariamente se eu terminei de escrever a dissertação. Ao meu irmão Pedro, por ser positivo e sempre me dizer que as coisas iam dar certo.

Ao meu orientador, João Dallyson, obrigada por todo apoio, aprendizado e cuidado durante os mais de dois anos de orientação. Obrigada pelas sugestões, e principalmente, por sua compreensão e paciência, que parecem inesgotáveis. Ao Dr. Jorge Meireles, por sugerir o tema dessa pesquisa, por fornecer as bases de dados e sempre esclarecer minhas dúvidas, dando ótimas sugestões.

Aos meus queridos amigos Anderson e Mayara. A Anderson, obrigada por me ajudar durante todo o mestrado, seja nas disciplinas ou na dissertação, por sua positividade e companheirismo, e por sempre me lembrar dos prazos. À Mayara, obrigada pelos excelentes conselhos e companheirismo. A Petterson, Thayane, Kelson, Anderson e Mayara novamente, obrigada por me ajudarem nas incontáveis vezes em que eu pedi que ligassem meu computador para que eu pudesse executar meus testes. A Jonnison, Ítalo, Ricardo e Paulo Renato, obrigada por todos os *insights* e sugestões em relação à dissertação. Aos participantes do projeto SAUL, obrigada por me acolheram e ajudarem durante minha breve participação neste projeto.

À Isabela, obrigada pelo constante apoio, amizade e motivação. A Rodrigo e Wallas, meus quase-chefes, obrigada por toda ajuda, aprendizado e paciência. Em especial, a Wallas, por perguntar todo dia se eu terminei a dissertação, de alguma forma, funcionou como motivação. Ao meu chefe Márcio, obrigada por ser extremamente compreensivo e por me incentivar a focar nos estudos.

Aos professores Anselmo, Ari, Dallyson e Geraldo, obrigada por serem uma fonte de inspiração por sua dedicação ao trabalho.

A todos os professores do mestrado, obrigada pelo conhecimento passado e pela inspiração. Obrigada também a todas as pessoas que dedicaram um pouco de tempo para responder as dúvidas postadas no *Stack Overflow*.

Por fim, obrigada a todas as pessoas que participaram da construção da base de vídeos usada nessa dissertação. Sem a colaboração deles, este trabalho não poderia ter acontecido.

“Where there is life, there is hope.”

The Hobbit

RESUMO

O sexto nervo óptico, também conhecido como nervo abducente, está diretamente ligado à contração do músculo reto lateral. A paralisia deste nervo impede que alguns dos músculos que controlam o movimento dos olhos funcionem adequadamente, causando dor de cabeça, enxaqueca, visão turva, vertigem e visão dupla. É importante que esta paralisia seja diagnosticada nos estágios iniciais, para que seja tratada sem deixar sequelas. Os métodos usuais para diagnóstico desta paralisia são invasivos ou dependentes de equipamentos caros, e métodos computacionais voltados diretamente para a detecção ou diagnóstico da mesma, até o momento da escrita deste trabalho, não foram encontrados. Portanto, um método não invasivo e de baixo custo pode ser útil para apoiar ou guiar o diagnóstico do oftalmologista. Neste contexto, este trabalho apresenta uma metodologia computacional para auxiliar o diagnóstico da paralisia do sexto nervo óptico utilizando vídeos, a fim de assistir oftalmologistas no processo de análise, servindo como uma opinião complementar. O método proposto usa redes neurais convolucionais e técnicas de processamento de imagens para registrar a trajetória de movimentação dos olhos durante o vídeo. A partir desse registro, calcula-se a velocidade de movimentação de cada olho, comparando-as para determinar se o olho é saudável ou parético, visto que olhos com paralisia se movem mais devagar que olhos saudáveis. Os resultados obtidos com o método proposto sugerem que olhos com paralisia movem-se cerca de 19,65% mais devagar que olhos saudáveis. Esse limiar, junto com a velocidade média de movimentação dos olhos, pode ajudar oftalmologistas em sua análise. Testes realizados mostraram que o método atingiu 92,64% de acurácia no diagnóstico da paralisia do sexto nervo, com índice *Kappa* de 0,925, o que ressalta a confiabilidade dos resultados e dá perspectivas favoráveis para uma futura aplicação clínica do método.

Palavras-chave: Paralisia do Sexto Nervo Óptico, Redes Neurais Convolucionais, Processamento de Imagens, Vídeos Digitais.

ABSTRACT

The sixth cranial nerve, also known as the abducens nerve, is responsible for controlling the movements of the lateral rectus muscle. Palsies on the sixth nerve prevent some muscles that control eye movements from proper functioning, causing headaches, migraines, blurred vision, vertigo, and double vision. Hence, such palsy should be diagnosed in the early stages so that it can be treated without leaving any sequela. The usual methods for diagnosing the sixth nerve palsy are invasive or depend on expensive equipment, and computer-based methods designed specifically to assist on the diagnose of the aforementioned palsy were not found, until the publication of this work. Therefore, a low-cost, non-invasive method can support or guide the ophthalmologist's diagnosis. In this context, this work presents a computational methodology to aid in diagnosing the sixth nerve palsy using videos, in order to assist ophthalmologists in the diagnostic process, serving as a second opinion. The proposed method uses convolutional neural networks and image processing techniques to track both eyes' movement trajectory during the video. With this trajectory, it is possible to calculate the average speed in which each eye moves. Since it is known that parietic eyes move slower than healthy eyes, comparing the average speed of both eyes can determine if the eye is healthy or parietic. The results obtained with the proposed method showed that parietic eyes move at least 19.65% slower than healthy ones. This threshold, along with the average speed of the movement of the eyes, can help ophthalmologists in their analysis. The proposed method reached 92.64% accuracy in the diagnosis of the sixth nerve palsy, with a *Kappa* index of 0.925, which highlights the reliability of the results and gives favorable perspectives for further clinical application.

Keywords: Sixth Nerve Palsy, Convolutional Neural Networks, Image Processing, Digital Videos

LISTA DE ILUSTRAÇÕES

Figura 1 – Principais estruturas do olho humano.	28
Figura 2 – Ilustração dos músculos extraoculares e os movimentos proporcionados por eles.	29
Figura 3 – Exemplos da Posição Primária do Olhar e Duções.	30
Figura 4 – Exemplos de Versões e Vergências.	31
Figura 5 – Exemplos de como executar movimentos sacádicos e persecutórios. . .	32
Figura 6 – Relação entre o sexto nervo e o músculo reto lateral.	33
Figura 7 – Paralisia do sexto nervo no olho esquerdo, com grande limitação de abdução.	34
Figura 8 – Formas tradicionais de avaliar a presença da paralisia do sexto nervo óptico.	36
Figura 9 – Exemplo de uma imagem digital bidimensional.	37
Figura 10 – Exemplo de uma imagem em escala de cinza e sua representação matricial com os valores de cada <i>pixel</i> . Nesta escala de cor, os valores de intensidade variam de $[0, 255]$	37
Figura 11 – Principais etapas do Processamento Digital de Imagens.	38
Figura 12 – Passo a passo de um convolução em 2D com um <i>kernel</i> de tamanho 2×2	39
Figura 13 – Resultado da operação convolução da imagem (x) pela máscara (w) de tamanho 3×3	40
Figura 14 – Aplicação da Filtragem Gaussiana com diferentes tamanhos de <i>kernel</i> . Em ambos os casos, o valor do σ aplicado é 1.	41
Figura 15 – Exemplo da aplicação do operador gradiente em uma imagem.	42
Figura 16 – Exemplo da aplicação do operador Laplaciano de <i>kernel</i> (a) sob a imagem (b).	44
Figura 17 – Etapas do <i>Isophotes</i> . (a) Mapa de Curvaturas <i>Isophotes</i> . (b) Vetores de deslocamento. (c) Mapa de centros. (d) Centro de maior valor (ponto vermelho).	45
Figura 18 – Estrutura de funcionamento de um neurônio artificial.	46
Figura 19 – Arquitetura de uma rede MLP com duas camadas ocultas.	48
Figura 20 – Exemplo das etapas de processamento em uma CNN.	52

Figura 21 – <i>Pipeline</i> de processamento da CNN YOLO.	54
Figura 22 – O bloco básico de convoluções da YOLOv3.	55
Figura 23 – Bloco residual da YOLOv3.	56
Figura 24 – Camadas finais da YOLOv3.	56
Figura 25 – Ilustração de como a saída rede é obtida. Já que a rede usa regressão para informar a saída, a última convolução é simples, sem função de ativação definida, para que a saída seja dada por uma ativação linear.	56
Figura 26 – Arquitetura completa da YOLOv3.	57
Figura 27 – Principais etapas da metodologia desenvolvida.	63
Figura 28 – Exemplo dos movimentos oculares executados nos vídeos da base.	65
Figura 29 – <i>Loss</i> do treino e validação nas primeiras 50 épocas nas duas abordagens.	66
Figura 30 – <i>Loss</i> do treino e validação nas últimas 50 épocas nas duas abordagens.	67
Figura 31 – Exemplos de resultados indesejados.	68
Figura 32 – <i>Bounding boxes</i> resultantes após pós-processamento.	70
Figura 33 – Centros de olhos obtidos com a YOLOv3 (Pontos azuis).	71
Figura 34 – Comparação dos resultados da YOLOv3 e <i>Isophotes</i>	72
Figura 35 – Trajetória de movimentação dos dois olhos num intervalo de tempo. Em verde: olho esquerdo; em azul : olho direito.	73
Figura 36 – Exemplos de olhos que a abordagem <i>Not Freezing</i> não conseguiu detectar.	78
Figura 37 – Exemplo de olhos que a YOLO teve dificuldade em detectar.	79
Figura 38 – Trajetória do deslocamento dos olhos nos vídeos 3 (paciente apresenta paralisia) e 26 (ambos os olhos são saudáveis).	84
Figura 39 – Exemplo de uso da ferramenta desenvolvida para análise de partes da trajetória de movimentação dos olhos, referente ao vídeo 5.	85
Figura 40 – Trajetória de movimentação dos olhos presentes nos vídeos 1 a 6.	101
Figura 41 – Trajetória de movimentação dos olhos presentes nos vídeos 7 a 13.	102
Figura 42 – Trajetória de movimentação dos olhos presentes nos vídeos 14 a 19.	103
Figura 43 – Trajetória de movimentação dos olhos presentes nos vídeos 20 a 27.	104
Figura 44 – Trajetória de movimentação dos olhos presentes nos vídeos 28 a 32.	105

LISTA DE TABELAS

Tabela 1 – Abordagens para localização dos olhos.	23
Tabela 2 – Abordagens para determinar centro dos olhos.	25
Tabela 3 – Interpretação dos valores do coeficiente Kappa.	61
Tabela 4 – Resultado da Precisão da detecção em ambas as abordagens.	77
Tabela 5 – Precisão e Sensibilidade média da YOLOv3 ao ser testada nos vídeos da base.	78
Tabela 6 – Porcentagem do erro em relação ao centro dos olhos ao aplicar o método de <i>Isophotes</i> em diferentes resoluções de imagem.	79
Tabela 7 – Erro médio das abordagens testadas para detecção do centro do olho.	80
Tabela 8 – Erro médio para vídeos que não seguiram o protocolo de aquisição e para vídeos que seguiram.	81
Tabela 9 – Precisão e Sensibilidade da detecção de olhos da rede YOLO, e erro médio da detecção do centro dos olhos em X para cada vídeo. Vídeos de 1 a 16 foram gravados antes da definição do protocolo de aquisição e vídeos de 17 a 32 foram gravados de acordo com o protocolo de aquisição.	81
Tabela 10 – Velocidade Média de cada olho em pacientes em que um dos olhos apresenta a paralisia do sexto nervo óptico. A velocidade é dada em <i>pixels</i> /segundo.	82
Tabela 11 – Velocidade Média de cada olho em pacientes que não apresentam paralisia do sexto nervo óptico. A velocidade é dada em <i>pixels</i> /segundo.	83
Tabela 12 – Resultados obtidos com a validação cruzada usando 5 grupos.	87
Tabela 13 – Resultado final do diagnóstico dos pacientes em doentes (apresentam a paralisia do sexto nervo óptico) e saudáveis.	87
Tabela 14 – Artigo publicado que possui relação com o método proposto.	90

LISTA DE ABREVIATURAS E SIGLAS

CAD	<i>Computer Aided Diagnosis</i>
CBO	Conselho Brasileiro de Oftalmologia
CNN	<i>Convolutional Neural Networks</i>
DEXTRO	Dextroversão
DMA	Desvio Médio Absoluto
DPI	<i>Dots Per Inch</i>
EOG	Eletro-oculografia
FP	Falso-Positivo
FPS	<i>Frames Per Second</i>
GARD	Genetic and Rare Diseases Information Center
HCC	<i>Haar Cascade Classifiers</i>
HOG	<i>Histogram of Oriented Gradients</i>
INFRA	Infraversão
IOU	<i>Intersection Over Union</i>
LEVO	Levoversão
MLP	<i>MultiLayer Perceptron</i>
OI	Oblíquo Inferior
OMS	Organização Mundial de Saúde
OS	Oblíquo Superior
PDI	Processamento Digital de Imagens
RI	Reto Inferior
RL	Reto Lateral

RM	Reto Medial
RS	Reto Superior
SI	Sistema Internacional de Unidades
SVM	<i>Support Vector Machine</i>
SUPRA	Supraversão
TGF	Teste de Geração de Força
YOLO	<i>You Only Look Once</i>
YOLOv2	<i>You Only Look Once</i> versão 2
YOLOv3	<i>You Only Look Once</i> versão 3

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Objetivo	18
1.2	Objetivos Específicos	18
1.3	Organização do trabalho	18
2	TRABALHOS RELACIONADOS	20
3	FUNDAMENTOS TEÓRICOS	27
3.1	O Olho Humano	27
3.2	Fundamentos de Oculomotricidade	28
3.2.1	Sacadas	31
3.2.2	Movimento Persecutório	32
3.3	A Paralisia do Sexto Nervo Óptico	33
3.4	Processamento Digital de Imagens	36
3.4.1	Convolução	38
3.4.2	Suavização Através de Filtragem Gaussiana	40
3.4.3	Cálculo do Gradiente	41
3.4.4	Cálculo da Derivada de Segunda Ordem	42
3.4.5	Curvatura de Isophotes	43
3.5	Redes Neurais	46
3.5.1	Redes <i>FeedForward</i> de Múltiplas Camadas	47
3.5.2	Aprendizagem Profunda	50
3.5.3	Redes Neurais Convolucionais	51
3.5.4	YOLO - <i>You Only Look Once</i>	52
3.5.4.1	Arquitetura da YOLOv3	54
3.6	Filtro de Hampel	58
3.7	Métricas de Avaliação	59
3.8	Considerações Finais	61
4	MATERIAIS E MÉTODO	62
4.1	Método Proposto	62
4.1.1	Aquisição da Base de Vídeos	63

4.1.2	Localização dos Olhos	65
4.1.2.1	Treinamento da Rede YOLOv3	65
4.1.2.2	Pós-processamento de <i>Bounding Boxes</i>	67
4.1.3	Detecção do Centro dos Olhos	70
4.1.4	Registro da Trajetória de Movimentação dos Olhos	72
4.1.5	Cálculo da Velocidade Média	74
4.1.6	Diagnóstico da Paralisia do Sexto Nervo	75
4.2	Hardware e Softwares Utilizados	75
4.3	Considerações Finais	76
5	RESULTADOS E DISCUSSÃO	77
5.1	Localização da Região dos Olhos	77
5.2	Detecção do Centro dos Olhos	78
5.3	Velocidade Média	82
5.4	Diagnóstico da Paralisia do Sexto Nervo	86
6	CONCLUSÃO	88
6.1	Trabalhos Futuros	89
6.2	Produções Científicas	90
	REFERÊNCIAS	91
	APÊNDICES	100
	APÊNDICE A – REGISTROS	101
A.1	Trajetória de Movimentação dos Olhos em Vídeos	101

1 INTRODUÇÃO

O olho humano é um dos órgãos mais complexos e importantes do corpo humano, de tal maneira que o cérebro dedica mais espaço ao processamento e armazenamento de informações visuais do que de todos os outros sentidos combinados (SHERMAN, 2019). Em resultado disso, estima-se que 80% das coisas que percebemos dão-se através dos olhos (ZEISS, 2017). Em um processo cognitivo complexo, os olhos captam o ambiente ao redor através da luminosidade, formando uma imagem nítida quando os componentes do sistema visual estão precisamente integrados. Entretanto, doenças oculares podem interferir na transparência e regularidade da superfície dos olhos, bem como na estabilidade da anatomia ocular, prejudicando o bom funcionamento do sistema visual (SIHOTA; TANDON, 2011).

Um estudo publicado em 2019 pela Organização Mundial de Saúde (OMS) reportou que, mundialmente, pelo menos 2,2 bilhões de pessoas são afetadas por alguma doença ocular ou cegueira, dentre os quais, cerca de 1 bilhão de casos poderiam ter sido evitados (OMS, 2019). Nacionalmente, um estudo do Conselho Brasileiro de Oftalmologia (CBO) publicado em 2019 estimou que 0,75% da população tem cegueira, cerca de 1,6 milhões de pessoas (OTTAIANO et al., 2019). Ainda segundo a OMS, a chave para controlar a deficiência visual e outras patologias oculares é a prestação acessível e eficaz de serviços oftalmológicos, visto que mais de 75% dos problemas visuais poderiam ser evitados, curados ou controlados se diagnosticados precocemente.

Entretanto, especialmente em países em desenvolvimento, serviços oftalmológicos especializados não estão disponíveis para toda a população, ou quando estão, concentram-se nas grandes cidade. De acordo com o CBO, apenas 15% dos municípios brasileiros contam com o atendimento de especialistas em saúde ocular. Apesar de possuir um número considerável de oftalmologistas - mais de 16 mil - a má distribuição desses profissionais faz com que 23% da população brasileira esteja desassistida (CBO, 2014). Tendo em vista essa lacuna, métodos que possam auxiliar no processo de análise ou diagnóstico de patologias oculares podem ser grandes aliados para oftalmologistas e pacientes.

Crescentes avanços nas áreas de Processamento de Imagens Digitais e Aprendizado de Máquina, dentre outros, possibilitaram o desenvolvimento de sistemas computacionais para auxílio no diagnóstico de diversas patologias com base em imagens médicas, chamados

de *Computer Aided Diagnosis* (CAD) (DOI, 2007). CADs não tem por objetivo substituir médicos, mas sim, assisti-los no processo de diagnóstico, servindo como uma opinião complementar, aumentando a acurácia no diagnóstico, e contribuindo para a descoberta de anomalias em estágios iniciais. Em relação a patologias oculares, já foram desenvolvidos CADs para auxiliar na detecção e análise de Glaucoma (CLARO et al., 2016; SOUSA et al., 2017; ARAÚJO et al., 2019), Degeneração Macular Relacionada à Idade (SRINIVASAN et al., 2014), Estrabismo (ALMEIDA et al., 2015; VALENTE, 2017), patologias relacionados ao Reflexo Vermelho Retiniano (KUMAR et al., 2009; SILVA et al., 2018), etc. Devido à natureza complicada de seu diagnóstico, outra patologia ocular que poderia usar da ajuda de CADs é a Paralisia do Sexto Nervo Óptico.

O sexto nervo óptico, também conhecido como nervo abducente, é um importante nervo que está diretamente ligado à contração do músculo reto lateral (ver Seção 3.2) (EHRENHAU, 2018). Este músculo permite que o olho se mova em direção ao canto externo, possibilitando o direcionamento do olhar para a esquerda ou direita. A paralisia deste nervo impede que alguns dos músculos que controlam o movimento dos olhos funcionem adequadamente (GARD, 2015). Além disso, o mau funcionamento do nervo pode causar dor de cabeça, enxaqueca, visão turva, vertigem e visão dupla (diplopia) quando a pessoa tenta movimentar o olho em direção ao canto externo (RUBIN, 2018) (CURI; COSTA; BARROSO, 2013).

A paralisia do sexto nervo óptico é considerada uma disfunção rara; estudos feitos em crianças norte-americanas mostraram que ocorrem 2,5 casos a cada 100 mil habitantes (EHRENHAU, 2018). O diagnóstico deve ser feito o mais breve possível, pois quando a causa é tratada, a paralisia geralmente desaparece sem deixar sequelas. Assim, a avaliação clínica e neurológica é essencial para o diagnóstico e tratamento adequado. A avaliação oftalmológica visa observar alterações motoras, suas consequências no sistema oculomotor e sua evolução, a fim de minimizar os sintomas durante a evolução da doença e evitar sequelas (CURI; COSTA; BARROSO, 2013).

Existem duas maneiras principais de detectar a paralisia do sexto nervo, que são o Teste de Geração de Força (TGF) e a Eletro-oculografia (EOG). De acordo com Curi, Costa e Barroso (2013), o TGF explora o potencial da força muscular e pode diferenciar entre paralisia parcial ou total. É um teste simples e confiável, mas o resultado depende da

experiência do examinador. Após a anestesia tópica¹, um fórceps (ver Seção 3.3) é anexado ao membro temporal do olho lesionado, em seguida, o olho é levado para a adução (mover o olho em direção ao lado nasal) e o paciente é solicitado a mover rapidamente o olho em direção ao músculo reto lateral examinado, fazendo com que o examinador sinta a força desse músculo no fórceps. A intensidade de força sentida é então analisada empiricamente pelo oftalmologista, que vai decidir se há paralisia, e se ela é parcial ou total (CURI; COSTA; BARROSO, 2013). Como esse teste não fornece um valor para a intensidade da força, a sua análise é subjetiva e depende muito da experiência do oftalmologista.

A outra maneira de avaliar a força muscular ocular é por meio do exame de eletro-oculografia (EOG), que registra os movimentos oculares durante os testes eletro-nistagmográficos. Tais testes captam a variação do potencial córneo-retiniano durante a movimentação dos olhos. Para medir e acompanhar os movimentos dos olhos no exame de EOG, eletrodos são colocados ao redor dos olhos; quando os movimentos se aproximam dos eletrodos, uma diferença de potencial é registrada para ser usada como uma medida para a posição do olho (MARTINSEN; GRIMNES, 2011). Exames como o EOG informam a velocidade sacádica (ver Seção 3.2.1) com que cada olho se move, que é usada para avaliar a presença da paralisia (METZ, 1976).

Ambas as técnicas são eficazes na detecção da paralisia do sexto nervo, mas possuem suas desvantagens. O TGF é um procedimento invasivo e dependente da experiência do oftalmologista, além disso, a sua execução em crianças pequenas não é possível, visto que o exame exige a colaboração do paciente. Por sua vez, a EOG é um exame que depende de equipamentos de alto custo, que só se encontram em grandes laboratórios de pesquisa, e praticamente inviáveis de se terem em consultórios oftalmológicos comuns. A fim de apoiar ou orientar o diagnóstico do oftalmologista com um método não invasivo, acessível e de baixo custo, este trabalho apresenta uma metodologia computacional para auxílio no diagnóstico da paralisia do sexto nervo óptico através da quantificação da velocidade média dos olhos, obtida através do processamento de vídeos. Obtém-se a velocidade média comum com que cada olho se move durante um período de tempo, fazendo o mapeamento da movimentação de cada olho, a fim de auxiliar o oftalmologista a diagnosticar se há paralisia ou não, visto que um olho com paralisia se move mais devagar que um olho

¹ A anestesia tópica é uma forma de anestesia sem injeção ao redor do olho, aplicada através de colírios anestésicos (HAGEMANN, 2020). Estudos mostram que esse tipo de anestesia causa menos dor intra-operatória, baixo índice de complicações, mais conforto e recuperação visual mais rápida. Entretanto, a sua efetividade analgésica é menor (GUEDES, 2007; COELHO et al., 2005).

saudável. Assim, serão fornecidos dados que poderão guiar os oftalmologistas em sua análise além de sugerir o percentual de diferença de velocidade para diferenciar olhos saudáveis de olhos que apresentam a paralisia do sexto nervo.

1.1 Objetivo

Diante do contexto apresentado, este trabalho tem por objetivo propor um método computacional, automático e de baixo custo, para auxiliar no diagnóstico da paralisia do sexto nervo óptico através de vídeos adquiridos enquanto o paciente realiza movimentos persecutórios. Para isso, utiliza-se técnicas de Processamento de Imagens e Aprendizagem de Máquina em vídeos de pacientes que apresentam a paralisia e em pacientes que não apresentam.

1.2 Objetivos Específicos

Para atingir o objetivo geral, os seguintes desafios precisam ser satisfeitos:

- Definir um protocolo para aquisição dos vídeos a serem utilizados no método proposto;
- Construir uma base de dados de vídeos com portadores da paralisia e com pessoas saudáveis, registrando a movimentação dos olhos;
- Desenvolver um método para localizar olhos em vídeos;
- Desenvolver um método para localizar a pupila;
- Registrar a trajetória de movimentação dos olhos;
- Calcular a velocidade com que cada olho se move, a fim de sugerir aos especialistas um limiar que distinga olhos saudáveis de olhos com a paralisia;
- Comparar os resultados obtidos pelo método proposto com o diagnóstico do médico especialista, para fins de validação;

1.3 Organização do trabalho

Este trabalho está organizado da seguinte forma: o **Capítulo 2** apresenta os trabalhos relacionados que serviram como referência para esta dissertação. O **Capítulo**

3 apresenta toda a fundamentação teórica deste trabalho, explicando todos os conceitos e técnicas necessárias para o entendimento da metodologia desenvolvida. Este capítulo cobre assuntos como fundamentos de oculomotricidade, explicação da paralisia do sexto nervo, técnicas de Processamento de Imagens e Redes Neurais, dentre outros. O **Capítulo 4** apresenta o método desenvolvido neste trabalho, descrevendo cada uma de suas etapas, desde a obtenção da base de dados usada, até a obtenção dos resultados. O **Capítulo 5** apresenta os resultados obtidos com a metodologia desenvolvida, bem como os discute e os interpreta. Por fim, o **Capítulo 6** apresenta uma visão geral sobre o trabalho desenvolvido, avaliando sua eficácia, pontos em que falhou, pontos em que poderia melhorar, bem como as contribuições do trabalho, e apresenta ideias para dar continuidade ao trabalho.

2 TRABALHOS RELACIONADOS

Vários trabalhos têm utilizado metodologias computacionais para pré-diagnosticar automaticamente patologias oculares, ou para assistir ao oftalmologista em seu diagnóstico (KUMAR et al., 2009; SRINIVASAN et al., 2014; ALMEIDA et al., 2015; CLARO et al., 2016; SOUSA et al., 2017; VALENTE, 2017; SILVA et al., 2018; ARAÚJO et al., 2019). Entretanto, ao se tratar da paralisia do sexto nervo, métodos computacionais voltados diretamente para a detecção ou diagnóstico desta paralisia, por meio de imagens ou vídeos digitais, até o momento da escrita desta dissertação e usando os meios de busca tradicionais, não foram encontrados. Portanto, esta seção apresenta trabalhos dedicados ao diagnóstico da paralisia do sexto nervo óptico, mesmo que estes não sejam feitos através de métodos computacionais. Além disso, este capítulo apresenta trabalhos que abordam a localização da região dos olhos e da pupila usando métodos computacionais, apresentados em ordem cronológica. Por fim, apresentam-se trabalhos de diagnóstico e detecção de patologias oculares através de imagens e vídeos.

Em relação ao diagnóstico da paralisia do sexto nervo óptico, Metz (1983) utilizou uma medida chamada de velocidade sacádica para analisar a força dos músculos oculares. Segundo o autor, as sacadas (um tipo de movimento ocular) estão diretamente relacionadas à força produzida por um músculo extra-ocular, sendo bons indicativos da força do músculo. Para monitorar os movimentos sacádicos de cada olho, o autor usou a eletro-oculografia (EOG). Após analisar os exames de 37 pacientes que possuem a paralisia, em casos de pacientes com paralisia total, o autor concluiu que a velocidade do olho que possui a paralisia é consideravelmente menor que a do olho saudável. Enquanto olhos saudáveis apresentaram velocidade sacádica de $200^\circ/\text{segundo}$ a $320^\circ/\text{segundo}$, olhos com a paralisia variaram entre $40^\circ/\text{segundo}$ em caso de paralisia total e $160^\circ/\text{segundo}$ em caso de paralisia parcial. Sendo assim, a diferença de velocidade entre um olho saudável e um olho com paralisia pode variar de 20% e 87,5%. Em casos de pacientes com paralisia parcial, o autor também analisou a trajetória do movimento ocular para acompanhar a redução na velocidade. Por sua vez, um outro trabalho que também usou EOG para identificar a paralisia do sexto nervo, reportou que a diferença média de velocidade sacádica entre um olho saudável e um olho com paralisia é de 40% ou mais, nos casos de paralisia total, e abaixo de 40% nos casos parciais (ROSENBAUM; SANTIAGO, 1999). Ou seja, ainda

não há um consenso na literatura quanto a magnitude da diferença de velocidade entre olhos saudáveis e olhos com paralisia.

Quanto à localização da região dos olhos em imagens, diversos trabalhos abordam esta tarefa (SONG; CHI; LIU, 2006; KASINSKI; SCHMIDT, 2010; MONZO et al., 2011; NI; SUN, 2018; CUI. et al., 2019). Uma dessas abordagens é através da segmentação. Song, Chi e Liu (2006) usaram o conhecimento prévio de como os componentes da face estão dispostos na mesma para segmentar os olhos e em seguida, definir a região em que os olhos estão localizados baseado no resultado da segmentação. A partir de imagens do rosto em nível de cinza, os autores aplicaram a transformada de *wavelet* (GONZALEZ; WOODS, 2010) para obter uma imagem que contenha somente bordas. Na imagem resultante, aplicaram vários limiares para excluir as bordas referentes ao contorno do rosto e cabelo, deixando apenas conjuntos de *pixels* correspondentes às regiões dos olhos, sobrancelhas, nariz e boca. A partir daí, os autores usaram conhecimento prévio da estrutura da face para localizar a região dos olhos. Partindo-se do princípio de que a distância vertical entre olhos e nariz é normalmente maior do que entre os olhos e as sobrancelhas, maior que entre nariz e boca, que há dois olhos e duas sobrancelhas, mas apenas uma boca e um nariz no rosto, os autores usaram o método de projeção integral horizontal e vertical para segmentar a região ocular (ZHOU; GENG, 2004). Ao aplicar a metodologia desenvolvida nas bases de imagens BERN e AR, os olhos foram localizados corretamente em 98,7% e 96,6% delas, respectivamente.

Uma das formas mais utilizadas na localização da região dos olhos dá-se através dos classificadores em Cascata de Haar - do inglês *Haar Cascade Classifiers* (HCC) (VIOLA; JONES, 2001). Kasinski e Schmidt (2010) combinaram a arquitetura de detecção hierárquica dos HCC com regras baseadas em conhecimento. Os autores aplicaram uma regra básica de alinhamento de olhos para descartar falso-positivos (FP); caso um par de olhos apresente ângulo entre si maior que 20° , este será um FP. Ao aplicarem a metodologia por eles desenvolvida, em um conjunto de 10 mil imagens, os olhos foram localizados corretamente em 94% delas. Apesar do bom resultado, por serem hierárquicos, a fim de detectar olhos, os HCC precisam detectar a face primeiro. Em imagens em que o rosto não aparece por completo, esta técnica não seria a mais indicada. A base de vídeos apresentada nesta dissertação contém imagens de pacientes cujo rosto não está completamente visível, portanto, a técnica de HCC não é apropriada para uso neste trabalho.

Outras abordagens combinam HCC a outras técnicas, a fim de melhorar seu desempenho. Como exemplo, Monzo et al. (2011) usaram HCC para pré-selecionar regiões candidatas à olhos, e nessas regiões, aplicaram o descritor HOG - do inglês *Histograms of Oriented Gradients* (MCCONNELL, 1986) - para extrair características, e a partir delas, usaram o método de classificação supervisionada SVM - do inglês *Support Vector Machine* (CORTES; VAPNIK, 1995) - para para diferenciar olhos e não-olhos. Ao aplicar essa metodologia nas bases de imagens de olhos FERET e FRGCv1, foram localizados pelo menos 1 par de olhos válido em 97,12% das imagens da primeira base, e em 99,28% da segunda base, sendo o estado da arte nessas bases de dados na época em que o artigo foi publicado.

Outra forma de localizar a região dos olhos que tem ganhado popularidade nos últimos anos é através de Redes Neurais Convolucionais - do inglês *Convolutional Neural Networks* (CNN - ver Seção 3.5.3). Ni e Sun (2018) usaram a CNN YOLOv2 (ver Seção 3.5.4) na primeira fase de seu trabalho, em que desenvolveram um pupilômetro para medir a distância entre as pupilas através de estímulos visuais. O sistema desenvolvido possui 3 partes principais: detecção dos olhos, rastreamento dos olhos e estimação do tamanho da pupila. Na parte de detecção da região dos olhos, a utilização da YOLOv2 garantiu alta acurácia e robustez na detecção dos olhos mesmo sob interferência da luz ambiente. Resultados experimentais em um *dataset* privado apresentaram acurácia de detecção de 90,6%.

A Tabela 1 apresenta um resumo das abordagens mais usadas para localizar a região dos olhos e mostra a abordagem usada neste trabalho. Na metodologia a ser apresentada nesta dissertação, escolheu-se a CNN YOLOv3 para localizar os olhos devido ao seu ótimo desempenho quando aplicada em vídeos (funcionando em tempo quase real), e por contemplar as características da base de vídeos usada neste trabalho. A CNN escolhida não precisa detectar a face inteira para detectar os olhos - limitação apresentada pelos métodos que usam HCC como em Kasinski e Schmidt (2010) e Monzo et al. (2011) - o que pode garantir o seu bom funcionamento nos vídeos em que apenas os olhos do paciente estão visíveis. Além disso, outra vantagem de usar a CNN YOLOv3 se estende à capacidade de generalização dos olhos, que podem ser detectados em iluminações e posições diferentes - algo que é difícil de se alcançar em métodos muito parametrizados, como em Song, Chi e Liu (2006). Em relação à segunda versão da CNN YOLO, usada por

Ni e Sun (2018), a terceira versão da mesma é mais precisa na detecção, principalmente em relação à objetos pequenos. Essa característica se mostra apropriada para detecção dos olhos, que são uma pequena porção da face. Portanto, por se encaixar nas características presentes na base de vídeos, a YOLOv3 foi a técnica escolhida para a primeira fase da metodologia deste trabalho.

Tabela 1 – Abordagens para localização dos olhos.

Autor	Abordagem	Imagens	Tamanho da Base
Song, Chi e Liu (2006)	- Transformada de Wavelet, Limiarização	BERN e AR	150 e 564
Kasinski e Schmidt (2010)	- Classificadores em Cascata de Haar (HCC)	Base proprietária	10.000
Monzo et al. (2011)	- HCC, HOG e SVM	FERET e FRGCv1	3.365 e 5.658
Ni e Sun (2018)	- CNN YOLOv2	Base proprietária	18.603
Neste trabalho	- CNN YOLOv3	Base proprietária	9.445

Outra etapa importante do método proposto neste trabalho é a detecção do centro dos olhos. Diversos trabalhos tem abordado este problema, usando desde métodos baseado em formas (IVINS; PORRILL, 1998; TIAN; KANADE; COHN, 2000; HANSEN; PECE, 2005), a métodos baseados em características locais (PENTLAND et al., 1994; HUANG; WECHSLER, 1999; HUANG; MARIANI, 2000) como luminosidade (HALLINAN, 1991; GRAUMAN et al., 2001). Uma das técnicas mais utilizadas da abordagem baseada em forma é a Transformada de *Hough*. Como a íris tem formato circular, Khairosfaizal e Nor'aini (2009) usaram a Transformada Circular de *Hough* (KIMME; BALLARD; SKLANSKY, 1975) para delimitar o limbo e a posição da pupila, a fim de marcar pares de olhos. A pupila, ou centro do olho, é marcada como o centro do círculo obtido pela transformada, e a forma circular da íris é delimitada a partir da posição da pupila e do tamanho de raio especificado durante a aplicação da transformada. Os autores aplicaram a técnica em 50 imagens do *dataset* Face DB em que as pessoas aparecem com olhos abertos, obtendo acurácia de 86% na detecção dos olhos e seus centros. Apesar de não ser fortemente afetada por imagens ruidosas, e por detectar formas mesmo que estas estejam parcialmente oclusas, a Transformada Circular de *Hough* é uma técnica muito parametrizada, que requer vários testes e ajustes de parâmetros para que se possa obter bons resultados.

Usar os padrões de luminosidade dos olhos também pode ser uma forma de

detectar o centro dos mesmos. Os olhos são caracterizados por padrões de brilho simétricos radialmente, tendo isso em vista, Valenti e Gevers (2008) propuseram uma nova abordagem para localização do centro ocular baseada na observação desses padrões. O método proposto pelos autores utiliza *Isophotes* (POESCHL, 1984) ou isofotos para inferir o centro de padrões semi-circulares de imagens de baixa resolução, através de um esquema de votação, atingindo acurácia de aproximadamente 98% na localização do centro do olho. Também usando o método de *Isophotes*, Haritha et al. (2016) projetaram um sistema para rastrear a movimentação do olho de uma pessoa através de vídeos. Primeiramente, detecta-se a região dos olhos utilizando um *template* dos olhos, em seguida, a região é recortada e o método de *Isophotes* é aplicado a fim de obter a posição do centro da pupila. O procedimento é aplicado em cada *frame* ou quadro obtido pela câmera. O método apresentou acurácia em relação ao centro dos olhos de 99,20%. Nesta dissertação, também testou-se a utilização do método de *Isophotes* para detectar o centro dos olhos. Entretanto, inicialmente, as imagens em que o método seria aplicado tinham boa resolução e foram redimensionadas.

Outra forma de detectar o centro dos olhos é através de filtros de correlação (GONZALEZ; WOODS, 2010). No trabalho desenvolvido por Ouabida, Essadique e Bouzid (2017), usa-se princípios da ótica de Fourier para encontrar a densidade espectral de uma imagem I ao ser conjugada a um filtro de referência R . A ideia básica desta abordagem é usar uma imagem de referência R , parecida com o padrão que se quer encontrar na imagem I . Após a filtragem da imagem I pelo filtro R , as partes da imagem I que forem similares a R são destacadas, apresentando altos valores de correlação. Parte da metodologia dos autores visava detectar o centro dos olhos, para isso, os filtros de correlação usados foram imagens de olhos e pupilas. Ao aplicar tais filtros em imagens que possuem olhos, o ponto de maior correlação será o do centro da pupila. Os autores aplicaram a técnica desenvolvida em 4 bases de imagens de íris: CASIA v4, WVU non-ideal, MMU2 e UBIRIS v2, obtendo erro máximo em relação ao centro da pupila de 2,10%.

As CNNs também têm sido usadas para detectar o centro dos olhos, como exemplo, Vera-Olmos e Malpica (2017) desenvolveram um método robusto para detecção de pupila que funciona em condições adversas de iluminação. Os autores usaram uma arquitetura baseada em duas redes, uma para obter a região de interesse (ROI) - do inglês *Region Of Interest* - que contém a pupila, e outra rede para obter a posição precisa do centro da pupila. A metodologia proposta pelos autores faz segmentação semântica, em que há uma etapa

de extração de características baseadas em convoluções e *max pooling*, seguida por uma etapa de *upsampling* seguida por deconvoluções, similar à arquitetura de *encoder/decoder* da CNN U-Net (RONNEBERGER; FISCHER; BROX, 2015). A primeira rede faz uma segmentação mais grosseira da pupila, recebendo como entrada uma imagem da região dos olhos, enquanto que a segunda rede recebe como entrada a região referente à segmentação da primeira rede, fazendo uma segmentação mais precisa e apresentando como resultado a posição central da pupila. Ao testar o modelo desenvolvido em 18 *datasets* de imagens de olhos da Universidade de Tuebingen, os autores obtiveram resultados 19% melhores que o estado da arte. Entretanto, o método desenvolvido é extremamente custoso, sendo recomendado para execução *offline* e com hardware sofisticado.

A Tabela 2 apresenta um resumo das abordagens mais usadas para localização do centro dos olhos, enquanto apresenta a abordagem que será usada neste trabalho. Na metodologia desenvolvida, uma das abordagens testadas para a detecção de centro dos olhos foi o método de *Isophotes*, por este não ser um método muito parametrizado, funcionando bem em imagens com ruído e por apresentar alta acurácia.

Tabela 2 – Abordagens para determinar centro dos olhos.

Autor	Abordagem	Imagens	Tamanho da Base
Jerry e Eizenman (2008)	- CNNs em cascata	Universidade de Tuebingen	60.000
Valenti e Gevers (2008)	- <i>Isophotes</i>	BioID e Yale Face B	1.521 e 5.760
Khairosfaizal e Nor'aini (2009)	- Transformada de <i>Hough</i>	Face DB	72
Ouabida, Essadique e Bouzid (2017)	- Filtros de correlação	CASIA v4 etc.	54.607
Neste trabalho	- <i>Isophotes</i>	Base proprietária	18.344

Quanto à sistemas voltados para a detecção e diagnóstico de patologias oculares através de imagens e vídeos, os crescentes avanços nas áreas de Processamento de Imagens Digitais e Inteligência Artificial têm contribuído para a popularização de tais sistemas, chamados de CADs. Como exemplo, Valente (2017) desenvolveu um método para detecção de estrabismo em vídeos digitais baseado no exame de *CoverTest*, através de Técnicas de Processamento de Imagens e Reconhecimento de Padrões. O sistema desenvolvido viabilizaria atingir pessoas de diversas faixas etárias para tratamento do estrabismo, incluindo a melhor fase para tratamento, a infância.

Ainda no campo do estrabismo, Almeida et al. (2012) desenvolveram um método para auxiliar na detecção e diagnóstico do estrabismo através de imagens e do método de *Hirschberg*. A metodologia desenvolvida usou técnicas de processamento de imagens, funções geoestatísticas, *Support Vector Machine* e *Support Vector Regression* para detectar estrabismo automaticamente. Os resultados obtidos foram promissores, apresentando acurácia de 88% na identificação de esotropias, 100% em exotropias, 80,33% nas hipertrofia e 83,33% nas hipotrofia.

Silva et al. (2018) desenvolveram um método automático de detecção de patologia na visão baseado em imagens do Teste de *Bruckner*. Esse teste pode ser usado como parte da rotina de exames de crianças recém-nascidas, contribuindo para a identificação precoce de problemas oculares, como a catarata congênita, erros refrativos e retinoblastoma. O método desenvolvido usou algoritmos de Processamento de Imagens e Aprendizado de Máquina para auxiliar especialistas na análise da saúde visual dos pacientes, informando a qualidade apresentada pelo reflexo retiniano durante o exame, classificando cada olho como saudável ou não-saudável.

Já Cruz et al. (2020) desenvolveram um método para assistir especialistas no diagnóstico da síndrome do olho seco, através da classificação automática da camada lipídica do filme lacrimal. Ao usar índices de diversidade filogenética para extração de características e diversos classificadores, o método desenvolvido apresentou acurácia superior a 99%, mostrando ser uma alternativa viável para auxiliar especialistas.

A metodologia desenvolvida neste trabalho também visa construir novos métodos computacionais para auxiliar no diagnóstico de patologias oculares, neste caso, a paralisia do sexto nervo óptico. O processo de diagnóstico dessa paralisia envolve a realização de um exame muito invasivo, e de difícil aplicação em crianças. Outra forma de avaliar a paralisia envolve equipamentos sofisticados e caros, não encontrados na maior parte dos consultórios oftalmológicos. De forma não invasiva e de baixo custo, através de vídeos obtidos com câmeras comuns, este trabalho apresenta um método para auxiliar no diagnóstico da paralisia do sexto óptico, a fim de assistir médicos no processo de análise desta patologia, servindo como uma opinião complementar. O presente método rastreia a movimentação dos olhos para registrar sua trajetória, calculando a velocidade de movimentação de cada olho, a fim comparar as velocidades entre cada olho e determinar se o olho é saudável ou se apresenta a paralisia.

3 FUNDAMENTOS TEÓRICOS

Este capítulo descreve os fundamentos teóricos necessários para o entendimento do problema abordado neste trabalho, e as técnicas que embasam a metodologia desenvolvida. Inicialmente, explica-se o funcionamento do olho humano e o conjunto das funções que permitem a movimentação dos mesmos. Com essas informações, é possível explicar como acontece a paralisia do sexto nervo óptico (Seção 3.3), fundamental para o entendimento do domínio do problema. Em seguida, explica-se os conceitos e técnicas aplicados no desenvolvimento da metodologia, como técnicas de Processamento Digital de Imagens (Seção 3.4) e Redes Neurais (Seção 3.5).

3.1 O Olho Humano

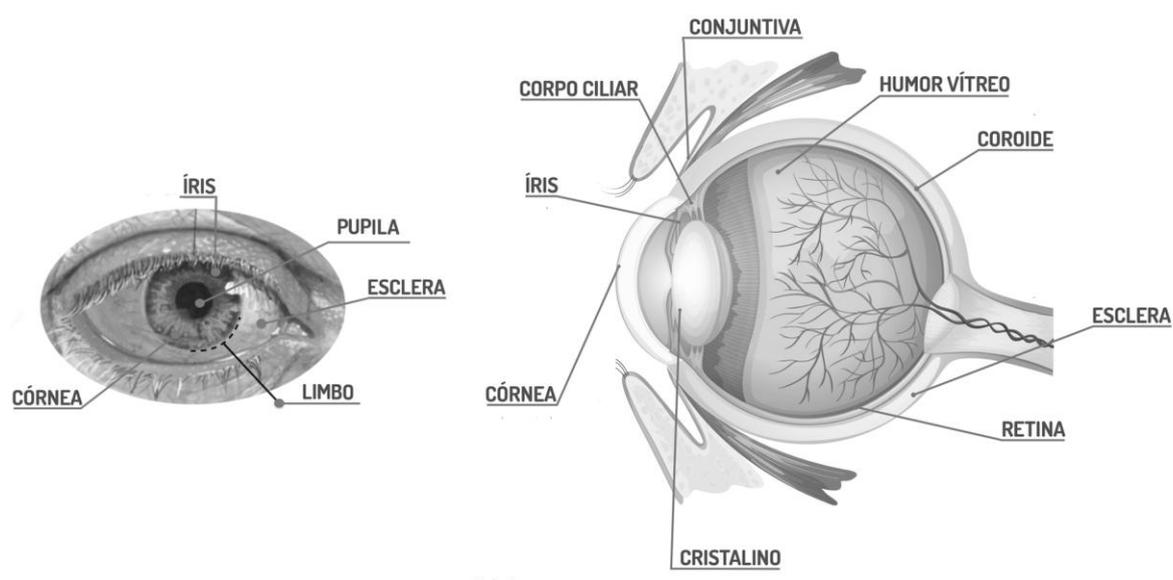
Os olhos são estruturas anatômicas complexas, formadas por tecidos delicados. Frequentemente, eles são comparados a uma câmera fotográfica, uma vez que ambos utilizam lentes para focalizar a luz incidente. No entanto, enquanto câmeras utilizam sensores ópticos para registrar a imagem, os olhos utilizam camadas especializadas de células (REMINGTON, 2005). Nos olhos humanos, são encontradas 3 camadas diferentes; na camada mais externa, encontram-se a córnea e a esclera. Já a camada intermediária divide-se em duas partes: a anterior, formada pela íris e corpo ciliar, e a posterior, formada pela coróide. Por fim, na camada interna, encontra-se a parte sensorial do olho, a retina (GONZALEZ; WOODS; COAUT, 2002).

Presente na camada externa, a córnea é responsável pela refração e por transmitir a luz para a retina, protegendo os olhos contra estruturas danosas. Além da córnea, outras estruturas contribuem para a refração, como o filme lacrimal, o humor vítreo e o cristalino. Ainda na camada externa, encontra-se a esclera. Formada por uma camada de tecido conjuntivo, a esclera também protege o olho de lesões internas e forças externas, ajudando o olho a manter a sua forma. A córnea e a esclera conectam-se no limbo - região intermediária entre a esclera e a íris.

Na camada intermediária, dentro do globo ocular, encontra-se a íris, que é responsável pelo controle da quantidade de luz que alcança a retina, mediante o aumento ou diminuição de sua abertura central, a pupila (região central e escura da íris) (GONZALEZ; WOODS; COAUT, 2002). Ainda na camada intermediária, o corpo ciliar controla a pressão

ocular, mantém o formato esférico do globo e produz fluidos intraoculares como o humor aquoso. A coróide, por sua vez, fornece oxigênio e nutrientes para a parte externa da retina. Já na camada mais interna, localiza-se a retina, a responsável por capturar e processar a luz através de sua estrutura complexa (WILLOUGHBY et al., 2010; KOLB, 2007; SIHOTA; TANDON, 2011). A Figura 1 apresenta algumas das estruturas presentes em cada uma dessas camadas, tanto externa quanto internamente.

Figura 1 – Principais estruturas do olho humano.



Fonte – Adaptado de (TURNER, 2012; ROSA, 2019).

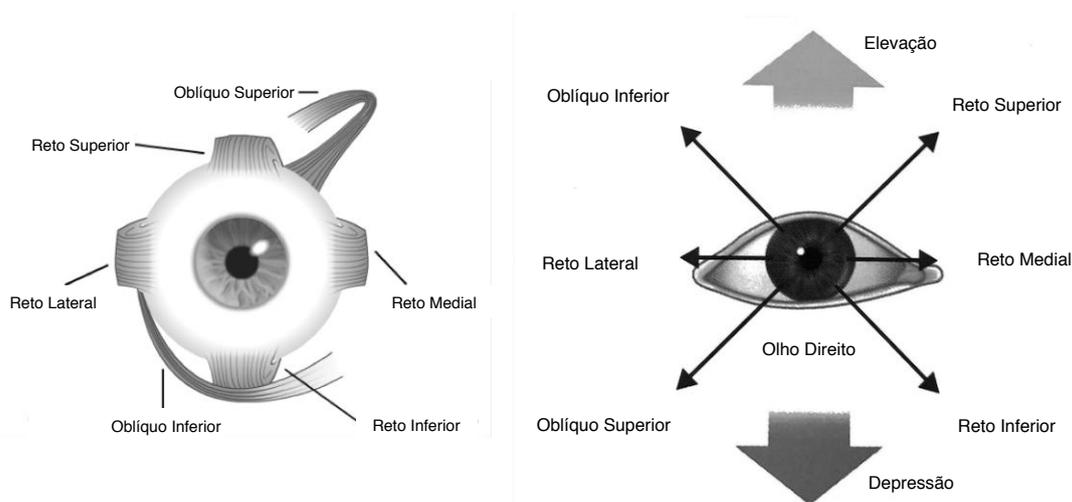
3.2 Fundamentos de Oculomotricidade

Nas tarefas do cotidiano, os movimentos oculares permitem a captação de informações do ambiente ao redor, o que não acontece de forma passiva. Os olhos buscam ativamente por informações, o que faz com que estes mudem de direção até 3 vezes por segundo (LAND; FURNEAUX, 1997). Para que os olhos se posicionem corretamente de acordo com as variadas demandas visuais, é preciso muita coordenação dos movimentos (BICAS, 2003). Ao conjunto das funções que permitem a movimentação dos olhos, dá-se o nome de oculomotricidade. Os parágrafos a seguir apresentam os conceitos básicos da oculomotricidade, como os músculos responsáveis por mover os olhos e os principais movimentos oculares.

De acordo com Snell e Lemp (2013), os 6 músculos extraoculares responsáveis

por mover os olhos são: reto superior (RS) e oblíquo inferior (OI), que tem como função primária elevar o olho; os músculos reto inferior (RI) e oblíquo superior (OS), cuja função primária é baixar o olho (depressão ou abaixamento); reto medial (RM), responsável por um dos movimentos no plano horizontal dos olhos, o de mover o olho em direção ao lado nasal; o músculo reto lateral (RL), que também é responsável pelo outro movimento no plano horizontal dos olhos, o de mover o olho em direção ao lado temporal. A Figura 2 exemplifica a ação de cada um desses músculos no olho direito.

Figura 2 – Ilustração dos músculos extraoculares e os movimentos proporcionados por eles.

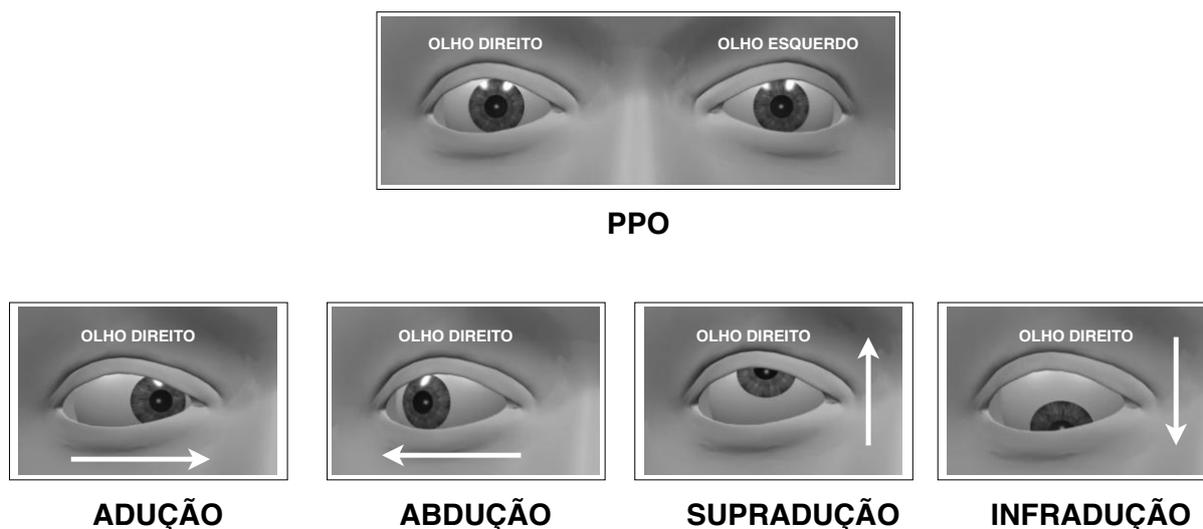


Fonte – Adaptado de (LIMA, 2015; ALVARADO, 2013).

Os movimentos oculares proporcionados pelos músculos apresentados na Figura 2 são divididos em três grupos: duções, versões e vergências (BICAS, 2003; SNELL; LEMP, 2013). Antes de descrever cada um desses grupos, define-se a posição primária do olhar (PPO), que acontece quando os olhos estão voltados diretamente para a frente. O primeiro grupo de movimentos oculares, as duções, abrangem as posições secundárias do olhar. Duções são as rotações monoculares, ou movimentos isolados de um único olho. Dentre elas, as principais são: **adução**, que é a rotação do olho em direção à região nasal; **abdução**, que refere-se à rotação do olho em direção à região temporal; há a **supradução** ou elevação, que acontece quando o olho aproxima-se do lado superior; há também a **infradução** ou abaixamento, que é a rotação do olho para baixo. Cada uma das duções são ilustradas na Figura 3, bem como a posição primária do olhar.

Outro tipo de movimento ocular são as versões. As versões são os movimen-

Figura 3 – Exemplos da Posição Primária do Olhar e Duções.



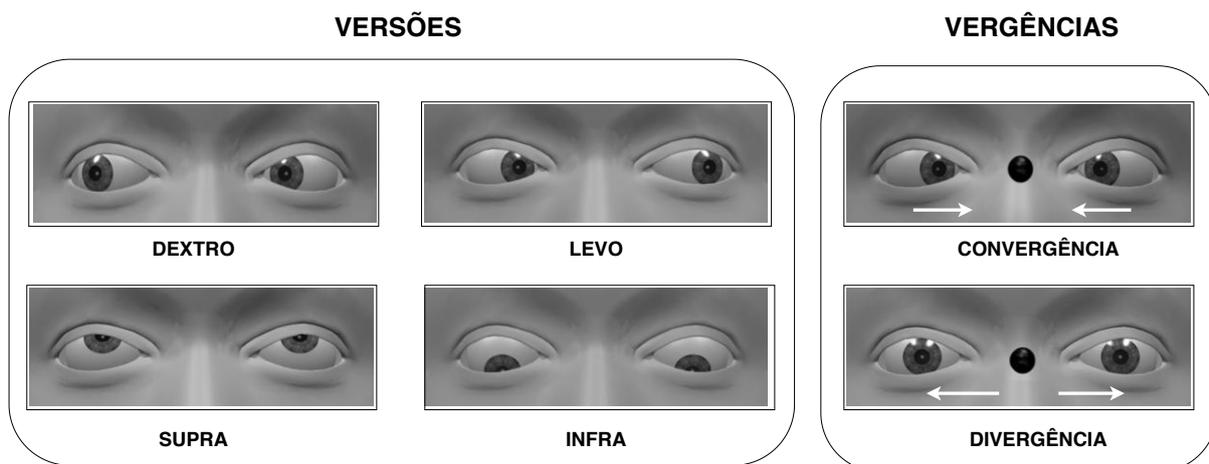
Fonte – Adaptado de (TAPSELL, 2009).

tos conjugados de ambos os olhos na mesma direção. Dentre elas, as principais são: a **dextroversão** (DEXTRO), que acontece quando os dois olhos se movimentam para a direita; há o movimento oposto de **levoversão** (LEVO), que ocorre quando os dois olhos movimentam-se para a esquerda; há a **supraversão** (SUPRA) quando os olhos se movem para cima; há também a **infraversão** (INFRA), que refere-se ao ato de olhar para baixo.

O terceiro grupo de movimentos oculares são as vergências, que abrangem o movimento conjugado dos olhos em direções opostas. Dentre as principais vergências, há a **convergência**, onde ambos os olhos fazem o movimento de adução (olho direito e esquerdo se movem em direção ao nariz); há a **divergência**, onde os olhos fazem o movimento de abdução e voltam para a posição primária do olhar. Portanto, as versões e vergências são combinações de duções nos dois olhos. Ambos os movimentos de versão e vergência são ilustrados na Figura 4.

Para o entendimento do problema abordado nesta dissertação, é preciso conhecer a função do músculo reto lateral e o movimento de abdução. Além destes conceitos, é necessário entender outros dois tipos de movimentos oculares que não estão diretamente relacionados aos músculos apresentados nesta seção: as sacadas e o movimento persecutório. Os dois movimentos serão apresentados nas subseções a seguir.

Figura 4 – Exemplos de Versões e Vergências.



Fonte – Adaptado de (TAPSELL, 2009).

3.2.1 Sacadas

De acordo com Duchowski (2007), sacadas são movimentos oculares rápidos, repentinos e simultâneos, que acontecem quando os olhos mudam o foco de um ponto de interesse para outro, como mostra a Figura 5 (a). Movimentos sacádicos são tanto voluntários quanto reflexivos; eles podem ser executados voluntariamente ou como uma medida corretiva do sistema vestibular. As sacadas podem acontecer num intervalo mínimo de 10 milissegundos, o que para registrar tais movimentos, demanda um sistema com alta frequência de amostragem (DUCHOWSKI, 2007). Sistemas com frequência de 50 a 60 *Hz* não são considerados apropriados para rastrear sacadas, apesar de serem usados (WIERTS; JANSSEN; KINGMA, 2008).

Sacadas são usadas como um indicativo da força de um músculo, e podem auxiliar na diferenciação de restrições neurogênicas e mecânicas. Uma medida obtida através do rastreamento das sacadas - a velocidade sacádica, pode ser usada para identificar várias patologias oculares, como estrabismo (BUCCI et al., 2002; SAUNDERS, 1984) e paralisias oculomotoras (LIM; ROSENBAUM; DEMER, 1995; ZIFFER et al., 1992), como a do sexto nervo óptico (METZ, 1983; ROSENBAUM; SANTIAGO, 1999). Entretanto, para se obter tal medida, é preciso usar equipamentos sofisticados com alta frequência de amostragem, como mostrado anteriormente. Tais equipamentos possuem custo alto e geralmente são encontrados em grandes laboratórios de pesquisa, o que dificulta seu uso em consultórios oftalmológicos.

Neste trabalho, as sacadas e a velocidade sacádica não serão usadas para avaliar

a presença da paralisia do sexto nervo óptico. Isso se deve à dificuldade de encontrar equipamentos aptos a rastrear as sacadas na frequência correta, e porque a base de vídeos usada nesta dissertação foi gravada em uma câmera comum com frequência de 27 FPS (do inglês *Frames Per Second*), portanto, inapta a captar movimentos tão rápidos. Além disso, o movimento ocular executado pelos pacientes durante a gravação dos vídeos não foi o movimento de sacadas, mas sim o movimento persecutório, a ser apresentado na seção seguinte.

3.2.2 Movimento Persecutório

Os movimentos persecutórios acontecem quando os olhos estão seguindo um alvo em movimento (DUCHOWSKI, 2007). Enquanto o objeto de interesse se move gradualmente, os olhos se movem suavemente para seguir um objeto em movimento para mantê-lo focado na retina. Ao contrário das sacadas, que são movimentos bruscos, o movimento persecutório é mais suave, e sua velocidade é determinada pela velocidade com que o alvo se move. Dependendo da extensão do movimento do alvo, os olhos podem corresponder à velocidade do alvo em movimento. Os movimentos oculares persecutórios são mais eficazes quando a velocidade do alvo é relativamente lenta (LISBERGER; MORRIS; TYCHSEN, 1987).

Para captar esse movimento em vídeo, basta mover um objeto suavemente nas direções desejadas e pedir que o paciente acompanhe o movimento. O movimento persecutório foi executado pelos pacientes presentes nos vídeos gravados, que seguiram a movimentação horizontal de uma caneta ou dedo durante um intervalo de tempo. Assim, utiliza-se este movimento para analisar a presença da paralisia do sexto nervo óptico. A Figura 5 (b) exemplifica como esse movimento é realizado.

Figura 5 – Exemplos de como executar movimentos sacádicos e persecutórios.



(a) Sacadas

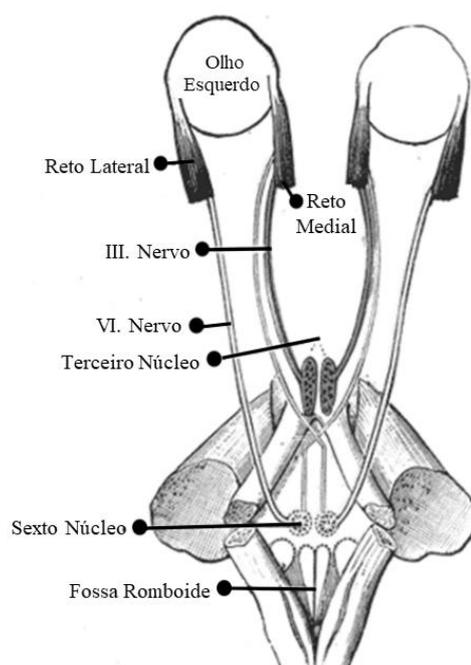
(b) Movimento Persecutório.

Fonte – Adaptado de (EMBALANCE, 2016; MORGENLANDER, 2017).

3.3 A Paralisia do Sexto Nervo Óptico

O músculo reto lateral é inervado pelo sexto nervo óptico, também chamado de nervo abducente, como ilustra a Figura 6. Este nervo é diretamente responsável pela contração do músculo reto lateral. Como apresentado na Seção 3.2, o reto lateral permite que o olho faça o movimento de abdução, ou seja, que se mova em direção ao seu canto externo. Uma lesão no sexto nervo causa a paralisia parcial ou total do músculo reto lateral, impossibilitando ou limitando a abdução (ROSENBAUM; SANTIAGO, 1999; IVAMOTO, 2014). A paralisia parcial, também chamada de paresia, acontece quando o músculo está parcialmente paralisado, o que faz com que o paciente consiga mover os olhos até um pouco depois da linha média. Na paralisia total, por sua vez, o músculo reto lateral está completamente paralisado. A Figura 7 exemplifica a diferença de movimentação entre um olho saudável e um olho com a paralisia do sexto nervo; percebe-se que um dos olhos não consegue completar o movimento de abdução, parando na posição primária do olhar. Além dessa limitação, o mau funcionamento do sexto nervo pode causar dor de cabeça, enxaqueca, visão turva, vertigem e visão dupla (diplopia) ao tentar abduzir o olho (CURI; COSTA; BARROSO, 2013; RUBIN, 2018).

Figura 6 – Relação entre o sexto nervo e o músculo reto lateral.



Fonte – Adaptado de (GRAY; CARTER, 1918).

Devido à natureza longa e curva do sexto nervo óptico, e à sua localização - perto do seio cavernoso - qualquer pressão intracraniana elevada pode prender ou “apertar” o nervo, causando paralisia (parcial ou total) do reto lateral (ROSENBAUM; SANTIAGO, 1999). Além disso, as causas para tal paralisia podem variar de acordo com a idade. Em adultos, é mais provável que a paralisia se manifeste em pessoas com mais de 40 anos que apresentem vasculopatias devido a hipertensão, diabetes ou aterosclerose. Em crianças, a paralisia causada por doenças vasculares é menos frequente, enquanto que a paralisia causada por traumas e neoplasias (tumores) ocorrem com maior frequência. Apesar desses fatores contribuintes, a paralisia do sexto nervo é considerada uma disfunção rara; estudos feitos em crianças norte-americanas mostraram que ocorrem 2,5 casos a cada 100 mil habitantes (EHRENHAU, 2018). Mesmo se tratando de uma disfunção rara, quando manifestada, a paralisia do sexto nervo deve ser diagnosticada e tratada precocemente para que a mesma desapareça sem deixar sequelas (CURI; COSTA; BARROSO, 2013). Como já apresentado no **Capítulo 1**, há duas formas principais de avaliar a paralisia do sexto nervo: através do teste de geração de força e da eletro-oculografia.

Figura 7 – Paralisia do sexto nervo no olho esquerdo, com grande limitação de abdução.



Fonte – Adaptado de (CURI; COSTA; BARROSO, 2013).

O teste de geração de força (TGF), do inglês *Force Generation Test* explora o

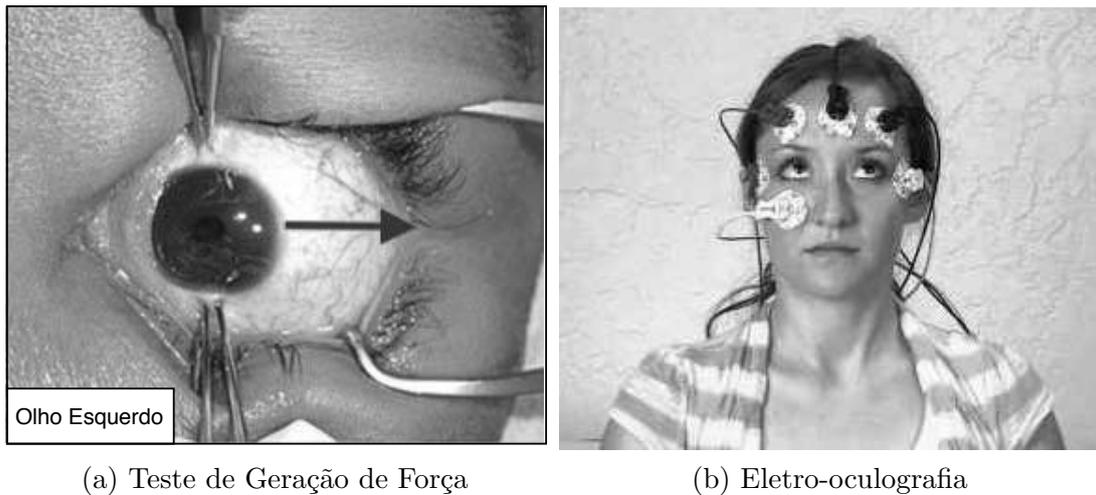
potencial da força muscular, sendo útil para identificar debilidade muscular, seja por lesão muscular direta ou por paralisia do nervo craniano (MELLEN; BILYK, 2018). É um teste simples, mas que requer a cooperação do paciente e não pode ser feito em um paciente sedado. De acordo com Curi, Costa e Barroso (2013), após aplicar anestesia tópica no olho, pede-se que o paciente mova o olho na direção contrária ao músculo a ser examinado, em direção ao lado nasal. Em seguida, fixa-se um fórceps (pinça) ou cotonete no limbo temporal do olho examinado, conforme exemplificado na Figura 8 (a). O fórceps é usado para “segurar” o olho, forçando-o a ficar parado. Depois disso, pede-se que o paciente mova o olho rapidamente em direção ao músculo reto lateral. Neste momento, o oftalmologista analisa a força sentida fórceps. Ao tentar mover o olho quando há algo o segurando, uma contração será sentida caso o músculo esteja com inervação saudável. Um músculo cujo nervo está paralisado não apresentará resistência ao fórceps (MELLEN; BILYK, 2018). Apesar de ser um teste simples, a eficácia do mesmo vai depender da experiência do examinador ao analisar a força gerada. Além disso, esse teste não pode ser realizado em crianças, visto que a cooperação deles é insuficiente, e a anestesia geral não pode ser aplicada (METZ, 1976).

Outra forma de avaliar a presença da paralisia do sexto nervo é através da eletro-oculografia (EOG). A eletro-oculografia é um teste que avalia a função do aparelho vestibular ¹ através da observação dos movimentos oculares, baseando-se nos sinais elétricos que os músculos extra-oculares emanam (MACHADO et al., 2010). A EOG mantém um registro dos movimentos oculares a partir diferença de potencial córneo-retiniana durante a movimentação dos olhos (GANAN; CAOVIILLA; FREITAS, 2010). Este exame é útil para avaliar diversas patologias oculares, como estrabismo, paralisias oculomotoras, dentre outras (BUCCI et al., 2002; LIM; ROSENBAUM; DEMER, 1995; ZIFFER et al., 1992). Para realizar este exame, fixam-se eletrodos ao redor dos olhos do paciente, que registram a atividade muscular a medida que os olhos se movem, conforme mostra a Figura 8 (b). Como resultado, o exame de EOG apresenta a trajetória da movimentação ocular e a velocidade sacádica, brevemente apresentada na Seção 3.2.1.

Apesar de ser um procedimento não invasivo e bastante preciso, a EOG é pouco utilizada em clínicas oftalmológicas, devido ao alto custo de seu equipamento. Com base

¹ O sistema vestibular é responsável pelo senso de movimento e orientação no espaço, contribuindo para manter o equilíbrio do corpo. Por detectar o movimento da cabeça no espaço, o aparelho vestibular estabiliza o eixo visual e mantém a postura da cabeça e do corpo (CULLEN; SADEGHI, 2008).

Figura 8 – Formas tradicionais de avaliar a presença da paralisia do sexto nervo óptico.



Fonte – Adaptado de (MELLEN; BILYK, 2018) e (BIOPAC, 2016), respectivamente.

nisso, esta dissertação propõe uma abordagem não invasiva e de baixo custo para auxiliar no diagnóstico da paralisia do sexto nervo. Através de vídeos gravados em uma câmera simples, registra-se a movimentação ocular de um paciente, a fim de obter a velocidade média e a trajetória da movimentação de cada olho. Partindo do princípio de que há diferença na velocidade de movimentação de um olho saudável e um olho com paralisia, o trabalho a ser apresentado aqui visa mostrar que mesmo usando a velocidade média comum, essa diferença se mantém, fornecendo um limiar base para essa diferença.

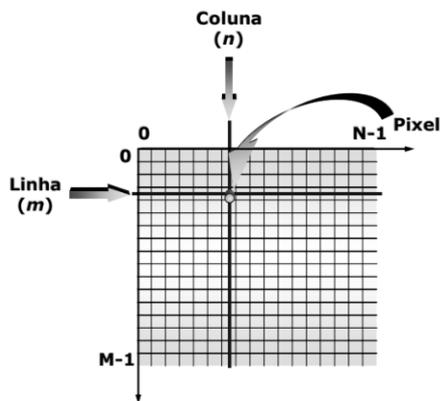
3.4 Processamento Digital de Imagens

De acordo com Castleman (1996), o Processamento Digital de Imagens (PDI) refere-se ao ato de submeter uma representação numérica de um objeto - uma imagem digital - a uma série de operações, a fim de obter algum tipo de resultado. Gonzalez e Woods (2010) ressaltam que processos cuja entrada e saída são imagens, além da extração de atributos de imagens e reconhecimento de objetos também fazem parte da área de PDI. A diminuição do preço dos computadores, seu crescente poder de processamento, bem como o aumento da disponibilidade de equipamentos de digitalização contribuíram para a popularização do Processamento Digital de Imagens, que tem aplicações nos campos mais diversos.

Uma imagem digital pode ser definida como uma função bidimensional $f(x, y)$, onde x e y são coordenadas espaciais, e o valor de $f(x, y)$ em cada ponto do plano

corresponde à intensidade de luz naquele ponto (GONZALEZ; WOODS, 2010). Cada ponto presente na grade bidimensional que representa uma imagem digital é chamado de *pixel*, do inglês *picture element* (QUEIROZ; GOMES, 2006). Os *Pixels* são elementos de uma imagem digital, como mostra a Figura 9.

Figura 9 – Exemplo de uma imagem digital bidimensional.



Fonte – Adaptado de (QUEIROZ; GOMES, 2006).

Em caso de imagens monocromáticas, a intensidade de luz é chamada de escala de cinza. Esta escala de luminância apresenta valores num intervalo definido $[L_{min}, L_{max}]$, onde L_{min} corresponde à cor preta e L_{max} à cor branca. A Figura 10 apresenta uma imagem em escala de cinza e seus valores de *pixel* correspondentes.

Figura 10 – Exemplo de uma imagem em escala de cinza e sua representação matricial com os valores de cada *pixel*. Nesta escala de cor, os valores de intensidade variam de $[0, 255]$.

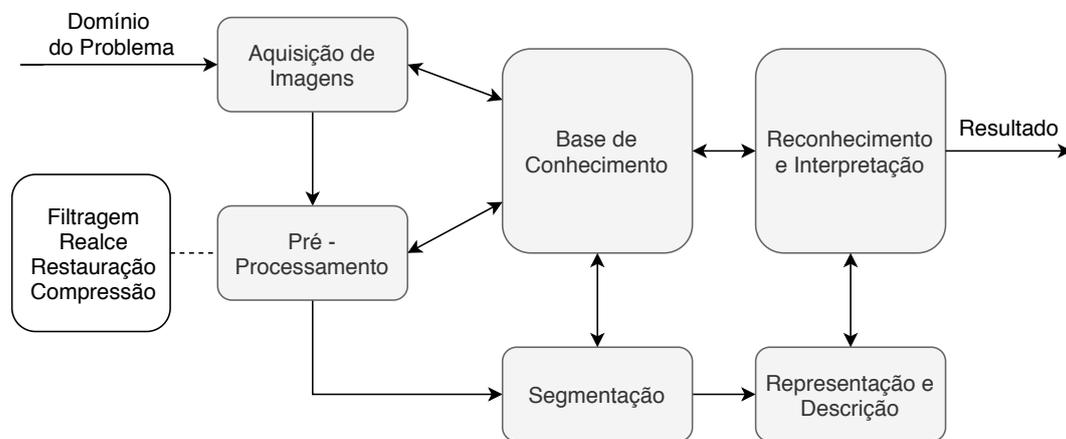
Imagem em escala de cinza	Representação Matricial
	0 16 32 48 64 80 96 112 128 144 160 176 192 208 224 240
	1 17 33 49 65 81 97 113 129 145 161 177 193 209 225 241
	2 18 34 50 66 82 98 114 130 146 162 178 194 210 226 242
	3 19 35 51 67 83 99 115 131 147 163 179 195 211 227 243
	4 20 36 52 68 84 100 116 132 148 164 180 196 212 228 244
	5 21 37 53 69 85 101 117 133 149 165 181 197 213 229 245
	6 22 38 54 70 86 102 118 134 150 166 182 198 214 230 246
	7 23 39 55 71 87 103 119 135 151 167 183 199 215 231 247
	8 24 40 56 72 88 104 120 136 152 168 184 200 216 232 248
	9 25 41 57 73 89 105 121 137 153 169 185 201 217 233 249
	10 26 42 58 74 90 106 122 138 154 170 186 202 218 234 250
	11 27 43 59 75 91 107 123 139 155 171 187 203 219 235 251
	12 28 44 60 76 92 108 124 140 156 172 188 204 220 236 252
	13 29 45 61 77 93 109 125 141 157 173 189 205 221 237 253
	14 30 46 62 78 94 110 126 142 158 174 190 206 222 238 254
	15 31 47 63 79 95 111 127 143 159 175 191 207 223 239 255

Fonte – Adaptado de (VUČKOVIĆ, 2008).

Gonzalez e Woods (2010) dividem o Processamento Digital de Imagens em etapas ou passos fundamentais, de forma que aplicações de PDI devem incluir alguns desses sub-

processos. O primeiro passo é definição ou estudo do **Domínio do Problema**, seguido da **Aquisição de Imagens** relacionadas ao problema, o que geralmente envolve algum tipo de **Pré-Processamento**. Outra etapa pode ser a **Segmentação**, considerada uma das tarefas mais difíceis em PDI. A partir do resultado da segmentação, em geral, parte-se para o estágio de **Representação e Descrição**, e com os resultados obtidos nesta etapa, pode-se fazer um processo de **Reconhecimento**, culminando em um **Resultado** final. Boa parte desses módulos precisam interagir com uma **Base de Conhecimento** - um banco de dados de informações codificadas sobre o problema. A Figura 11 mostra como cada uma dessas etapas se relacionam.

Figura 11 – Principais etapas do Processamento Digital de Imagens.



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

Na metodologia deste trabalho, os módulos de Pré-Processamento e Segmentação foram aplicados. As subseções a seguir explicam que técnicas foram usadas em cada etapa.

3.4.1 Convolução

De acordo com Gonzalez e Woods (2010), a convolução de uma função de entrada 2D discreta $f[x, y]$ com uma função 2D discreta $h[x, y]$ de tamanho $m \times n$, é definida como:

$$s(x, y) = h[x, y] * f[x, y] \equiv \sum_{s=-a}^a \sum_{t=-b}^b h(s, t) f(x - s, y - t) \quad (3.1)$$

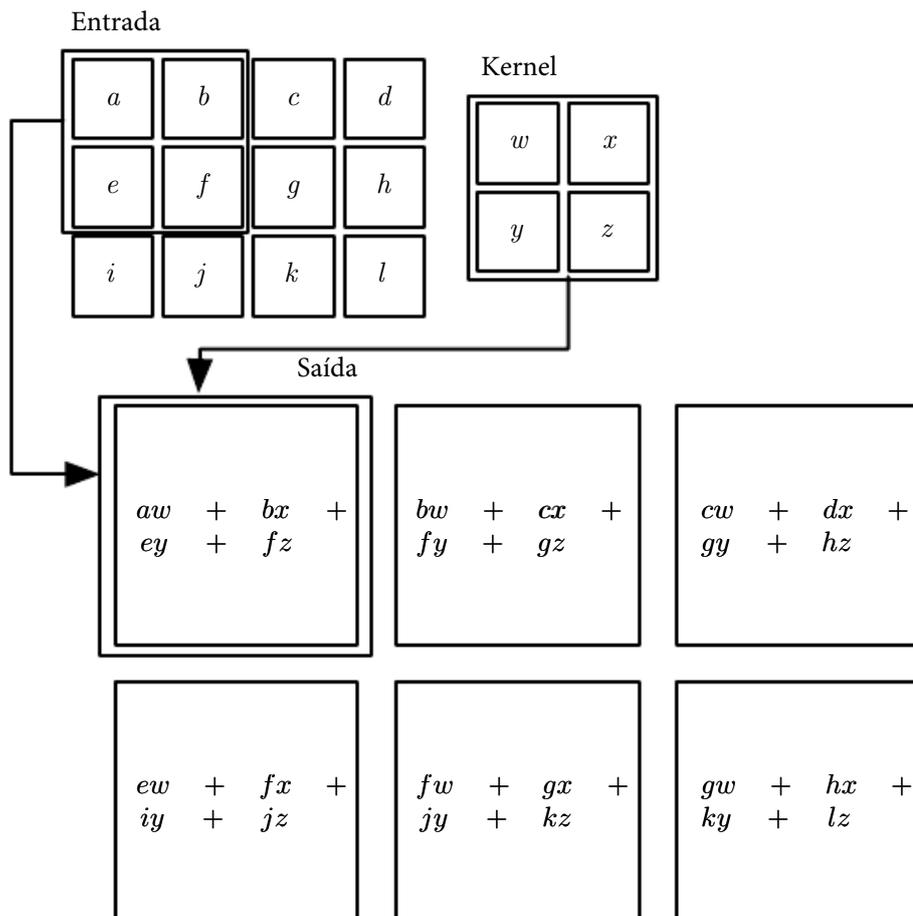
onde, $a = \frac{m-1}{2}$ e $b = \frac{n-1}{2}$.

No contexto de Processamento Digital de Imagens, $f[x, y]$ pode ser vista como uma matriz que representa uma imagem e $h[x, y]$, como um *kernel*. *Kernel* (também

chamado de filtro, máscara ou *template*) é um arranjo matricial de dimensões $m \times n$ cujos valores são definidos como fatores de ponderação (pesos) a serem aplicados sobre *pixels* da imagem (QUEIROZ; GOMES, 2006). De forma simplificada, a convolução é vista como um processo de filtragem da imagem f pelo *kernel* h (EASTON, 2010).

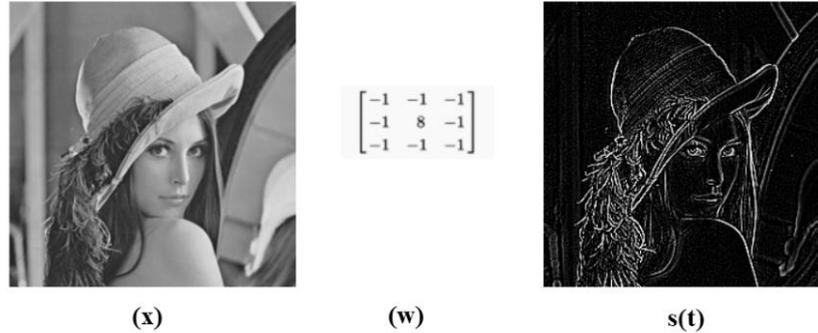
A operação de convolução preserva a relação entre *pixels*, já que as transformações são aplicadas progressivamente sobre porções da imagem, coluna a coluna, linha a linha (GOODFELLOW; BENGIO; COURVILLE, 2016). Em qualquer ponto (x, y) da imagem de entrada f , a resposta s é a soma dos produtos dos coeficientes do *kernel* com os *pixels* da imagem englobados pelo filtro, conforme ilustra a Figura 12. A Figura 13 mostra o resultado de uma convolução sobre uma imagem. A convolução é a base de várias operações em Processamento Digital de Imagens, por isso, seu funcionamento é necessário para o entendimento de outras técnicas.

Figura 12 – Passo a passo de um convolução em 2D com um *kernel* de tamanho 2×2 .



Fonte – Adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016).

Figura 13 – Resultado da operação convolução da imagem (x) pela máscara (w) de tamanho 3×3 .



3.4.2 Suavização Através de Filtragem Gaussiana

O processo de suavização de imagens é feito para remover pequenos detalhes, “borrar” a imagem ou atenuar ruídos. Na metodologia deste trabalho, usou-se suavização para degradar a imagem, removendo detalhes que poderiam atrapalhar nas outras etapas de processamento, que será melhor discutido na Seção 3.4.5. A suavização dá-se através de filtros/*kernels* específicos, aplicando a operação de convolução do filtro sob a imagem original, conforme apresentado na Seção 3.4.1. Dentre as etapas em Processamento Digital de Imagens mencionadas na Seção 3.4, a suavização faz parte do Pré-Processamento.

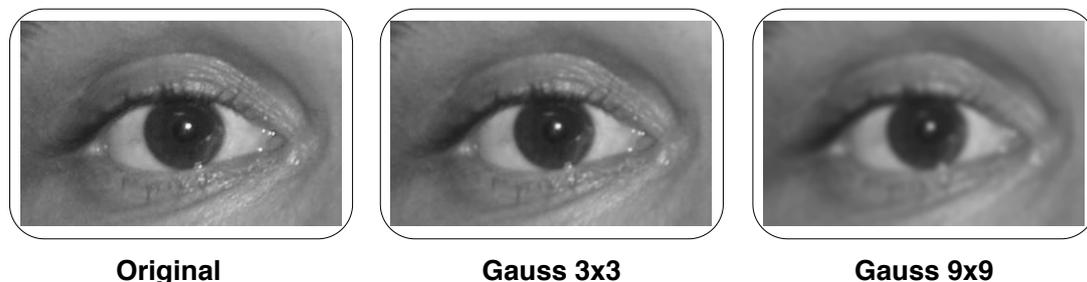
Neste trabalho, a suavização da imagem deu-se através da filtragem Gaussiana 2D. Ao aplicar este filtro, o valor resultante de cada *pixel* corresponde à média ponderada dos *pixels* vizinhos. A quantidade de vizinhos é definida pelo tamanho da matriz do *kernel*. O *pixel* que fica no centro do *kernel* recebe o maior peso, enquanto que os *pixels* vizinhos recebem pesos menores, à medida que se distanciam do *pixel* central (BELLIDO; PIETRO; SANNINO, 2016). Este filtro possui um outro parâmetro que afeta seu resultado, o desvio padrão, que define a intensidade da aplicação do filtro e os pesos do *kernel* de filtragem. O filtro de suavização Gaussiana em 2D assume a seguinte forma:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (3.2)$$

onde σ^2 é o desvio padrão da distribuição gaussiana, ou a variância, e as coordenadas x e y são inteiros que representam a distância para o centro do *kernel* horizontalmente e verticalmente (BELLIDO; PIETRO; SANNINO, 2016). É importante lembrar que quanto maior a dimensão do *kernel*, mais intensa será a suavização na imagem filtrada. A Figura

14 mostra a aplicação da filtragem Gaussiana usando um *kernel* de tamanho 3×3 e um *kernel* de tamanho 9×9 . Percebe-se que as bordas da imagem em que foi aplicado o *kernel* 9×9 estão bem menos definidas, dando um aspecto desfocado à imagem.

Figura 14 – Aplicação da Filtragem Gaussiana com diferentes tamanhos de *kernel*. Em ambos os casos, o valor do σ aplicado é 1.



3.4.3 Cálculo do Gradiente

Na seção anterior, apresentou-se um filtro empregado na suavização de imagens, entretanto, há filtros usados para acentuar regiões de interesse na imagem, como bordas. Bordas são regiões onde ocorrem alta variação dos níveis de cinza. Um dos operadores que faz aguçamento de bordas é o gradiente da imagem.

De acordo com Easton (2010), o gradiente de uma função 2D f contínua em $[x, y]$ é o vetor composto das derivadas parciais em x e em y , descrito pela Equação 3.3:

$$\nabla f[x, y] = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right] \quad (3.3)$$

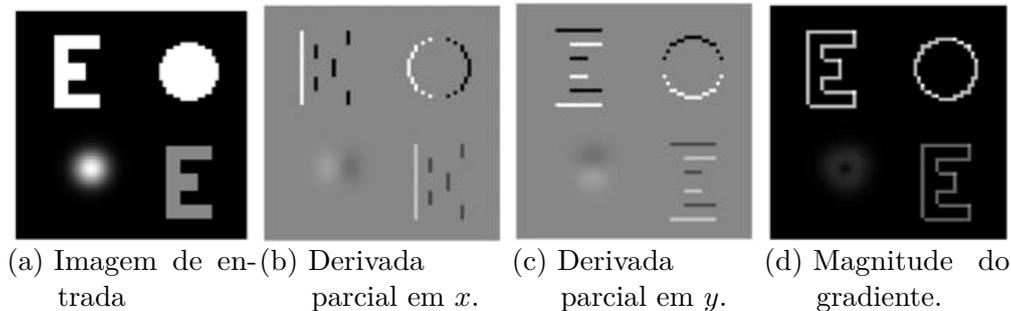
onde $f[x, y]$ é uma imagem monocromática que atribui um valor em nível de cinza para cada coordenada (x, y) . Este processo calcula um vetor para cada coordenada (x, y) em que seus componentes cartesianos são $\frac{\partial f}{\partial x}$ e $\frac{\partial f}{\partial y}$. O vetor resultante aponta para onde há a maior variação em $f[x, y]$, o que ocorre onde há bordas. Também pode-se calcular a magnitude do gradiente da imagem, que é um forma de representar a variação de $f[x, y]$ por unidade de distância no sentido de ∇ (QUEIROZ; GOMES, 2006). O cálculo da magnitude do gradiente é apresentado na Equação 3.4:

$$|\nabla f[x, y]| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (3.4)$$

Os maiores valores de magnitude do gradiente encontram-se nos *pixels* onde há um “salto” de intensidade no nível de cinza. Portanto, o gradiente da imagem pode ser

usado como um operador de detecção de bordas. A magnitude do gradiente também apresenta a propriedade de isotropia, ou seja, são invariantes à rotação, o que possibilita identificar bordas independentemente da orientação, como mostra a Figura 15 (QUEIROZ; GOMES, 2006).

Figura 15 – Exemplo da aplicação do operador gradiente em uma imagem.



Fonte – Adaptado de (EASTON, 2010).

Neste trabalho, o cálculo do gradiente e sua magnitude são etapas fundamentais da técnica a ser explicada na Seção 3.4.5. O cálculo do gradiente também está incluso na etapa de Pré-Processamento, ilustrada Seção 3.4.

3.4.4 Cálculo da Derivada de Segunda Ordem

De acordo com Gonzalez e Woods (2010), as derivadas de uma função digital são definidas por meio de diferenças. Em imagens, a diferenciação realça bordas e discontinuidades, enquanto atenua áreas de menor variação. O processo de diferenciação nas áreas de descontinuidade muda de acordo com a ordem da derivada; na derivada de primeira ordem na posição x , subtrai-se o valor da função em x do valor da função em $x + 1$. Já na derivada de segunda ordem, usa-se tanto o valor da função em $x + 1$ quanto em $x - 1$, conforme mostra a Equação 3.5:

$$\frac{\partial^2 f}{\partial x^2} = f(x + 1) + f(x - 1) - 2f(x) \quad (3.5)$$

O sinal da segunda derivada muda no início e no final de um degrau ou rampa (áreas de mudança brusca na intensidade do nível de cinza) (EASTON, 2010). Na prática, significa que ao calcular a segunda derivada de uma imagem, a imagem resultante apresenta bordas mais finas, com espessura de um *pixel*. Dentre as vantagens da derivada de segunda

ordem, destacam-se a sua facilidade de implementação, e sua capacidade de realçar muito mais os detalhes finos da imagem, sendo indicada para o aguçamento de imagens.

Gonzalez e Woods (2010) ressaltam que a forma discreta mais simples de representar uma derivada de segunda ordem é através do operador Laplaciano. Para uma função 2D $f(x, y)$, o Laplaciano pode ser definido como:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (3.6)$$

onde $\frac{\partial^2 f}{\partial x^2}$ corresponde a:

$$\frac{\partial^2 f}{\partial x^2} = f(x + 1, y) + f(x - 1, y) - 2f(x, y) \quad (3.7)$$

e $\frac{\partial^2 f}{\partial y^2}$ corresponde a:

$$\frac{\partial^2 f}{\partial y^2} = f(x, y + 1) + f(x, y - 1) - 2f(x, y) \quad (3.8)$$

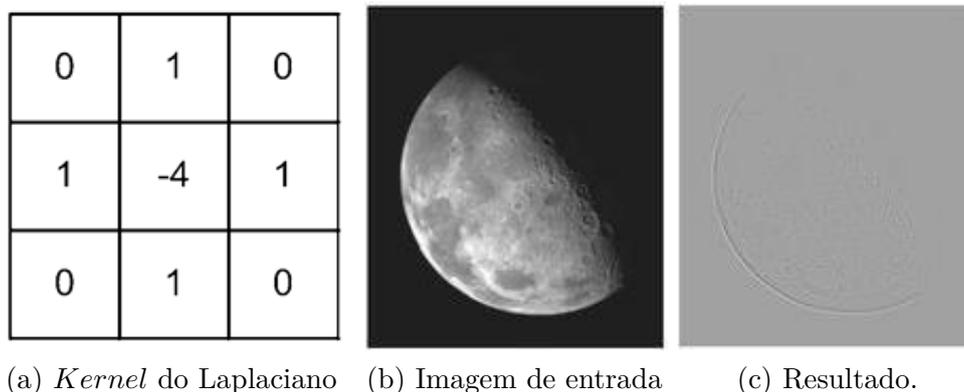
Uma aproximação numérica da Equação 3.6, resultaria na máscara de filtragem apresentada na Figura 16 (a). A Figura 16 (c) mostra o resultado da aplicação do Laplaciano com o referido *kernel*. Nota-se que a sua utilização realçou as descontinuidades de intensidade, enquanto atenuou regiões de menor variação de intensidade. Como resultado, a imagem final apresenta linhas de borda bem finas, enquanto que os detalhes da imagem original estão quase imperceptíveis. Neste trabalho, o Laplaciano é umas das etapas fundamentais da técnica apresentada na Seção 3.4.5, agindo como um Pré-Processamento da imagem usada.

3.4.5 Curvatura de Isophotes

O centro do olho é um bom indicativo de atenção visual, sendo usado em diferentes aplicações de visão computacional, como alinhamento de face, reconhecimento facial, interação entre humano e computador, dentre outros (VALENTI; GEVERS, 2012).

Tendo em vista que os olhos são caracterizados por padrões de brilho simétricos radialmente, Valenti e Gevers (2008) propuseram uma nova abordagem para a localização do centro do olho baseada na observação desses padrões, chamada de Curvatura de *Isophotes*. *Isophotes* ou isofotos são curvas que conectam *pixels* na imagem que possuem a mesma intensidade (MARCO et al., 2015). Já que as curvas de *Isophotes* não se cruzam,

Figura 16 – Exemplo da aplicação do operador Laplaciano de *kernel* (a) sob a imagem (b).



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

uma imagem pode ser totalmente descrita por elas. Além disso, foi demonstrado por Lichtenauer, Hendriks e Reinders (2005) que a forma dos *Isophotes* é invariante à rotação e a mudanças de iluminação. Devido à essas características, o uso das curvas de *Isophotes* nas tarefas de detecção de objetos e segmentação de imagens tem obtido sucesso (VALENTI; GEVERS, 2008).

De forma resumida, a curvatura de *Isophotes* pode ser calculada como mostra a Equação 3.9:

$$k = -\frac{L_{vv}}{L_w} \quad (3.9)$$

onde, L_{vv} pode ser descrita como:

$$L_{vv} = L_y^2 L_{xx} - 2L_x L_{xy} L_y + L_x^2 L_{yy} \quad (3.10)$$

e L_w representa:

$$(L_x^2 + L_y^2)^{\frac{3}{2}} \quad (3.11)$$

$L(x, y)$ é a função de luminância, que nesse contexto, é a própria imagem. L_x e L_y são as derivadas parciais da imagem em x e em y , respectivamente, enquanto L_{xx} se refere à derivada parcial de x em função de y , e L_{yy} , à derivada parcial de y em função de x . L_{xy} representa o gradiente da imagem, conforme explicado na Seção 3.4.3. L_x^2 e L_y^2 são as derivadas de segunda ordem com relação à x e y , respectivamente, previamente apresentado na Seção 3.4.4.

O K resultante é um mapa de curvaturas, apresentado na Figura 17 (a). A fim de encontrar os centro do(s) círculo(s) presentes no mapa de curvatura, é preciso calcular os vetores de deslocamento que representam a posição estimada dos centros, que podem ser mapeadas em um acumulador. Os vetores de deslocamento podem ser definidos como $\{L_x, L_y\}$ se refere ao gradiente):

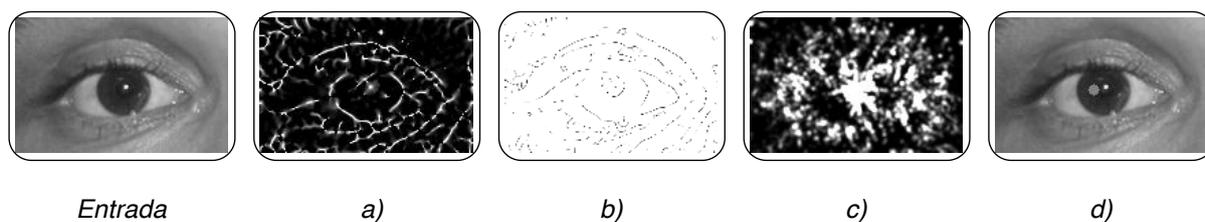
$$D(x, y) = -\frac{\{L_x, L_y\}(L_x^2 + L_y^2)}{L_y^2 L_{xx} - 2L_x L_{xy} L_y + L_x^2 L_{yy}} \quad (3.12)$$

Em seguida, é necessário calcular a curvatura da forma, que não deve ser confundida com K , que representa o mapa de curvaturas *Isophotes*. A curvatura da forma é dada por:

$$Curvatura = \sqrt{L_{xx}^2 + 2L_{xy}^2 + L_{yy}^2} \quad (3.13)$$

Após obter os vetores de deslocamento e a curvatura da forma, é preciso limitar as curvaturas que se deseja, definindo um raio mínimo e um raio máximo, a fim de descartar *pixels* que possuem curvatura muito baixa (*pixels* pertencentes a linhas) ou curvatura muito alta (*pixels* pertencentes a círculos muito pequenos). Após isso, constrói-se um mapa de centros, onde verifica-se se para cada pixel, o vetor de deslocamento correspondente possui valor entre os limites de raio mínimo e máximo definido, caso possua, o *pixel* naquela posição receberá o valor da curvatura na respectiva posição. A Figura 17 (c) mostra o resultado de um mapa de centros. Após isso, a fim de saber o centro com o maior acumulador, procura-se a posição do mapa de centros com o maior valor - este será o centro do círculo, apresentado na Figura 17 (d). O acumulador funciona como um mecanismo de votação de centros, que aumenta e pondera votos importantes para reforçar as estimativas da posição central.

Figura 17 – Etapas do *Isophotes*. (a) Mapa de Curvaturas *Isophotes*. (b) Vetores de deslocamento. (c) Mapa de centros. (d) Centro de maior valor (ponto vermelho).

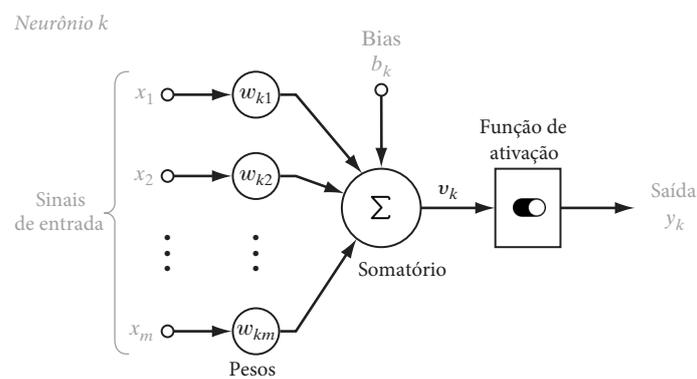


O método de *Isophotes* é usado para inferir o centro de padrões semi-circulares em imagens de baixa resolução. Para isso, antes de aplicar o método, é preciso borrar a imagem original, o que pode ser feito através da suavização por filtragem Gaussiana, apresentada na Seção 3.4.2. O método de *Isophotes* foi escolhido para ser uma das abordagens testadas por não ser muito parametrizado, e por apresentar resultados satisfatórios na detecção do centro dos olhos, atingindo acurácia de aproximadamente 98% (VALENTI; GEVERS, 2008). Com relação às etapas em PDI mencionados na Seção 3.4, o método de *Isophotes* pode ser classificado como um estágio inicial da fase de Segmentação.

3.5 Redes Neurais

De acordo com Haykin (2004), uma rede neural é uma máquina projetada para representar a forma com que o cérebro executa uma tarefa específica. Uma rede neural é composta por uma série de unidades simples de processamento, que coletam e armazenam o conhecimento obtido a partir de um conjunto de dados de entrada. A essas unidades de processamento dá-se o nome de neurônios artificiais. Na rede neural, os neurônios são interligados por meio de conexões chamadas de sinapses. As sinapses entre os neurônios são fundamentais para o processo de aprendizado no qual a rede adquire conhecimento.

Figura 18 – Estrutura de funcionamento de um neurônio artificial.



Fonte – Adaptado de (HAYKIN, 2010).

A Figura 18 ilustra a estrutura de um neurônio artificial, que de acordo com Haykin (2010), apresenta os seguintes componentes básicos:

- Um conjunto de sinapses, em que cada uma é caracterizada por um peso, que representa a “força” da conexão. No neurônio k , um sinal de entrada x_i é multiplicado

pelo peso sináptico w_{ki} , onde i varia entre $[1, m]$;

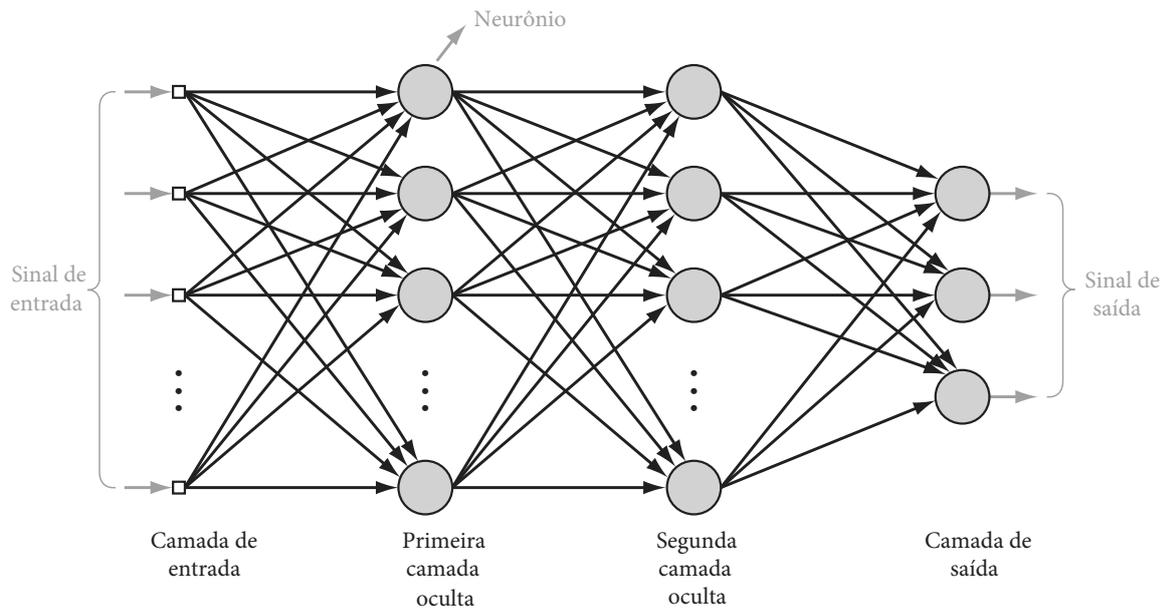
- Um somatório Σ que realiza a soma ponderada dos sinais de entrada x_i de acordo com os respectivos pesos sinápticos w_{ki} ;
- Uma variável de *bias* b_k que pode aumentar ou diminuir o valor do sinal resultante do somatório Σ ;
- Uma função de ativação que limita a amplitude da saída do neurônio. Esta função mapeia a saída da soma v_k na saída final do neurônio, representada por y_k .

Dentre as vantagens das redes neurais, lista-se a capacidade de aprender e generalizar, adaptando os pesos sinápticos de acordo com mudanças nos dados. Essas capacidades de processamento possibilitam às redes neurais encontrar boas soluções aproximadas para problemas complexos, sendo aplicadas em tarefas como classificação, aproximação de funções, otimização, mineração de dados etc.

3.5.1 Redes *FeedForward* de Múltiplas Camadas

Uma importante classe de redes neurais são as redes *feedforward* de múltiplas camadas, também conhecidas como redes MLP, do inglês *Multilayer Perceptron*. Redes MLP são redes neurais que apresentam uma ou mais camadas de neurônios ocultos, que não fazem parte da entrada ou saída da rede (HAYKIN, 2004). Cada camada em uma rede MLP tem funções específicas; a camada de entrada, em geral, não processa os sinais de entrada, mas apenas os recebe e os distribui para os neurônios na camada oculta seguinte. As camadas ocultas são usadas para extrair características específicas dos dados de entrada, o que capacita a rede a aprender tarefas complexas. A camada de saída, por sua vez, recebe o sinal da camada oculta e produz o sinal de saída de toda a rede. Os neurônios presentes em cada camada são completamente conectados aos neurônios da camada seguinte, e os sinais são passados de uma camada para outra através do *feedforward*. O termo *feedforward* implica que os sinais são passados adiante, mas que não há ciclos entre as camadas da rede (GOODFELLOW; BENGIO; COURVILLE, 2016). Portanto, neurônios de uma camada n se conectam somente com os neurônios da camada $n + 1$, não havendo retroalimentação, já que a conexão não é bidirecional. A Figura 19 exemplifica como é a estrutura de uma rede de múltiplas camadas.

Figura 19 – Arquitetura de uma rede MLP com duas camadas ocultas.



Fonte – Adaptado de (HAYKIN, 2010).

As redes MLP usam a regra de aprendizado por correção de erro, através do algoritmo de retropropagação de erro, do inglês *backpropagation*. O processo de aprendizagem por retropropagação de erro consiste em dois passos: um passo para a frente (*feedforward*, já explicado acima) e um passo para trás. Durante o *feedforward*, os pesos sinápticos são fixos, enquanto que durante a retropropagação, os pesos são todos ajustados de acordo com uma regra de correção de erro, baseada na saída correta esperada (HAYKIN, 2004). De forma resumida, as redes MLP aprendem com os erros, reunindo conhecimento a partir da experiência. O processo de aprendizado dá-se pela repetição das etapas de propagação e retropropagação de erro. A esta repetição por várias iterações dá-se o nome de treinamento da rede, e cada iteração também é chamada de época.

O algoritmo de retropropagação inicia-se com o cálculo do erro de saída. Após calcular o resultado do neurônio k na camada de saída na iteração p ($y_k(p)$), calcula-se um sinal de erro ($e_k(p)$) para este neurônio na iteração em questão, subtraindo o resultado recebido pelo resultado esperado naquela iteração, chamado de $y_{d,k}(p)$, conforme mostra a Equação 3.14:

$$e_k(p) = y_{d,k}(p) - y_k(p) \quad (3.14)$$

Este sinal de erro é propagado para trás através da rede, indo em direção contrária

ao do *feedforward*, para que haja um ajuste dos pesos. Os pesos sinápticos de cada neurônio são ajustados de acordo com o respectivo sinal de erro, a fim de que a resposta real da rede se aproxime mais da resposta esperada. O ajuste de pesos é feito de forma diferente na camada de saída e nas camadas ocultas. Antes de explicar como se calcula o erro em cada camada, destaca-se que o símbolo k se refere à camada de saída, j às camadas ocultas e i à camada de entrada. Na camada de saída, calcula-se o gradiente do erro no neurônio k na iteração p . Além de usar os valores de $e_k(p)$ e $y_k(p)$, calcula-se a soma ponderada das entradas do neurônio k na iteração p , representado por $X_k(p)$, conforme mostra a Equação 3.15. O valor $\delta_k(p)$ é o gradiente de erro para os neurônios na camada de saída. Com esse valor, ajusta-se os pesos a partir da camada de saída em direção a camada de entrada.

$$\delta_k(p) = \frac{\partial y_k(p)}{\partial X_k(p)} \cdot e_k(p) \quad (3.15)$$

Usa-se o valor $\delta_k(p)$ para calcular a correção dos pesos na iteração p e atualizar o valor dos pesos a serem usados na iteração $p + 1$, como exemplifica a Equação 3.16. A variável w_{jk} representa o peso da sinapse entre neurônio j da camada oculta, e o neurônio k da camada de saída, enquanto que α representa a taxa de aprendizagem e y_j é a saída do neurônio j na camada escondida.

$$w_{jk}(p + 1) = w_{jk}(p) + \alpha \cdot y_j(p) \cdot \delta_k(p) \quad (3.16)$$

Após atualizar os pesos das sinapses que incidem nos neurônios da camada de saída, atualiza-se os pesos dos neurônios nas camadas ocultas. Para isso, calcula-se o gradiente do erro dos neurônios na camada escondida, conforme mostra a Equação 3.16. O gradiente do erro do neurônio j da camada oculta na iteração p é apresentado por $\delta_j(p)$, onde y_j é a saída do neurônio j na camada oculta. As variáveis $\delta_k(p)$ e w_{jk} já foram calculadas nas Equações 3.15 e 3.16.

$$\delta_j(p) = y_j(p) \cdot [1 - y_j(p)] \cdot \sum_{k=1}^l \delta_k(p) w_{jk}(p) \quad (3.17)$$

A partir desse gradiente, atualiza-se os pesos das sinapses que incidem nos neurônios da camada oculta, através da Equação 3.18. O peso da sinapse entre o neurônio i da camada de entrada e o neurônio j da camada oculta é representado por w_{ij} . Seu valor é

obtido de forma análoga aos pesos da camada de saída. Entretanto, na camada oculta, usa-se o termo x_i , que representa a soma ponderada das entradas do neurônio j , vinda dos neurônios da camada de entrada, que pode ser calculada da seguinte forma:

$$w_{ij}(p+1) = w_{ij}(p) + \alpha \cdot x_i(p) \cdot \delta_j(p) \quad (3.18)$$

Após atualizar os pesos da camada oculta, repete-se o processo de *feedforward* seguido pelo *backpropagation* até que o número máximo de iterações seja atingido.

3.5.2 Aprendizagem Profunda

Técnicas de aprendizado de máquina deram aos computadores a capacidade de adquirirem seu próprio conhecimento, ao aprender com os dados e melhorar automaticamente a partir da experiência (LIU et al., 2017). Entretanto, alguns conceitos tem natureza tão complexa, que para um computador os aprender, é preciso construí-los a partir de uma série de conceitos mais simples. De acordo com Goodfellow, Bengio e Courville (2016), se construirmos um gráfico mostrando quantos conceitos básicos são necessários para se entender um conceito complexo, o gráfico será profundo, com muitas camadas. À essa abordagem do aprendizado de máquina baseada em muitas camadas, dá-se o nome de aprendizagem profunda. O aprendizado hierárquico é a chave da aprendizagem profunda e é o que a diferencia dos algoritmos tradicionais de aprendizado de máquina.

A ideia por trás da aprendizagem profunda vêm do potencial das camadas ocultas. Como visto na Seção 3.5.1, camadas ocultas podem extrair características complexas dos dados. Portanto, supõe-se que adicionar muitas camadas ocultas à arquitetura de uma rede neural daria à ela a capacidade de aprender padrões cada vez mais complexos. Por usar muitas camadas de processamento, a aprendizagem profunda tem mais chances de aprender a representação correta para um conjunto de dados. Através do algoritmo de retropropagação de erro, a aprendizagem profunda pode descobrir estruturas complexas em grandes conjuntos de dados (LECUN; BENGIO; HINTON, 2015). Além disso, redes com maior profundidade podem executar mais instruções em sequência. Entretanto, não é preciso que uma rede neural tenha muitas camadas ocultas para ser considerada profunda, redes com mais de três camadas (incluindo entrada e saída) já se qualificam como aprendizagem profunda.

Devido ao seu poder de processamento, a aprendizagem profunda têm proporcio-

nado grandes avanços em tarefas complexas que envolvem imagens e vídeo, através de uma classe específica de redes neurais profundas, as Redes Neurais Convolucionais - a serem apresentadas na Seção a seguir.

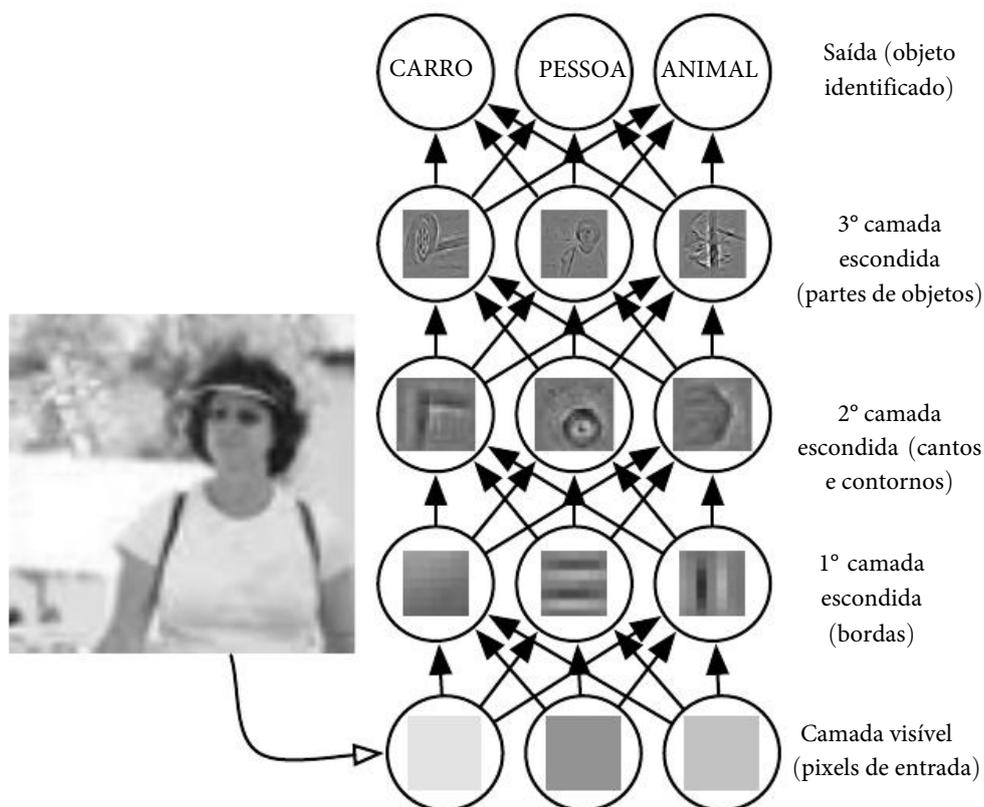
3.5.3 Redes Neurais Convolucionais

De acordo com Goodfellow, Bengio e Courville (2016), as redes neurais convolucionais (CNNs) são tipos especiais de redes neurais que foram projetadas para processar imagens ou dados com topologia de grade. Em pelo menos uma de suas camadas, as CNNs realizam a operação de convolução, apresentada na Seção 3.4.1, em vez da multiplicação de matrizes convencional. Os filtros de convolução de uma CNN tentam identificar características mais simples primeiro, usando-as para identificar características mais complexas gradualmente, por várias camadas.

A primeira camada de convolução de uma CNN tradicional usa um conjunto de filtros para identificar características de baixo nível na imagem de entrada. Em seguida, as características identificadas são agrupadas (pelas camadas de *pooling*) e passadas como entrada para a camada de convolução seguinte, que por sua vez, usa outro conjunto de filtros para identificar características mais complexas que as identificadas anteriormente. Esse processo se repete por várias camadas, nas quais cada camada de convolução usa as entradas da camada anterior para identificar características de nível superior ao da camada anterior. Por fim, a saída da última camada de convolução é passada para um conjunto de camadas totalmente conectadas para a classificação final da imagem (GOODFELLOW; BENGIO; COURVILLE, 2016). A Figura 20 ilustra este processo, onde uma imagem de entrada é representada em uma série de mapeamentos simples, descritos por camadas diferentes do modelo. A camada de entrada recebe uma coleção de valores de *pixel* que representam a imagem, que são passados para uma série de camadas escondidas que extraem características cada vez mais abstratas da imagem. A partir dos *pixels*, bordas são identificadas, que são usadas para identificar cantos e contornos, que por sua vez, ajudam a detectar partes inteiras de objetos. Por fim, a descrição da imagem em termos das partes contidas nela é usada na camada de saída para a classificação dos objetos presentes na imagem (GOODFELLOW; BENGIO; COURVILLE, 2016).

Neste trabalho, usou-se uma arquitetura de rede neural convolucional para identificar olhos em vídeos: a rede YOLO em sua terceira versão. A arquitetura em questão foi

Figura 20 – Exemplo das etapas de processamento em uma CNN.



Fonte – Adaptado de (GOODFELLOW; BENGIO; COURVILLE, 2016).

escolhida por funcionar bem em vídeos e por apresentar alta confiabilidade na detecção de objetos. A CNN YOLO será apresentada em detalhes na seção a seguir.

3.5.4 YOLO - *You Only Look Once*

YOLO, sigla para *You Only Look Once* (Você só olha uma vez) é uma rede neural convolucional para detecção de objetos em tempo real, com uma abordagem atual para detecção proposta por Redmon et al. (2015). A tarefa de detecção de objetos consiste em determinar o local na imagem em que certos objetos estão presentes, além de classificar esses objetos. Os modelos usuais de CNNs para detecção de objetos primeiro geram regiões candidatas (*bounding boxes*) a terem objetos, e depois executam um classificador nessas regiões. A YOLO, por sua vez, não vê a detecção de objetos como um problema de classificação, mas como de regressão. Através de um modelo unificado, a YOLO prevê a região em que um objeto está, enquanto informa a qual classe o objeto detectado pertence. Isso faz com que a rede seja extremamente rápida, processando até 45 quadros por segundo

(FPS) em sua versão convencional e até 155 FPS em sua versão reduzida.

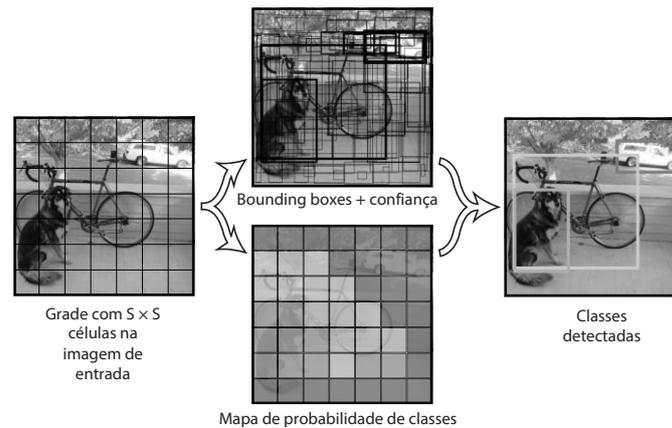
A YOLO inicia seu processamento dividindo a imagem de entrada em uma grade de células $S \times S$. Cada célula prevê um número B de *bounding boxes* e a probabilidade (ou confiança) dessas regiões conterem um objeto. Se nenhum objeto existir nessa célula, o valor da confiança deve ser zero. Mas se existir algum objeto, o valor de confiança também deve refletir o quanto a *bounding box* encontrada se encaixa na região de interesse. Para isso, calcula-se a confiança através da equação $Pr(Object) \times IOU$. $Pr(Object)$ é a probabilidade de existir um objeto naquela célula, e IOU é a interseção sobre a união da *bounding box* predita com a *bounding box* real (TAN; STEINBACH; KUMAR, 2016). Se o valor de confiança for alto (acima de um limiar estipulado), a rede irá então prever qual tipo de objeto está dentro daquela célula, criando um mapa de probabilidade de classes.

Cada célula prevê C probabilidades condicionais de classe, sendo calculada por $Pr(Classe_i|Objeto)$, onde i varia de 1 a quantidade de classes existentes. Por fim, as probabilidades condicionais de classe são multiplicadas pelo valor de confiança individual de cada *bounding box* para se obter os valores de confiança para uma classe específica em uma *bounding box*, conforme mostra a Equação 3.19. Esse valor codifica tanto a probabilidade de tal classe aparecer naquela região, quanto o quão bem a região prevista se encaixa no objeto. A Figura 21 ilustra cada uma das etapas de processamento da YOLO.

$$Pr(Classe_i \times IOU) = Pr(Classe_i|Objeto) \times Pr(Objeto) \times IOU \quad (3.19)$$

Nesta dissertação, usou-se a YOLO em sua terceira versão, YOLOv3, cuja estrutura será explicada na seção a seguir. Esta CNN foi escolhida por apresentar alta precisão e por ser rápida, se mostrando apropriada para ser aplicada em vídeos. Dentre as principais mudanças dessa nova versão estão a classificação *multilabel* e predição da saída da rede em três escalas diferentes. A classificação *multilabel* implica que uma mesma *bounding box* pode conter mais de um objeto. Cada *bounding box* prevê as classes que podem estar presentes nelas através de uma classificação por vários rótulos, permitindo que uma mesma região da imagem tenha classes/objetos sobrepostos. Para isso, trocou-se o *softmax* presente nas versões antigas da rede por classificadores logísticos independentes, já que o uso de *softmax* supõe que cada *bounding box* possui exatamente uma classe, o que geralmente não é o caso.

Figura 21 – *Pipeline* de processamento da CNN YOLO.



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

3.5.4.1 Arquitetura da YOLOv3

A arquitetura original da CNN YOLO proposta por Redmon et al. (2015) tem 24 camadas convolucionais seguidas de 2 camadas totalmente conectadas. A arquitetura usada no desenvolvimento deste trabalho é uma versão atualizada da YOLO - a terceira versão da mesma, chamada de YOLOv3, com 53 camadas de convolução. Esta nova versão foi proposta por Redmon e Farhadi (2018), e implementada na biblioteca *Keras* por Trade (2018).

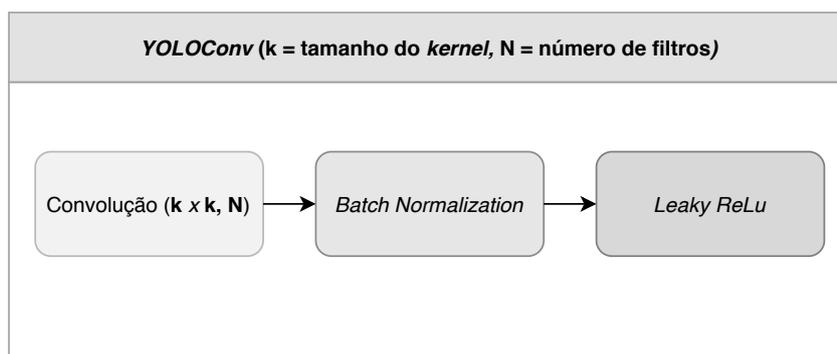
Nessa nova versão, a YOLO prevê as classes presentes nas *bounding boxes* usando três escalas diferentes, a fim de tentar contornar o problema de identificar objetos muito pequenos que as versões anteriores da rede apresentam. Para isso, várias camadas de convolução foram adicionadas, incluindo blocos de convolução e blocos residuais. Para entender a arquitetura completa da rede, precisa-se entender cada um dos blocos de convolução utilizados nela.

O primeiro bloco de convolução utilizado nesta nova arquitetura faz uma convolução em 2D com número N de filtros e *kernel* de tamanho k , seguida por uma camada de *Batch Normalization* e pela função de ativação *LeakyReLU* (Leaky Rectified Linear Activation), calculada como mostra a Equação 3.20. Esse bloco de convolução é chamado de *YOLOConv*, e será usado como base nos blocos seguintes. A Figura 22 ilustra a

estrutura desse bloco.

$$\phi = \begin{cases} x, & \text{se } x > 0 \\ 0.1x, & \text{caso contrário} \end{cases} \quad (3.20)$$

Figura 22 – O bloco básico de convoluções da YOLOv3.



Outro bloco de convolução utilizado nessa rede é o bloco residual, que apresenta estrutura similar ao proposto em Szegedy et al. (2017). Esse bloco é chamado de *YOLOResidual* e tem seu funcionamento ilustrado na Figura 23. O bloco recebe como entrada um bloco do tipo *YOLOConv* de kernel 3×3 , N filtros e *stride* 2, que é somado ao seu resultado após passar por dois outros blocos *YOLOConv*, o primeiro com kernel 1×1 , $\frac{N}{2}$ filtros e o segundo com kernel 3×3 , e N filtros. O bloco *YOLOResidual* é adicionado à arquitetura final da rede de forma iterativa. Quando isso acontece, a entrada do bloco *YOLOResidual* seguinte é a saída do bloco *YOLOResidual* anterior.

O terceiro bloco usado na arquitetura da rede corresponde às camadas finais dela, antes da saída, chamado aqui de *ÚltimasCamadas*. Este bloco consiste em 5 blocos do tipo *YOLOConv*, alternando blocos de *kernel* 1×1 e N filtros, com blocos de *kernel* 3×3 e $N \times 2$ filtros, como mostrado na Figura 24.

Em seguida, o resultado do bloco *ÚltimasCamadas* é passado como entrada para o bloco final da rede, chamado de *SaídaYOLO*. Este bloco é responsável pela saída da rede, e consiste em um bloco do tipo *YOLOConv* de *kernel* 3×3 e $N \times 2$ filtros, seguido de uma camada de convolução 2D sem função de ativação definida, funcionando como uma ativação linear. Esta última convolução é responsável pela parte de regressão da rede, possuindo *kernel* 1×1 , e um número S de filtros, que é a multiplicação do número de âncoras a serem usadas pela soma do número de classes mais 5, ou seja, $S = 9 \times (1 + 5)$. A Figura 25 ilustra a estrutura deste último bloco.

Figura 23 – Bloco residual da YOLOv3.

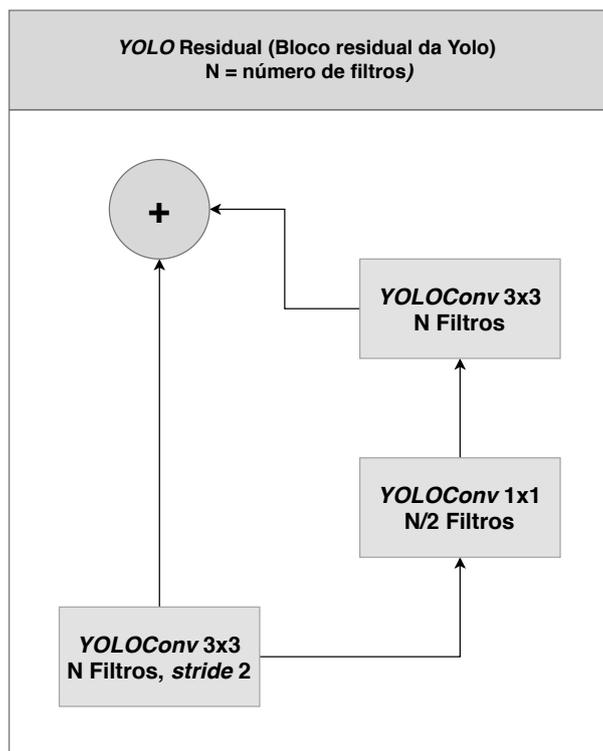


Figura 24 – Camadas finais da YOLOv3.

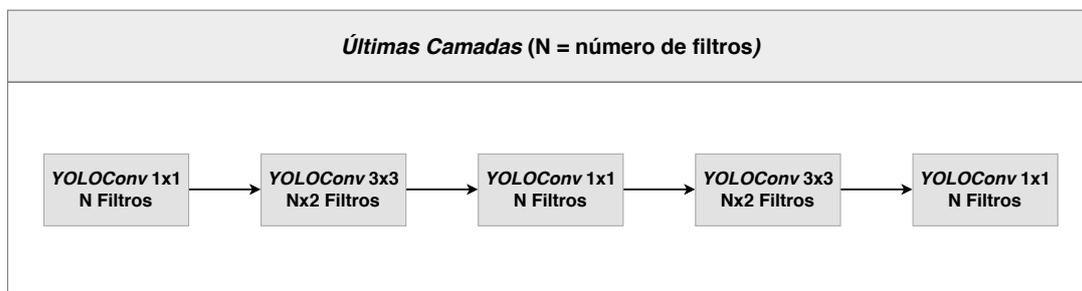
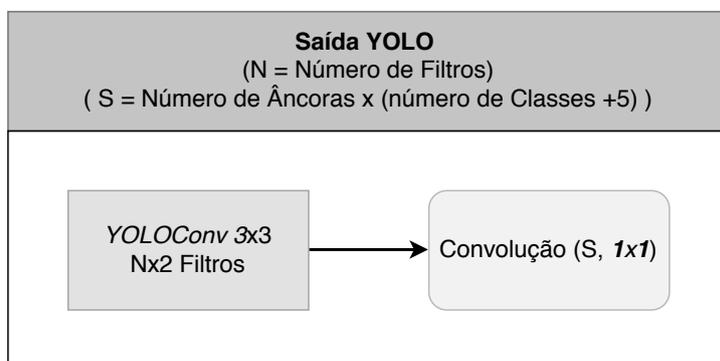


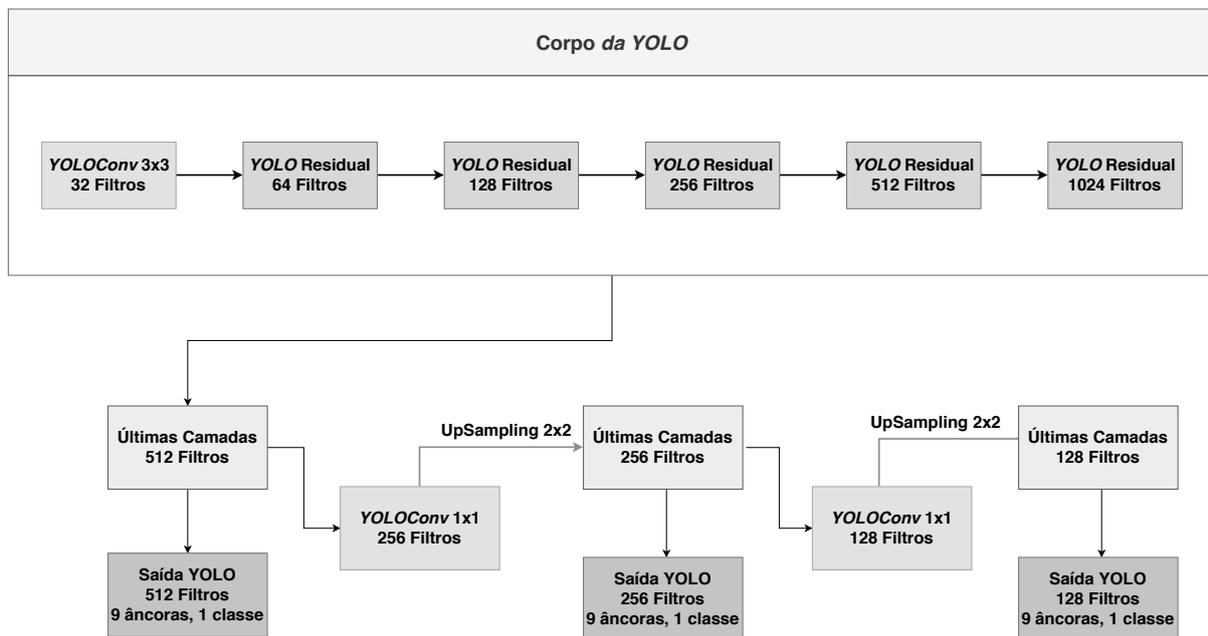
Figura 25 – Ilustração de como a saída rede é obtida. Já que a rede usa regressão para informar a saída, a última convolução é simples, sem função de ativação definida, para que a saída seja dada por uma ativação linear.



Tendo-se definido todos os blocos utilizados, pode-se entender a arquitetura completa da rede. O corpo da rede possui 24 blocos de convolução, resultando em 53 camadas de convolução; por isso, essa versão da YOLO também é conhecida por *DarkNet 53* (REDMON; FARHADI, 2018).

O resultado da rede é predito em 3 escalas diferentes, por isso são fornecidas 3 saídas. Para a primeira saída da rede, o resultado dos blocos de convolução que compõem o corpo da rede é passado para um bloco do tipo *Últimas Camadas*, que por sua vez é passado para o bloco *Saída YOLO* com 512 filtros. Para a segunda saída da rede, o resultado do bloco *Últimas Camadas* com 256 filtros é processado por um bloco de *YOLOConv* com 256 filtros com *kernel* 1×1 , por sua vez, esse resultado passa por uma camada de super amostragem. O resultado da super amostragem passa por um bloco de *Últimas Camadas* com 256 filtros, seguido por um bloco *Saída YOLO* com 256 filtros. A terceira saída da rede segue exatamente os mesmos passos da segunda saída, com a diferença que o número de filtros diminui para 128. A Figura 26 mostra a combinação de todos esses blocos, apresentando a visão geral da arquitetura da rede.

Figura 26 – Arquitetura completa da YOLOv3.



3.6 Filtro de Hampel

Um dos objetivos da metodologia deste trabalho é registrar a trajetória do movimento dos olhos durante um intervalo de tempo. Para isso, é preciso detectar a posição da pupila em cada quadro de cada vídeo gravado. Entretanto, os vídeos tem em média 285 quadros, e é provável que em alguns deles, o centro da pupila não seja detectado corretamente. Portanto, é necessário fazer um pós-processamento deste resultado para eliminar *outliers*, ou pontos fora da curva. Neste trabalho, usou-se o filtro de Hampel para esta tarefa.

Proposto por Hampel (1974), o filtro de Hampel é um filtro que pode ser aplicado em uma série temporal para identificar ruídos e substituí-los por valores mais condizentes. Este filtro usa uma janela deslizante de tamanho k que percorre toda a série temporal, onde para cada subconjunto englobado pela janela, calcula-se a mediana m_k e o desvio padrão adaptado S_k . Se o valor central x_k da janela de dados estiver distante o suficiente da mediana, substitui-se esse valor pela mediana. Para avaliar o quão distante um elemento está, é preciso definir um parâmetro de ajuste T , que indica a partir de quantos desvios padrões da mediana um número é considerado um ruído. O novo valor y_k da posição central da janela é calculado por:

$$y_k = \begin{cases} x_k, & \text{se } |x_k - m_k| \leq T \times S_k \\ m_k, & \text{se } |x_k - m_k| > T \times S_k \end{cases} \quad (3.21)$$

Para esse filtro, o desvio padrão S_k de cada janela é estimado de uma forma diferente, usando o desvio médio absoluto (DMA) (PEARSON et al., 2016). O DMA é o módulo da subtração da mediana m_k por cada elemento da janela deslizante, resultando em um novo subconjunto. Calcula-se a mediana desse novo subconjunto, multiplicando-a por um fator pré-definido, conforme mostra a Equação 3.22:

$$S_k = 1,4826 \times \text{mediana}_{j \in [-K, K]} \{ |x_{k-j} - m_k| \}, K = \frac{k}{2} \quad (3.22)$$

O filtro de Hampel pertence à classe dos filtros de decisão, e pode ser considerado uma adaptação do filtro da mediana (PEARSON et al., 2016). Um estudo comparativo feito por Dagar et al. (2018) mostrou que o filtro de Hampel é mais eficaz que o filtro da mediana na remoção de ruído em séries temporais, por isso, esse filtro foi escolhido

para pós-processamento em detrimento de outros filtros comuns.

3.7 Métricas de Avaliação

Para avaliar o resultado de algumas técnicas apresentadas neste capítulo, como a aplicação do método de *Isophotes* e da CNN YOLO, bem como do método como um todo, usou-se algumas métricas de validação.

Para analisar o desempenho geral da rede, quanto à classificação das imagens, usou-se as métricas precisão e sensibilidade. A precisão indica qual a porcentagem das previsões está correta, conforme a Equação 3.23: Ela é dada em função de Verdadeiros Positivos (**VP**) e Falsos Positivos (**FP**) (POWERS, 2011). Em relação à detecção dos olhos, **VP** são regiões da imagem que a rede classificou como olho e de fato são olhos, enquanto **FP** são partes da imagem que foram marcadas como olhos, mas não o são.

$$Precisão = \frac{VP}{VP + FP} \quad (3.23)$$

A sensibilidade avalia quão bem os resultados positivos foram encontrados, ou seja, em relação à classificação da rede neural, avalia quais dentre todos os olhos existentes na base de dados a YOLO conseguiu identificar corretamente. Ela é dada em função dos Verdadeiros Positivos (**VP**) e dos Falsos Negativos (**FN**), conforme mostra a Equação 3.24. Quanto à detecção dos olhos, **FN** são olhos que a CNN não classificou como tal.

$$Sensibilidade = \frac{VP}{VP + FN} \quad (3.24)$$

Para avaliar o resultado da detecção do centros dos olhos, usou-se a métrica proposta por Campadelli, Lanzarotti e Lipori (2007), que calcula o erro entre a posição estimada do centro dos olhos e a posição correta. O erro varia de 0 a 1, e quanto mais próximo de zero, mais preciso é o método de detecção. Tal erro é calculado usando a posição em x do olho esquerdo (C_{esq}) e a posição em x do olho direito (C_{dir}) obtidas pelo método de detecção do centro do olho, junto com a posição real de cada olho, C_{esqGT} e C_{dirGT} . A Equação 3.25 mostra como calcular o erro:

$$Erro = \frac{\max(\|C_{esq} - C_{esqGT}\|, \|C_{dir} - C_{dirGT}\|)}{\|C_{esqGT} - C_{dirGT}\|} \quad (3.25)$$

Para analisar o resultado do método proposto como um todo, em relação à classificação dos pacientes em doentes (apresentam paralisia) ou saudáveis, usou-se sensibilidade, especificidade e acurácia. Em relação à classificação dos pacientes em paréticos ou saudáveis, **VP** são pacientes que apresentam a doença e que o método classificou como doentes, e **FN** são pacientes que apresentam a doença mas o método classificou como saudável. Os VP e FN são usados no cálculo da sensibilidade, conforma já mostrado na Equação 3.24.

Quanto à especificidade, ela avalia os pacientes saudáveis, quantificando quantos deles foram classificados como saudáveis. Ela é dada em função dos Verdadeiros Negativos (**VN**), ou seja, os casos de pacientes saudáveis que foram classificados como tal, e dos Falsos Positivos (**FP**), referente aos pacientes saudáveis que o método classificou como doentes. A Equação 3.26 mostra como a especificidade é calculada:

$$Especificidade = \frac{VN}{VN + FP} \quad (3.26)$$

Para avaliar o resultado geral do método, usa-se a acurácia, que quantifica quantos pacientes foram classificados corretamente. A acurácia é dada em função dos **VP**, **VN**, **FP** e **FN**, conforme mostra a Equação 3.27:

$$Acurácia = \frac{VP + VN}{VP + FP + VN + FN} \quad (3.27)$$

Por fim, para avaliar confiabilidade dos resultados obtidos pelo método proposto, visando a sua aplicabilidade clínica, usa-se o coeficiente Kappa, proposto por Cohen (1960). Este coeficiente quantifica o relacionamento entre concordância e causalidade, e também o desacordo esperado, indicando o quanto os resultados obtidos são representações corretas das variáveis medidas (MCHUGH, 2012). O coeficiente Kappa é calculado da seguinte forma:

$$K = \frac{A - P_c}{1 - P_c} \quad (3.28)$$

onde A representa a acurácia e P_c é a probabilidade de concordância devido ao acaso, calculado como:

$$P_c = \frac{VP + FN}{D} \times \frac{VP + FP}{D} + \frac{VN + FP}{D} \times \frac{VN + FN}{D} \quad (3.29)$$

onde D é número total de exemplos (nesse caso, pacientes) presentes na base de teste.

Um estudo recente, apresentado por Koo e Li (2016) propõe a interpretação dos valores do coeficiente Kappa de acordo com a Tabela 3, onde quanto maior o valor, maior a sua confiabilidade. Para aplicação clínica, é desejável que o método proposto apresente o valor de coeficiente presente nas linhas 3 ou 4 da tabela.

Tabela 3 – Interpretação dos valores do coeficiente Kappa.

Coeficiente Kappa	Interpretação
$K < 0,5$	Pobre
$0,5 \leq K < 0,75$	Razoável
$0,75 \leq K < 0,90$	Bom
$K \geq 0,90$	Excelente

3.8 Considerações Finais

Neste capítulo, apresentou-se os fundamentos da patologia abordada nesta dissertação, bem como os conceitos e técnicas que fundamentam o desenvolvimento do método proposto neste trabalho. Explicou-se as estruturas básicas do olho, a natureza da paralisia do sexto nervo óptico, as formas usuais do seu diagnóstico, e a importância de um método alternativo para auxílio do diagnóstico dessa patologia. Em seguida, foram abordadas técnicas de Processamento Digital de Imagens e Redes Neurais que seriam usadas na metodologia, bem como métricas de avaliação dessas técnicas.

O capítulo a seguir explicará como cada técnica apresentada foi usada para desenvolver a metodologia desta dissertação, bem como os materiais usados na metodologia para auxílio do diagnóstico da paralisia do sexto nervo óptico através de vídeos.

4 MATERIAIS E MÉTODO

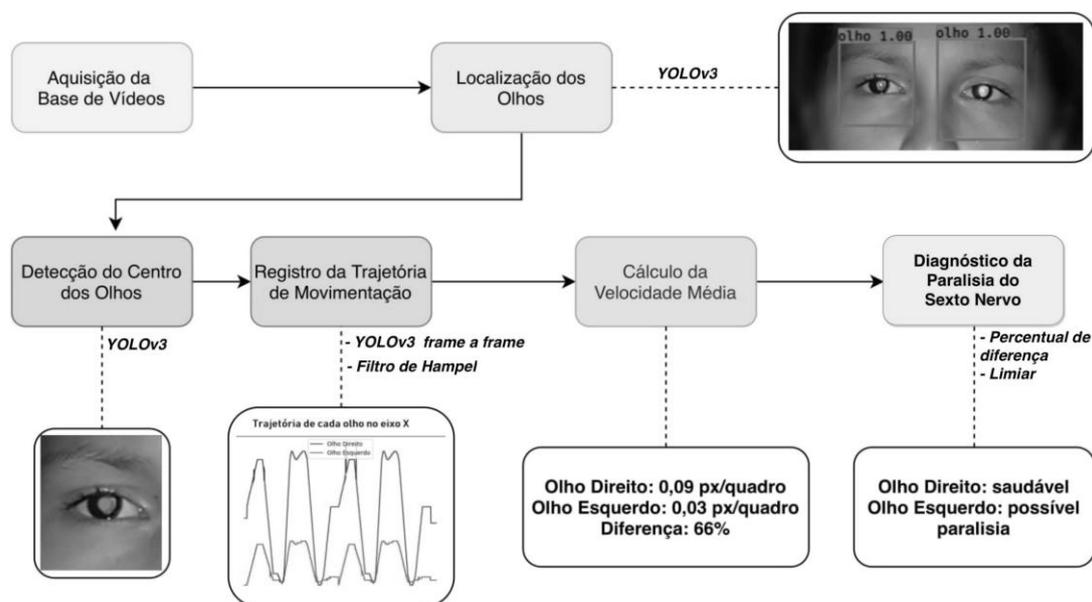
Este capítulo apresenta a metodologia desenvolvida e o que se pretende alcançar com ela, descrevendo cada etapa do método. Inicialmente, apresenta-se a base de vídeos em que o método desenvolvido foi aplicado, seguida pelas fases de localização dos olhos e detecção do centro da pupila. Os algoritmos de pós-processamento que foram aplicados para refinar o resultado de ambas as fases também são abordados. Por fim, descreve-se como foi obtido o resultado final do método, além de fornecer informações sobre o hardware e software utilizados no desenvolvimento desta pesquisa.

4.1 Método Proposto

A metodologia desenvolvida neste trabalho visa auxiliar no diagnóstico da paralisia do sexto nervo óptico. Para isso, desenvolveu-se um método baseado em vídeos que obtém a velocidade média com que cada olho se move em um período de tempo. Conforme a literatura, alguns trabalhos verificaram que há uma considerável diferença de velocidade entre um olho saudável e um olho com a paralisia do sexto nervo (METZ, 1976; ROSENBAUM; SANTIAGO, 1999). Entretanto, estes trabalhos usaram a velocidade sacádica como medida, que depende de equipamentos custosos para ser aferida. Neste trabalho, visa-se verificar se ao usar a velocidade média, obtida com câmeras comuns, a diferença ainda se mantém, e a partir dela, classificar olhos como “apresentam paralisia” ou “saudáveis”. Nesta metodologia, não haverá classificação específica entre paralisia parcial ou paralisia total, devido ao número limitado de vídeos de pacientes que apresentam a paralisia.

As etapas da metodologia desenvolvida consistem em: aquisição da base de vídeos, localização da região dos olhos através da CNN YOLOv3, pós-processamento do resultado da CNN, detecção do centro dos olhos baseado no centro das regiões detectadas pela YOLOv3 e pós-processamento desse resultado, registro da trajetória da movimentação de cada olho, cálculo da velocidade média para cada olho e diagnóstico da paralisia do sexto nervo. As principais etapas da metodologia estão representadas na Figura 27 e serão descritas nas seções a seguir.

Figura 27 – Principais etapas da metodologia desenvolvida.



4.1.1 Aquisição da Base de Vídeos

A base de dados usada nesta pesquisa consiste em vídeos de pacientes que apresentam a paralisia do sexto nervo e em pessoas que possuem os dois olhos saudáveis. Os vídeos de pacientes com a patologia foram fornecidos por um oftalmologista que colaborou com este trabalho e sugeriu o desenvolvimento do mesmo; os vídeos foram gravados em uma clínica oftalmológica em São Luís - MA. O próprio especialista gravou vídeos dos pacientes que apareciam em seu consultório apresentando a paralisia, por um período de 3 anos, começando em 2017. Entretanto, apenas 16 vídeos foram obtidos durante este período, dentre os quais 13 são de pessoas que apresentam a paralisia. O baixo número de vídeos é devido a raridade da patologia entre os pacientes atendidos pelo oftalmologista em questão. É importante lembrar que os pacientes que aceitaram colaborar com esta pesquisa, permitiram a gravação dos vídeos, assinando um termo de consentimento. Este projeto tem autorização do comitê de ética em pesquisa do Hospital Universitário Federal do Maranhão através do parecer N° 331/09, que garante o uso de imagens de humanos para fins de pesquisa acadêmica.

Inicialmente, o especialista não tinha grandes pretensões ao gravar os vídeos, portanto, não havia protocolo de aquisição. Alguns vídeos foram gravados com a câmera Samsung Galaxy EK-GC100 de 16 *megapixels* e outros com a câmera do *smartphone* do oftalmologista, sempre focada no rosto do paciente, enquanto o mesmo realiza os

movimentos persecutórios, apresentado no **Capítulo 3**. Os pacientes movimentavam ambos os olhos seguindo a trajetória de uma caneta ou de um dedo, que se movia da esquerda para a direita em um intervalo de tempo. Dezesseis vídeos da base foram adquiridos antes da definição do protocolo de aquisição. Entretanto, por não haver protocolo, em alguns vídeos, o rosto do paciente não está completamente voltado para a frente, o que faz com que parte de um dos olhos não fique completamente visível. Alguns vídeos precisaram ser manualmente editados (7 vídeos), cortando-se algumas partes em que o paciente não aparecia, ou em que havia muita movimentação da câmera, já que o especialista apoiou o *smartphone* nas mãos.

Posteriormente, definiu-se um protocolo de aquisição dos vídeos a fim de facilitar a aplicação da metodologia desenvolvida. Os vídeos podem ser gravados com câmeras comuns ou com *smartphones*, mas para isso, usa-se um tripé para garantir a estabilidade da câmera. Além disso, a fim de evitar ruídos no resultado final causados pelo balançar da cabeça do paciente, pede-se que este apoie os cotovelos em uma mesa, cerre os punhos, juntando-os e apoiando o queixo em ambos os punhos. O vídeo começa a ser gravado com o paciente olhando para a frente, com os olhos na posição primária (PPO). Ambos os olhos do paciente precisam estar visíveis no vídeo, e alinhados horizontalmente; não é preciso que o rosto inteiro do paciente esteja visível, mas não há problema se estiver. Em seguida, pede-se que o paciente olhe para a esquerda, depois que olhe para a direita. Esse movimento é repetido algumas vezes, com o objetivo de analisar a movimentação horizontal dos olhos. Após a definição deste protocolo, 16 novos vídeos foram gravados, todos com pessoas que não apresentam a paralisia em nenhum dos olhos. A base final contém 32 vídeos de 31 pacientes diferentes, sendo que 13 pacientes apresentam a paralisia e 19 são saudáveis.

Antes de serem processados, todos os vídeos da base foram redimensionados. Isso foi feito porque uma das técnicas testadas para a detecção do centro dos olhos, o método de *Isophotes*, funciona melhor em imagens degradadas ou de baixa resolução, como apresentado no **Capítulo 3**. Cada *frame* ou quadro de cada um dos vídeos obtidos foi utilizado na fase seguinte da metodologia, que será apresentada na seção a seguir.

4.1.2 Localização dos Olhos

A etapa de localização da região dos olhos é feita através da CNN YOLOv3, cuja arquitetura foi apresentada no **Capítulo 3**, Seção 3.5.4.1. Essa rede foi escolhida por sua rapidez de processamento quando aplicada em vídeos, e por sua eficiência na detecção de objetos. Para treinamento e validação da rede, foi utilizada uma base de dados privada apresentada por Almeida et al. (2015), que contém 243 imagens de rosto inteiro, totalizando 486 olhos. Nesta base, há imagens de 45 pacientes, com 5 ou 6 imagens de cada paciente olhando para diferentes direções, como para a frente, direita, esquerda etc. A base é composta por imagens de alta resolução, com 1.572×2.048 *pixels*. As imagens foram adquiridas pelo mesmo oftalmologista que forneceu a base de vídeos usada neste trabalho. As pessoas fotografadas concordaram em colaborar com a construção da base de dados, assinando um termo de consentimento. A Figura 28 exemplifica o tipo de imagens contidas na base. A seção a seguir apresenta os detalhes do treinamento da CNN.

Figura 28 – Exemplo dos movimentos oculares executados nos vídeos da base.



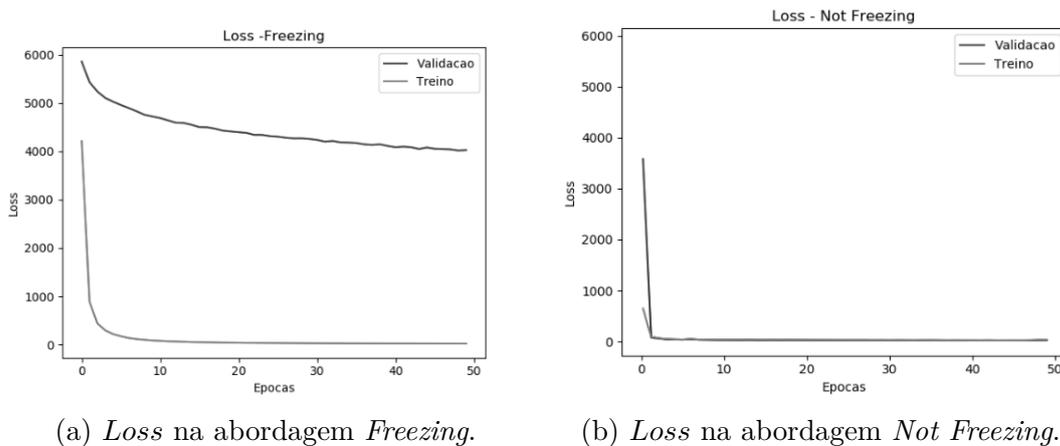
4.1.2.1 Treinamento da Rede YOLOv3

Para treinamento da CNN utilizada, fez-se aumento dos dados da base, que originalmente tem 243 imagens. Cada imagem foi espelhada, rotacionada em 90, 180 e 270 graus, gerando 4 novas imagens. Depois do aumento dos dados, a base passou a ter 1.215 imagens. Dessas imagens, 90% delas (1.095) foram usadas para treino e 10% (120) para validação. As imagens da validação foram separadas por paciente, para que não houvesse imagens do mesmo paciente durante treino e validação. As *bounding boxes* correspondentes à posição de cada olho foram definidas com base na marcação (*Ground Truth*) da íris de cada olho. Em seguida, foi gerado um arquivo de texto contendo o caminho para cada imagem, as *bounding boxes* encontradas na imagem referida, bem como a classe de cada uma delas. Tal arquivo foi gerado para o conjunto de imagens de treino e de validação.

Usou-se esta base de dados porque as imagens apresentam o rosto inteiro do paciente e porque se tem a marcação da posição dos olhos.

Para realizar *Transfer Learning*, a CNN usada nessa metodologia utilizou pesos pré-treinados na ImageNet (DAI et al., 2007; DENG et al., 2009). A rede foi treinada em 100 épocas utilizando a abordagem de *Freezing*, ou “congelamento” de algumas camadas. Todas as camadas entre o primeiro bloco *Yolo Residual* e o bloco *Saída YOLO* foram congeladas nas primeiras 50 épocas, sendo descongeladas da época 50 a 100. Essa abordagem faz com que os pesos dessas camadas não sejam atualizados, o que acelera o treinamento. Ao analisar as Figuras 29 e 30, que comparam as abordagens com congelamento (Figuras 29 (a) e 30 (a)) e a abordagem sem congelamento (Figura 29 (b) e 30 (b)) durante treino e validação, percebe-se que a *Loss* demora mais para decair na abordagem com congelamento, entretanto, uma vez que decai, ela é muito mais estável que na abordagem sem congelamento. A função de *Loss* avalia quão bem o método usado modela o conjunto de dados em que ele foi aplicado (ALGORITHMIA, 2018). Se as previsões do modelo estiverem muito distantes do valor esperado, a função de *Loss* apresentará valores altos. Como pode-se ver nas Figuras 29 (a) e 30 (a), o valor da *Loss* na abordagem com congelamento foi decaindo de forma gradual, mas estável. Isso pode indicar que a abordagem com congelamento, de forma geral, modelou melhor o problema, sendo mais confiável. Essa hipótese foi confirmada durante o teste do modelo, cujo resultado é apresentado na Seção 5.1.

Figura 29 – *Loss* do treino e validação nas primeiras 50 épocas nas duas abordagens.



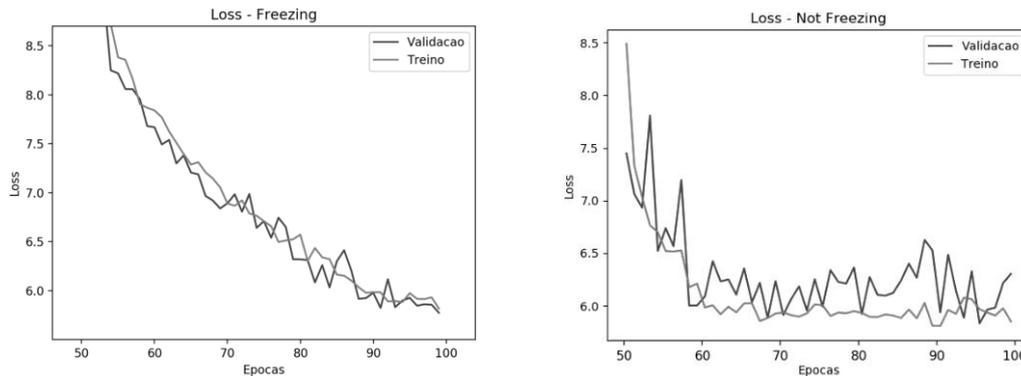
(a) *Loss* na abordagem *Freezing*.

(b) *Loss* na abordagem *Not Freezing*.

Após o treinamento, o modelo obtido foi usado para a detecção de olhos em cada *frame* dos vídeos utilizados, identificando *bounding boxes* referentes às regiões que contém

olhos. Cada região obtida foi analisada para determinar se pertencia ao olho esquerdo ou olho direito; a primeira *bounding box* da esquerda para a direita pertence ao olho esquerdo, e a segunda *bounding box*, ao olho direito. Entretanto, nem sempre a rede neural informa somente duas *bounding boxes*, o que demanda um pós processamento desse resultado, que será apresentado na seção a seguir.

Figura 30 – *Loss* do treino e validação nas últimas 50 épocas nas duas abordagens.



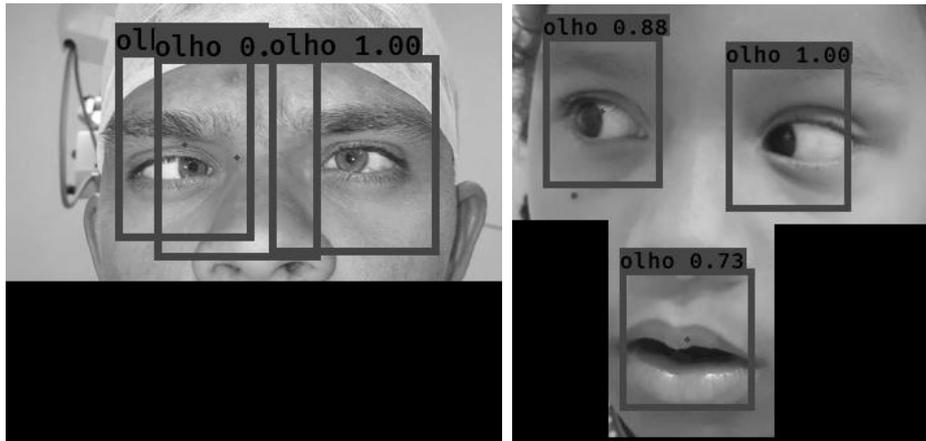
(a) *Loss* na abordagem *Freezing* nas 50 épocas finais. (b) *Loss* na abordagem *Not Freezing* nas 50 épocas finais.

4.1.2.2 Pós-processamento de *Bounding Boxes*

Apesar de apresentar bons resultados na detecção de objetos, algumas vezes a YOLO apresenta *bounding boxes* duplicadas, onde há interseção em uma mesma região, ou apresenta falsos positivos. Por isso, é necessário fazer algum pós-processamento a fim de eliminar resultados indesejados. Um dos casos indesejados é apresentado na Figura 31 (a), onde há *bounding boxes* repetidas numa mesma região. Para descartar *bounding boxes* duplicadas, as *bounding boxes* são arranjadas em pares e calcula-se a interseção entre cada par. Em seguida, organiza-se os pares em ordem crescente de acordo com a porcentagem de interseção; o primeiro par da lista representa o par desejável de *bounding boxes*, com a menor interseção entre si. Entretanto, esse não é o único critério a ser avaliado. É preciso que o par de *bounding boxes* esteja alinhado entre si. A Figura 31 (b) apresenta um caso em que nenhuma das *bounding boxes* se intercepta, entretanto, há a presença de um falso positivo. Para contornar esse resultado, é preciso aplicar a segunda etapa do pós-processamento.

Para contornar a presença de falsos positivos, seleciona-se apenas os pares de

Figura 31 – Exemplos de resultados indesejados.



(a) *Bounding boxes* repetidas para um mesmo olho.
 (b) Presença de falso positivo (Boca classificada como Olho).

bounding boxes que estejam alinhadas entre si. Para isso, verifica-se se as *bounding boxes* resultantes estão alinhadas, já que de acordo com Kasinski e Schmidt (2010), olhos costumam estar alinhados, apresentando ângulo máximo de alinhamento horizontal de 20° graus entre as pupilas. Usando esse limiar, é possível descartar *bounding boxes* cujo centros não estão alinhados. Esse processo é detalhado no Algoritmo 1:

Algoritmo 1: Verifica se um par *bounding boxes* está alinhado

Entrada: B_1 e B_2 , par de *bounding boxes*

Saída : Verdadeiro caso elas estejam alinhadas, Falso caso contrário

$p_1 \leftarrow (cx_{B_1}, cy_{B_1})$; // ponto de centro da *bounding box* 1

$p_2 \leftarrow (cx_{B_2}, cy_{B_2})$; // ponto de centro da *bounding box* 2

$x_{p_3} \leftarrow cx_{B_2}$;

$y_{p_3} \leftarrow cy_{B_1}$;

$p_3 \leftarrow (x_{p_3}, y_{p_3})$;

$angulo \leftarrow$ angulo entre p_1, p_2 e p_3 ;

if $angulo \leq 20^\circ$ **then**
 | **return** Verdadeiro;

else
 | **return** Falso;

end

Essa verificação é usada no Algoritmo 2, que apresenta como resultado o par de *bounding boxes* alinhadas que possuem a menor interseção possível, eliminando duplicações e falsos positivos.

Algoritmo 2: Encontra o par de *bounding boxes* alinhadas com menor interseção

Entrada: B , lista de *bounding boxes* detectadas pela YOLOv3

Saida : O par de *bounding boxes* alinhadas com a menor interseção entre si ou nulo, caso nenhum par atenda às restrições

$tamanho_B \leftarrow$ tamanho da lista B ;

$dict_I \leftarrow \emptyset$; // dicionário que armazena o valor de interseção entre pares de *bounding boxes*

for $i = 0$; $i < tamanho_B - 1$; $i = i + 1$ **do**

for $j = i$; $j < tamanho_B$; $j = j + 1$ **do**

$dict_I[i, j] \leftarrow$ interseção entre B_i e B_j ;

end

end

$dict_I \leftarrow sort(dict_I)$; // ordena $dict_I$ em ordem crescente de interseção

for each item em $dict_I$ **do**

$i, j \leftarrow$ índices das *bounding boxes* presentes em item;

if Algoritmo 1 (B_i, B_j) **then**

 // Algoritmo 1 verifica se as *bounding boxes* estão alinhadas

return (B_i, B_j);

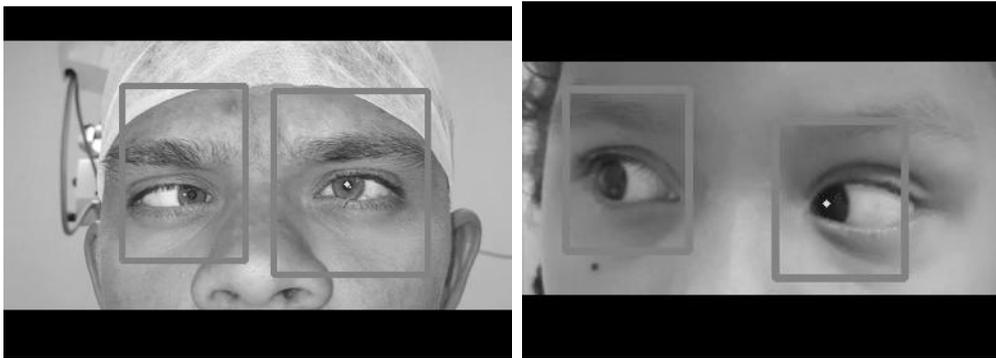
end

end

return Nulo

É importante lembrar que se em um *frame*, apenas um dos olhos é detectado ou se o pós-processamento resulta em um conjunto vazio, passa-se para o *frame* seguinte. O resultado final dessas duas etapas de pós-processamento (quando satisfeitas) corresponde somente a regiões que contém olhos, conforme mostram a Figura 32. Essas regiões são passadas para a segunda fase do método, a da detecção do centro da pupila.

Figura 32 – *Bounding boxes* resultantes após pós-processamento.



(a) Remoção de duplicações.

(b) Remoção de falso positivo.

4.1.3 Detecção do Centro dos Olhos

Para detecção do centro dos olhos, três abordagens principais foram aplicadas e testadas. Na primeira abordagem, a detecção do centro dos olhos deu-se através do cálculo da curvatura de *Isophotes*, apresentada na Seção 3.4.5. A segunda abordagem usa somente o centro das *bounding boxes* obtidas com a YOLOv3. A terceira abordagem é uma combinação da primeira e da segunda abordagem. No método final, apenas a segunda abordagem foi usada para a detecção do centro dos olhos, por apresentar a maior precisão na detecção. Entretanto, cada abordagem testada será explicada nos parágrafos a seguir e os resultados de cada uma delas serão apresentados no **Capítulo 5**, na Seção 5.2.

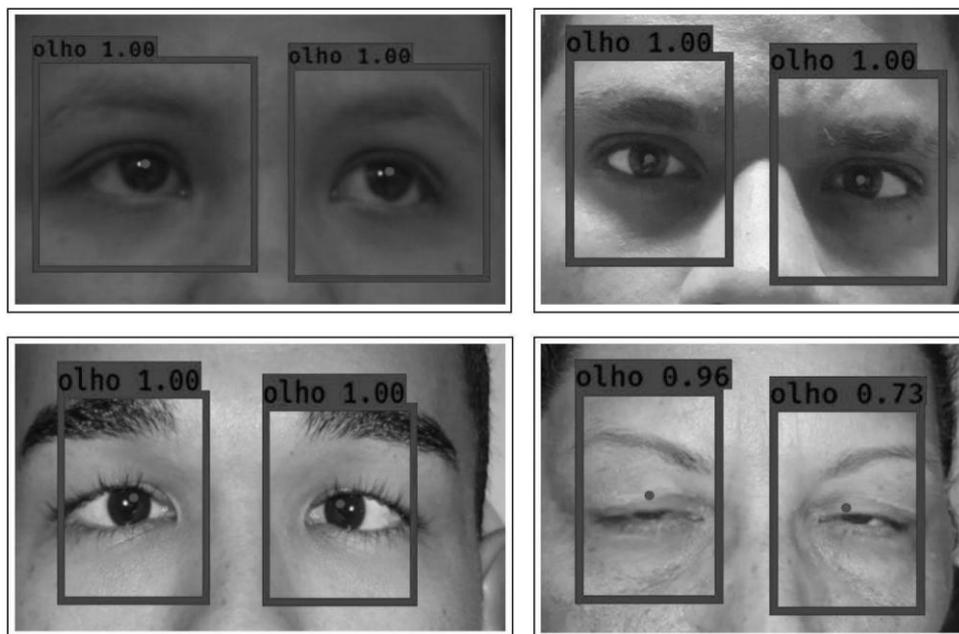
Para a abordagem 1, já que o método de *Isophotes* funciona melhor em imagens degradadas ou de baixa resolução, cada região da imagem obtida com a YOLOv3 foi transformada para níveis de cinza (SARAVANAN, 2010). Como pré-processamento, efetuou-se uma suavização da imagem através da filtragem Gaussiana, a fim de uniformizar as regiões da imagem. Verificou-se que ao utilizar um *kernel* 3×3 , a estimação do centro não retornou resultados satisfatórios, já que muitos pontos de brilho ainda permaneceram na imagem. Em vista disso, usou-se uma Gaussiana com *kernel* 9×9 . Em seguida, aplicou-se o método de *Isophotes*. Para o cálculo da curvatura de *Isophotes*, há apenas dois parâmetros que precisam ser definidos: raios mínimo e máximo, que são usados para rejeitar círculos muito grandes ou muito pequenos. Após alguns testes, os valores utilizados para raio mínimo e máximo ficaram entre 3% e 30%, respectivamente, em relação a largura da *bounding box* em *pixels*.

Inicialmente, apenas o resultado de *Isophotes* foi usado para estimar a posição do

centro dos olhos. Entretanto, em alguns vídeos, mesmo ao redimensionar, a área dos olhos permanecia muito grande e com resolução muito alta, o que dificultou o bom desempenho do *Isophotes*, conforme será mostrado no **Capítulo 5**, na Seção 5.2.

A segunda abordagem surgiu ao analisar as *bounding boxes* detectadas pela YOLOv3. Durante a etapa de pós-processamento das *bounding boxes* retornadas pela YOLOv3, ao plotar o centro das mesmas, percebeu-se que grande parte delas estavam centralizadas na pupila. Isso acontece porque, ao definir a região dos olhos na base de treino, tomou-se o centro de cada olho como ponto de partida, definindo-se a largura da *bounding box* como $[centro_x - offset, centro_x + offset]$ e altura como $[centro_y - offset, centro_y + offset]$ (*Offset* é um valor pré-definido correspondente à metade da largura de uma *bounding box*). Por a base de treino apresentar essa característica, a rede aprendeu e generalizou no teste. A Figura 33 mostra alguns resultados de centro de olhos obtidos com somente com a YOLOv3 (círculos azuis próximos às pupilas). Percebe-se que a rede conseguiu generalizar bem, centralizando a *bounding box* corretamente mesmo em imagens com iluminação ruim ou quando os olhos estão parcialmente fechados. A detecção do centro dos olhos usando somente o centro das regiões obtidas com a YOLOv3 foi a segunda abordagem testada.

Figura 33 – Centros de olhos obtidos com a YOLOv3 (Pontos azuis).

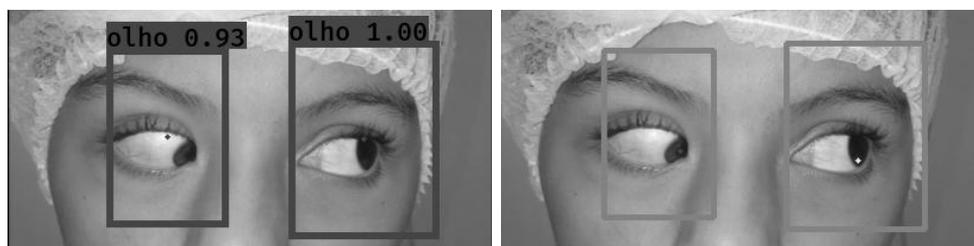


Entretanto, quando boa parte da íris está dentro dos cantos interno ou externo, estando parcialmente oclusa, a YOLOv3 tem dificuldade em centralizar o resultado na

pupila, conforme mostra a Figura 34 (a). Isso pode acontecer porque na base de imagens de treino não há muitos exemplos de pacientes olhando para os cantos do olhos de forma tão acentuada, o que geralmente ocorre em pacientes com estrabismo, onde há desvios de versão e os olhos não estão alinhados adequadamente (PINHEIRO, 2018). Por isso, decidiu-se combinar os resultados da detecção de centro dos olhos feito pelo *Isophotes* e pela YOLOv3, formando a terceira abordagem testada para detecção do centro dos olhos. Ao analisar visualmente o resultado dessa combinação, percebeu-se os problemas de ambas foram amenizados nos casos mais acentuados, como se pode ver na Figura 34 (a). A YOLOv3 é menos sensível ao tamanho e resolução das imagens, enquanto que o *Isophotes* detecta bem a pupila mesmo quando parte do olho está ocluso.

A terceira abordagem surgiu da combinação das abordagens 1 e 2. Para combinar o resultado de ambas as técnicas, primeiro calcula-se a diferença entre a posição encontrada pelo *Isophotes* e a posição encontrada pelo YOLOv3. Dá-se preferência à posição obtida com o *Isophotes*, mas se a diferença entre as posições for maior que um limiar (15 *pixels*), usa-se a posição obtida pela YOLOv3. Esse limiar foi obtido de forma empírica, após fazer vários testes e analisar o resultado nos vídeos.

Figura 34 – Comparação dos resultados da YOLOv3 e *Isophotes*.



(a) Centro do olho por YOLOv3 (Pontos azuis). (b) Centro do olho por *Isophotes* (Pontos azul e verde).

O resultado de cada uma dessas abordagens foi avaliado, a fim de determinar qual delas detectou o centro dos olhos de forma mais precisa. Por fim, somente a abordagem mais precisa é usada na detecção do centro dos olhos, que neste caso, foi a segunda abordagem, conforme mostra o **Capítulo 5**, na Seção 5.2.

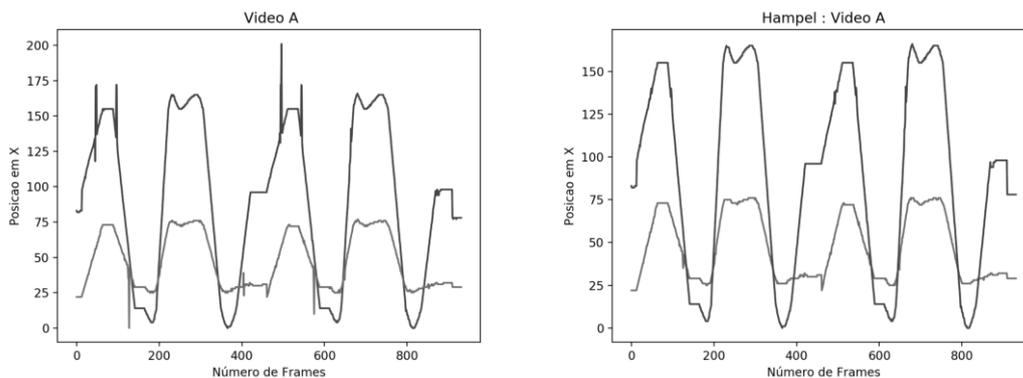
4.1.4 Registro da Trajetória de Movimentação dos Olhos

Para obter a velocidade média com que cada olho se move durante o vídeo, registra-se a trajetória de movimentação dos olhos. Isso é feito aplicando as etapas de

Localização dos Olhos (Seção 4.1.2) e **Detecção do Centro dos Olhos** (Seção 4.1.3) para cada olho presente em cada *frame* do vídeo. Assim, obtém-se o espaço percorrido por cada olho, e constrói-se a visão geral da movimentação de cada olho. Conforme apresentado no **Capítulo 2**, a visão geral da trajetória de movimentação dos olhos pode ajudar na análise de casos com paralisia parcial.

Após obter as posições de cada olho durante o vídeo, têm-se um histórico das posições que funciona como uma série temporal da movimentação dos olhos. Durante o processamento do vídeo, é possível que em alguns *frames*, as posições tenham sido detectadas erroneamente, configurando ruído. A fim de contornar esse problema, aplica-se o Filtro de Hampel (Seção 3.6) para atenuar ruídos presentes em séries temporais. Há duas variáveis a serem definidas ao se usar o Filtro de Hampel, o tamanho da janela deslizante e a quantidade de desvios padrões que configuram um ruído. Neste trabalho, após alguns testes, usou-se uma janela de tamanho 5 e desvio padrão 2. As Figuras 35 (a) e (b) mostram o efeito da aplicação da filtragem.

Figura 35 – Trajetória de movimentação dos dois olhos num intervalo de tempo. Em verde: olho esquerdo; em azul : olho direito.



(a) Resultado sem filtragem.

(b) Resultado pós filtro de Hampel.

Para registrar a trajetória de movimentação, optou-se por aplicar a detecção dos olhos através da YOLO em todos os *frames* do vídeo, em vez de usar algoritmos de *tracking* ou rastreamento. Isso aconteceu porque, ao testar o uso de *tracking* em um pequeno intervalo de *frames*¹, notou-se que a área referente aos olhos detectados pelo *tracking* era muito grande, dificultando a fase de detecção do centro dos olhos. Os algoritmos de *tracking* testados neste trabalho foram *GOTURN*, *Kernelized Correlation*

¹ A cada 5 *frames* aplicou-se a CNN YOLO, e nos 4 *frames* nesse intervalo, aplicou-se *tracking*.

Filters (KCF) e *MedianFlow*, todos disponíveis na biblioteca de visão computacional OpenCV (ITSEEZ, 2015). Além disso, o método de *tracking* não centraliza a *bounding box* na pupila, o que também dificulta o melhoramento do resultado do *Isophotes*. Entretanto, aplicar a rede neural e os algoritmos de pós-processamento em cada *frame* do vídeo comprometeu a velocidade do método. Sacrificou-se o funcionamento em tempo real por precisão; em média, cada vídeo foi processado a 3 FPS. Após a filtragem, o conjunto de posições resultantes é usado no cálculo da velocidade média, a ser explicada na seção seguinte.

4.1.5 Cálculo da Velocidade Média

De acordo com Halliday, Resnick e Walker (2013), uma das formas mais comuns de descrever o movimento de um objeto é através da sua posição em função do tempo, para isso, pode-se calcular a velocidade média. Ela é a razão entre o deslocamento (ΔS) e o tempo de duração do deslocamento (ΔT), conforme mostra a Equação 4.1. Para calcular o (ΔS), deve-se considerar as posições inicial e final, e para o cálculo do (ΔT) deve-se levar em conta o tempo inicial e final. No contexto desta metodologia, calculou-se a velocidade apenas em relação ao eixo X , já que o sexto nervo óptico inerva o músculo reto lateral, que é responsável por um dos movimentos dos olhos no plano horizontal. Neste caso, ao calcular o ΔS , verifica-se quantos *pixels* o olho se moveu de um *frame* para outro, começando da posição do olho no tempo (*frame*) inicial e terminando com a posição do olho no tempo (*frame*) final. Os conceitos de tempo inicial e final foram dados em função da quantidade de *frames* existentes no vídeo utilizado.

$$Velocidade_{media} = \frac{\Delta S}{\Delta T} = \frac{Posicao_{Final} - Posicao_{Inicial}}{Tempo_{Final} - Tempo_{Inicial}} \quad (4.1)$$

A velocidade média é expressada em *pixels/segundos*, convertendo-se a quantidade de *frames* para segundos através da divisão do número de *frames* pelo FPS do vídeo.

4.1.6 Diagnóstico da Paralisia do Sexto Nervo

Após calcular a velocidade média para os olhos esquerdo e direito, calcula-se também o percentual de diferença entre elas de acordo com a Equação 4.2.

$$Diferença = \left(1 - \frac{\text{mínimo}(velocidadeMedia_{esquerdo}, velocidadeMedia_{direito})}{\text{máximo}(velocidadeMedia_{esquerdo}, velocidadeMedia_{direito})}\right) \times 100 \quad (4.2)$$

Para o conjunto de vídeos de pacientes que apresentam a paralisia, obtém-se a média desse percentual, fazendo-se o mesmo para o conjunto de vídeos de pacientes saudáveis. As médias obtidas são usadas na definição de um limiar para diferenciar pacientes com paralisia de pacientes saudáveis. Se o percentual da diferença entre as velocidades dos olhos do paciente for maior que o limiar estipulado, o paciente é diagnosticado como portador da paralisia do sexto nervo no olho que apresenta a menor velocidade média. Caso contrário, o paciente é diagnosticado como saudável em ambos os olhos. Este limiar será apresentado e discutido no **Capítulo 5**.

4.2 Hardware e Softwares Utilizados

A metodologia apresentada neste capítulo foi desenvolvida por completo na linguagem de programação *Python*, na versão 3.6 (ROSSUM; JR, 1995; ROSSUM; DRAKE, 2009). Esta linguagem foi escolhida por oferecer grande suporte ao desenvolvimento de sistemas baseados em Aprendizado de Máquina, e ao processamento e análise de dados científicos. Dentre as principais bibliotecas em *Python* usadas no desenvolvimento desta pesquisa estão: *Keras* (CHOLLET et al., 2015), na versão 2.1 e *Tensorflow* na versão 1.8 (ABADI et al., 2016), ambas usadas para o treinamento de modelos de aprendizagem profunda. O computador utilizado para o desenvolvimento de boa parte da metodologia conta com sistema operacional *Windows 10*, com processador Intel Core i7-7700, 16GB de memória RAM e 1TB de HD. Para o treinamento da CNN YOLOv3, utilizou-se a GPU NVIDIA 1080Ti com *frame buffer* de 11 *gigabytes*.

4.3 Considerações Finais

Este capítulo apresentou uma metodologia desenvolvida para auxiliar na detecção da paralisia do sexto nervo óptico através de vídeos. Apresentou-se a base de vídeos em que a metodologia foi aplicada e seu protocolo de aquisição. Em seguida, detalhou-se cada etapa da metodologia, a começar pela localização da região dos olhos, feita pela CNN YOLOv3. Após localizar os olhos, determina-se o centro dos mesmos através dos valores de centro das regiões retornadas pela rede YOLOv3. Este processo é repetido durante toda a extensão dos vídeos da base, registrando-se assim a trajetória de movimentação de cada olho e calculando-se a velocidade com que cada olho se move. A partir das velocidades médias, calcula-se o percentual de diferença entre elas, a fim de obter um limiar que ajude a diagnosticar olhos que apresentam a paralisia e olhos saudáveis. Por fim, apresenta-se o software e hardware usado durante o desenvolvimento desta metodologia. Os resultados finais deste trabalho são apresentados e discutidos no capítulo a seguir.

5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta os resultados obtidos pela metodologia proposta para auxílio no diagnóstico da paralisia do sexto nervo óptico através de vídeos digitais. Os resultados de cada etapa da metodologia são apresentados e discutidos. Avaliou-se desde o resultado da rede neural usada para localização de olhos, como a técnica usada para detecção do centro de olhos, até o resultado final do método, com o cálculo das velocidades médias para cada olho e as diferenças entre elas, classificando os pacientes em portadores da paralisia ou saudáveis. Por fim, discute-se se o resultado final obtido pela metodologia proposta contribui para a resolução do problema abordado neste trabalho.

5.1 Localização da Região dos Olhos

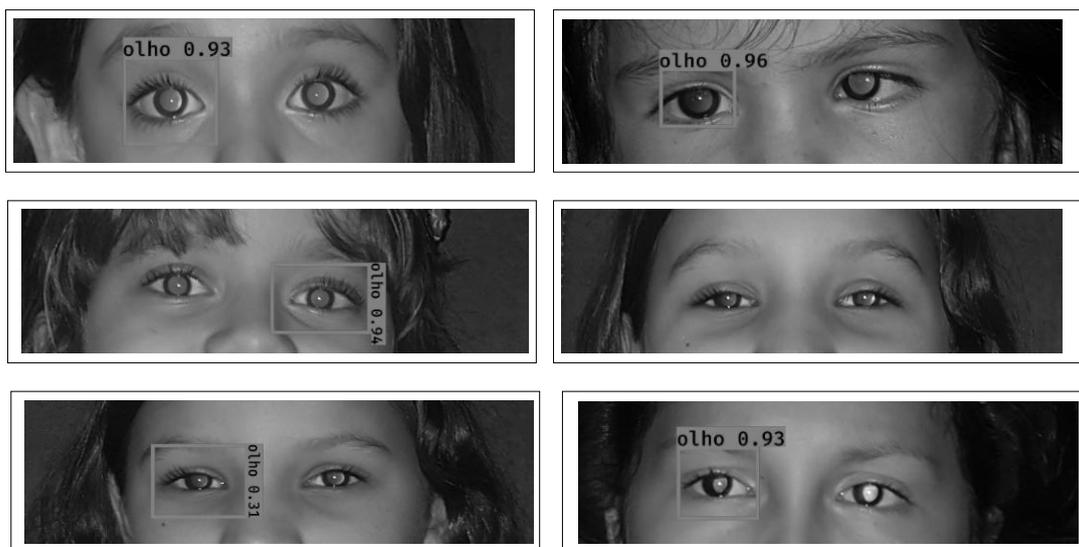
Para a localização da região dos olhos em vídeos, usou-se a CNN YOLOv3, apresentada na Seção 3.5.4.1. Foram aplicadas duas abordagens para treinamento da rede neural, uma baseada em congelamento de camadas (*Freezing*) e outra sem congelamento (*Not Freezing*). Para avaliar qual abordagem foi mais eficiente da detecção de olhos, testou-se cada uma delas em uma base de imagens privada com 60 imagens de rostos inteiros, totalizando 120 olhos. As imagens foram obtidas durante o exame de análise do reflexo vermelho e também foram fornecidas pelo oftalmologista local que colaborou com esta pesquisa. A Tabela 4 apresenta os resultados usando a métrica Precisão, apresentada na Seção 3.7.

Tabela 4 – Resultado da Precisão da detecção em ambas as abordagens.

Abordagem	Olhos Existentes	VP	FP	Precisão
1 - <i>Freezing</i>	120	120	5	96%
2 - <i>Not Freezing</i>	120	113	5	95,7%

A abordagem 1 com congelamento de camadas apresentou maior precisão, detectando todos os olhos presentes na base, enquanto que a abordagem 2 deixou de detectar alguns, como mostra a Figura 36. Os falsos positivos (FP) presentes em ambas as abordagens são descartados na fase de pós-processamento apresentada na Seção 4.1.2.2, portanto, levando em conta somente os verdadeiros positivos (VP), o resultado da abordagem 1 foi melhor. Diante da precisão superior da abordagem 1, escolheu-se aplicá-la em todos os vídeos da base. Para avaliar o resultado desta abordagem ao ser aplicada em vídeos, comparou-se o resultado da detecção de olhos em cada *frame* de cada um dos 32 vídeos

Figura 36 – Exemplos de olhos que a abordagem *Not Freezing* não conseguiu detectar.



da base com o resultado esperado, chamada de *Ground Truth*. Construiu-se a *Ground Truth* para 9.445 imagens, totalizando 18.890 olhos. A Tabela 5 apresenta a Precisão e Sensibilidade média da rede.

Tabela 5 – Precisão e Sensibilidade média da YOLOv3 ao ser testada nos vídeos da base.

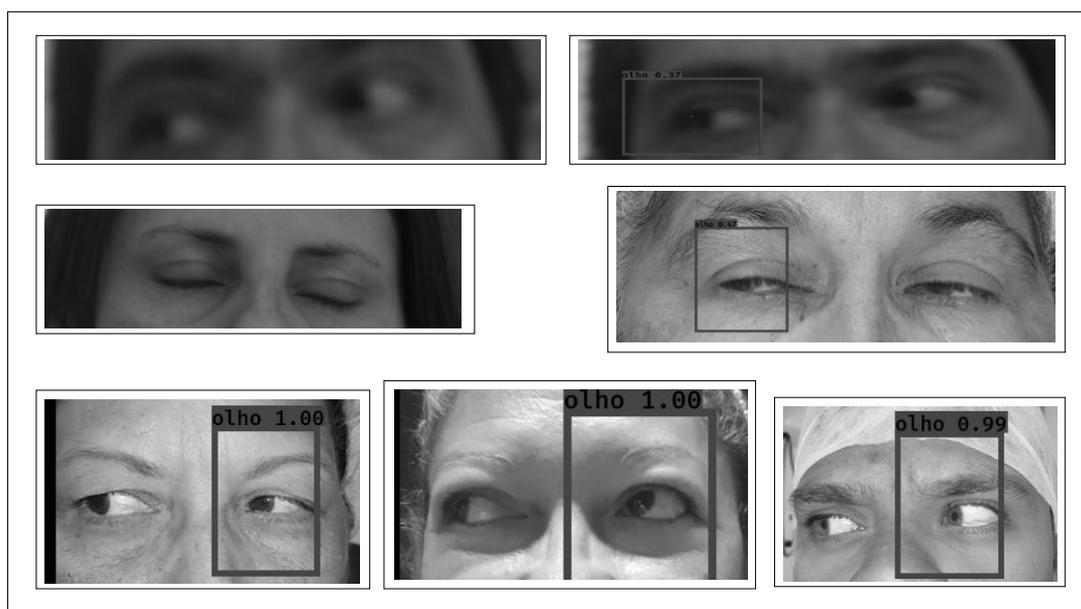
Técnica	Olhos	Precisão	Sensibilidade	Olhos Não Detectados
YOLOv3	18.890	98,99	98,46	273

O resultado da CNN YOLOv3 para a detecção de olhos indica grande confiabilidade na detecção. Ao analisar os poucos olhos que não foram detectados, percebeu-se que a rede teve problemas ao identificar olhos em imagens desfocadas, em imagens que o paciente aparece com olho totalmente ou parcialmente fechado e em imagens em que o paciente está olhando para o lado de forma acentuada, como mostra a Figura 37. A Tabela 9 mostra a precisão e sensibilidade para cada vídeo individualmente. Diante do bom resultado da CNN YOLOv3, cada região detectada como olho é passada para a fase seguinte, a detecção do centro dos olhos.

5.2 Detecção do Centro dos Olhos

Para a detecção do centro dos olhos, usou-se três abordagens principais: curvatura de *Isophotes*, centro das regiões detectadas pela YOLO e combinação de ambas as abordagens, conforme apresentado na Seção 4.1.3. Antes de aplicar a primeira abordagem na base de vídeos, avaliou-se o seu desempenho em uma base de imagens de olhos.

Figura 37 – Exemplo de olhos que a YOLO teve dificuldade em detectar.



Inicialmente, o método da curvatura de *Isophotes* foi aplicado em uma base privada com 96 imagens de olhos, com resolução de 400×400 pixels. Entretanto, ao ser aplicado em imagens com boa resolução, o método de *Isophotes* não apresentou bons resultados, mas apresentou resultados melhores quando aplicado em imagens de resolução menor (100×100). Isso vai de acordo com o que Valenti e Gevers (2008) observaram, que o método é indicado para aplicação em imagens de baixa resolução. Por isso, todas as imagens da base foram redimensionadas para terem resolução 100×100 pixels. A porcentagem do erro da posição obtida pelo método de *Isophotes* em relação à posição correta é apresentado na Tabela 6.

Tabela 6 – Porcentagem do erro em relação ao centro dos olhos ao aplicar o método de *Isophotes* em diferentes resoluções de imagem.

Técnica	Resolução	Erro
<i>Isophotes</i>	400×400 pixels	27,8%
<i>Isophotes</i>	100×100 pixels	8,4%

Quando aplicada em imagens de baixa resolução, o método de *Isophotes* obteve resultados melhores, reduzindo o erro de 27,8% para 8,4%. Entretanto, mesmo em baixa resolução, ao receber imagens com muita iluminação, detalhes como o reflexo do óculos comprometeram a detecção correta do centro do olho. Apesar disso, devido ao resultado promissor da abordagem com imagens redimensionadas, o método de *Isophotes* foi aplicado na base de vídeos para detecção do centro dos olhos, conforme explicado em 4.1.1. Além

dessa abordagem (*Isophotes*), também usou-se o centro das regiões detectadas pela YOLO (YOLOv3) e a combinação do resultado do método de *Isophotes* e da YOLO (*Isophotes* + YOLOv3).

Para avaliar qual abordagem obteve o melhor resultado ao ser aplicada na base de vídeos, comparou-se o resultado obtido por cada uma delas com a *Ground Truth*, que nesse contexto representa a posição correta do centro de cada olho. A comparação dos resultados produz um erro, cuja obtenção é descrita na Seção 3.7. A detecção do centro dos olhos foi aplicada em cada região obtida na etapa de localização dos olhos, totalizando 18.344 olhos. Antes do cálculo do erro, aplica-se o filtro de Hampel, apresentado na Seção 3.6, sobre o conjunto de centros de olhos de cada vídeo, a fim de atenuar ruídos. A Tabela 7 apresenta o erro em relação à *Ground Truth* para cada abordagem testada.

Tabela 7 – Erro médio das abordagens testadas para detecção do centro do olho.

Técnica	Erro
<i>Isophotes</i>	11,61%
YOLOv3	5,09%
<i>Isophotes</i> + YOLOv3	6,77%

A abordagem que usou somente a técnica de *Isophotes* obteve o maior erro, mesmo ao usar imagens com resolução menor. Entretanto, ao combinar esta técnica com o resultado da YOLOv3, o erro diminuiu de 11,61% para 6,77%. Ainda assim, a abordagem que usou somente os valores de centro das regiões retornadas pela rede YOLOv3 obteve o resultado mais próximo do real, com erro médio de 5,09%. O bom resultado da YOLOv3 pode ser explicado por sua boa capacidade de generalização, funcionando bem em diferentes cenários e iluminações, ao contrário da técnica de *Isophotes*, cujo funcionamento é comprometido se houver muitos detalhes e reflexos nas imagens, ou se mesmo após o redimensionamento, a imagem permanecer em boa resolução. Ao analisar o resultado da YOLOv3 na base de vídeos que foi obtida de acordo com o protocolo de aquisição, percebe-se que o erro é ainda menor (3,92%), conforme mostra a Tabela 8, o que comprova que gravar os vídeos com estabilidade na imagem e iluminação uniforme contribui para o bom funcionamento da metodologia proposta.

A Tabela 9 apresenta o erro médio para cada vídeo, onde os vídeos de 1 a 16 foram obtidos antes da definição do protocolo de aquisição, e os vídeos restantes seguiram o protocolo. Devido ao bom resultado da abordagem com YOLOv3 para detecção do

Tabela 8 – Erro médio para vídeos que não seguiram o protocolo de aquisição e para vídeos que seguiram.

Vídeos	Erro
<i>Sem protocolo</i>	6,25%
Com protocolo	3,92%

Tabela 9 – Precisão e Sensibilidade da detecção de olhos da rede YOLO, e erro médio da detecção do centro dos olhos em X para cada vídeo. Vídeos de 1 a 16 foram gravados antes da definição do protocolo de aquisição e vídeos de 17 a 32 foram gravados de acordo com o protocolo de aquisição.

Vídeo	Olhos	Precisão (YOLO)	Sensibilidade (YOLO)	Erro
1	444	100%	100%	5,65%
2	646	97,6%	97,5%	9,79%
3	468	100%	99,1%	0,83%
4	582	100%	100%	2,56%
5	452	100%	99,3%	1,99%
6	626	100%	98,2%	8,22%
7	662	99,5%	92,5%	18,87%
8	776	100%	100%	10,36 %
9	436	100%	100%	3,69%
10	848	100%	100%	2,01%
11	478	100%	94,7%	0,96%
12	318	100%	99%	6,94%
13	494	100%	85,6%	6,06%
14	314	100%	100%	4,91%
15	356	71,7%	100%	12,63%
16	528	100%	98,6%	4,63%
17	548	100%	93,2%	3,30%
18	662	100%	99,8%	6,84%
19	576	100%	100%	4,38%
20	556	100%	100%	0,79%
21	576	98,9%	99,3%	5,18%
22	730	100%	98,3%	3,99%
23	484	100%	100%	5,22%
24	496	100%	98,1%	1,49%
25	414	100%	100%	4,46%
26	512	100%	99,8%	1,42 %
27	610	100%	98,3%	4,05%
28	850	100%	99,6%	3,82%
29	1368	100%	99,9%	4,92%
30	814	100%	99,2%	3,39%
31	522	100%	100%	6,89%
32	744	100%	100%	2,69%

centro dos olhos, seus valores foram usados para o cálculo da velocidade média, cujos resultados são apresentados e discutidos na seção a seguir.

5.3 Velocidade Média

Para analisar o resultado do método como um todo, é preciso ter um limiar que ajude a diferenciar pacientes que apresentam a paralisia do sexto nervo óptico de pacientes saudáveis. Para isso, calculou-se a velocidade média do deslocamento de cada olho no eixo X, em todos os vídeos presentes na base de dados, conforme explicado na Seção 4.1.5. Em seguida, calculou-se a diferença entre as velocidades dos olhos de um mesmo paciente. A partir desse percentual de diferença, define-se um valor base para guiar o diagnóstico do oftalmologista.

Ao analisar o resultados por grupo, o primeiro grupo é composto por 13 vídeos de pacientes que apresentam a paralisia do sexto nervo óptico, de um total de 32 vídeos presentes na base. A Tabela 10 apresenta a velocidade média para cada olho, assim como a porcentagem de diferença entre eles. Para cada vídeo, as colunas marcadas em vermelho representam o olho com menor velocidade e possível paralisia de acordo com a metodologia proposta, enquanto que as colunas marcadas com (“doente”) representam o olho que de fato apresenta paralisia, de acordo com o diagnóstico do oftalmologista.

Tabela 10 – Velocidade Média de cada olho em pacientes em que um dos olhos apresenta a paralisia do sexto nervo óptico. A velocidade é dada em *pixels*/segundo.

Vídeo	Vel. Esquerdo <i>pixels</i> /s	Vel. Direito <i>pixels</i> /s	Diferença
1	3,91	4,51 (“doente”)	13,27%
2	14,47	7,23 (“doente”)	50,00%
3	10,12 (“doente”)	14,40	29,72%
4	6,56 (“doente”)	8,49	22,73%
5	6,18 (“doente”)	7,42	16,67%
6	3,66	2,72 (“doente”)	25,69%
7	4,94 (“doente”)	4,09	17,16%
8	5,44	4,18 (“doente”)	23,04%
9	7,86	3,39 (“doente”)	56,89%
10	8,73 (“doente”)	10,16	14,05%
11	19,54	13,40 (“doente”)	31,44%
12	7,77 (“doente”)	14,69	47,09%
13	9,33	5,13 (“doente”)	45,00%

Ao analisar os resultados individuais, percebe-se que 4 vídeos apresentaram percentual de diferença abaixo do mínimo estipulado por Metz (1983) (20%), nos vídeos

1, 5, 7 e 10, portanto, seriam considerados saudáveis. Em dois desses casos, nos vídeos 1 e 7, além do percentual de diferença estar abaixo do esperado, o método atribuiu a menor velocidade ao olho saudável, o que está incorreto. É importante lembrar que os vídeos do primeiro grupo foram adquiridos ao longo do desenvolvimento deste trabalho, mas antes da definição do protocolo de aquisição. Conforme visto na Tabela 8, o erro em relação à posição correta dos centros dos olhos é maior nos vídeos que não seguiram o protocolo. Isso pode refletir no bom funcionamento do restante do método, levando a resultados incorretos durante o cálculo da velocidade; nos casos dos vídeos 1, 5, 7 e 10, isso aconteceu.

Em relação ao segundo grupo, onde os pacientes não apresentam a paralisia em nenhum dos olhos, usou-se 19 dos 32 vídeos presentes na base de dados. Dentre esses 19 vídeos, 3 foram obtidos antes da definição do protocolo de aquisição (vídeos 14 a 16), e os 16 vídeos restantes foram gravados de acordo com o protocolo. A Tabela 11 apresenta a velocidade média para cada olho e o percentual da diferença entre elas.

Tabela 11 – Velocidade Média de cada olho em pacientes que não apresentam paralisia do sexto nervo óptico. A velocidade é dada em *pixels*/segundo.

Vídeo	Vel. Esquerdo <i>pixels</i> /s	Vel. Direito <i>pixels</i> /s	Diferença
14	5,53	5,14	7,06 %
15	4,42	4,24	3,90 %
16	18,79	18,32	2,53 %
17	11,73	11,64	0,72 %
18	4,95	5,57	11,11 %
19	5,42	4,89	9,80 %
20	7,75	7,49	3,32 %
21	5,19	4,69	9,66 %
22	4,56	3,67	19,50 %
23	6,78	5,55	18,12 %
24	6,14	6,75	8,97 %
25	6,61	6,36	3,73 %
26	6,53	6,20	4,91 %
27	9,32	8,80	5,51 %
28	10,76	9,48	11,91 %
29	4,21	3,93	6,74 %
30	7,21	7,50	3,75 %
31	2,19	2,11	3,57 %
32	8,01	7,00	12,67 %

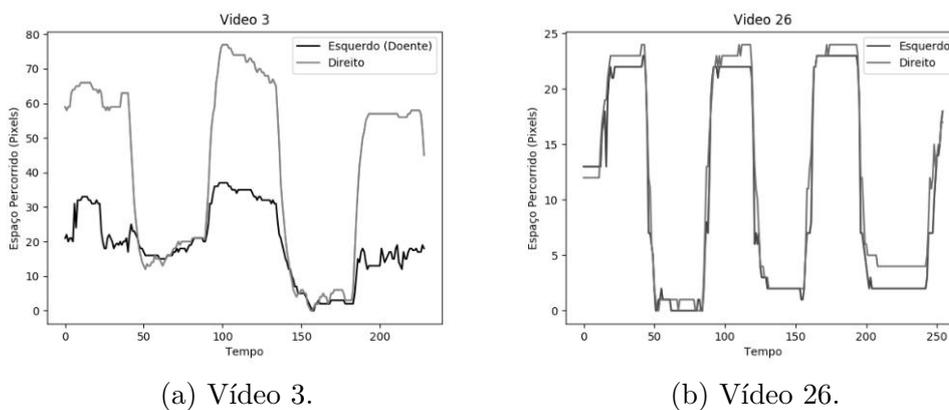
De forma geral, os resultados do segundo grupo foram mais consistentes. Nenhum vídeo apresentou diferença percentual que pudesse indicar que o paciente apresenta a

paralisia do sexto nervo, todos ficaram abaixo de 20%. O resultado mais uniforme se deve à detecção mais precisa do centro dos olhos, já que quando se segue o protocolo de aquisição o erro médio é menor, conforme visto na Tabela 8. A detecção mais precisa resulta no cálculo da velocidade média sob valores mais condizentes com o real.

Apesar do bom resultado geral pro segundo grupo, dois vídeos apresentaram percentual de diferença alto para olhos saudáveis. Os vídeos 22 e 23 apresentaram diferença de 19,50% e 18,12%, respectivamente, um valor muito próximo do esperado para olhos que apresentam a paralisia. Apesar de ambos os vídeos apresentarem valores pequenos para erro médio, 3,99% e 5,22%, respectivamente, ao analisar o resultado da detecção do centro dos olhos nesses vídeos, percebe-se que a rede não detectou bem o centro quando boa parte da íris estava dentro dos cantos interno ou externo, limitação já relatada na Seção 4.1.3. Mesmo com essa limitação prejudicando a precisão da detecção nos vídeos 22 e 23, de forma geral, a base de vídeos com pacientes saudáveis apresentou resultados consistentes. Os valores de diferença percentual de velocidade apresentados nessa seção são usados para definir um limiar que diferencie olhos que apresentam a paralisia de olhos saudáveis, a ser apresentado na Seção 5.4.

Além dos valores de velocidade média para cada olho e o percentual de diferença entre eles, também é possível analisar o comportamento dos olhos do paciente através da trajetória de movimentação dos olhos durante o vídeo. As Figuras 38 (a) e (b) apresentam a trajetória de movimentação dos olhos de dois pacientes. Em (a) o paciente apresenta a paralisia do sexto nervo no olho esquerdo, e em (b) o paciente tem ambos os olhos saudáveis.

Figura 38 – Trajetória do deslocamento dos olhos nos vídeos 3 (paciente apresenta paralisia) e 26 (ambos os olhos são saudáveis).



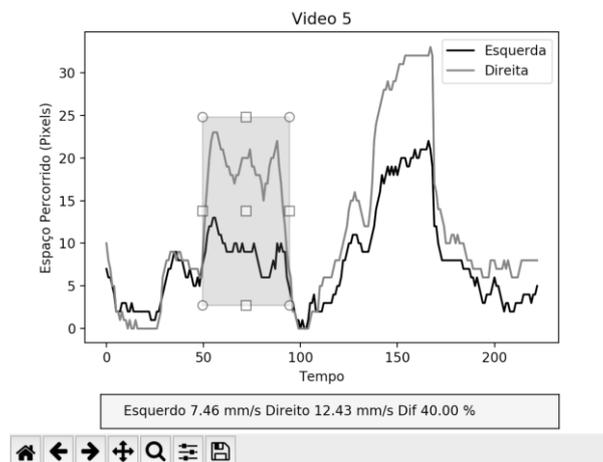
(a) Vídeo 3.

(b) Vídeo 26.

Ao comparar o espaço percorrido pelo olho com paralisia com o olho saudável em (a), percebe-se que o espaço percorrido pelo olho esquerdo (com paralisia) é de aproximadamente 40 *pixels*, enquanto que o olho direito (saudável) chega a 80 *pixels*. Ou seja, o deslocamento do olho com paralisia foi consideravelmente menor que o olho saudável. Em contrapartida, ao analisar o espaço percorrido por dois olhos saudáveis em (b), percebe-se que suas trajetórias são bem similares, e a diferença entre o deslocamento é pequena. As figuras com as trajetórias dos demais vídeos estão presentes no Apêndice A.

Com o registro da trajetória da movimentação dos olhos, é possível analisar partes específicas do vídeo, como partes em que a diferença entre os olhos é acentuada, ou em vídeos onde a detecção não funcionou bem no vídeo inteiro, possibilitando selecionar pedaços do vídeo em que o método detectou bem o centro dos olhos. Como exemplo, a Figura 39 apresenta a trajetória de movimentação do paciente do vídeo 5.

Figura 39 – Exemplo de uso da ferramenta desenvolvida para análise de partes da trajetória de movimentação dos olhos, referente ao vídeo 5.



Como já explicado anteriormente, em algumas partes deste vídeo o método teve problemas em detectar corretamente o centro do olho parético, o que resultou numa diferença pequena (16,67%) entre as velocidades do olho parético e saudável. Entretanto, ao selecionar partes dos vídeos em que o método funcionou melhor, percebe-se que a diferença é bem mais expressiva, chegando a 40% como mostra a Figura 39. A ferramenta apresentada nesta figura permite explorar partes específicas do vídeo, mas seu uso é facultativo.

5.4 Diagnóstico da Paralisia do Sexto Nervo

Para auxiliar especialistas no diagnóstico da paralisia do sexto nervo, é preciso definir um valor base para a diferença percentual entre velocidades, de forma que ajude a distinguir olhos que apresentam a paralisia de olhos saudáveis. Para isso, calcula-se um limiar a partir dos valores de diferença percentual apresentados na Tabela 10, referentes apenas aos pacientes que apresentam a paralisia.

Usa-se a validação cruzada *k-fold* para o cálculo do limiar, dividindo-se o conjunto de dados em k grupos ou *folds*, onde $k - 1$ grupos são usados para treino e o grupo restante é usado para teste (STONE, 1974). Esse processo é repetido até que cada grupo tenha sido usado para teste. Neste contexto, o treino consiste em calcular a média do percentual de diferença presente nos $k - 1$ grupos, resultando em um limiar, que é testado no grupo restante e em todo o conjunto dos pacientes saudáveis, apresentados na Tabela 11. Durante o teste, se o valor da diferença entre as velocidades dos olhos de um dado paciente for maior que o limiar, classifica-se o paciente como “apresenta a paralisia” no olho de menor velocidade; caso a diferença seja menor que o limiar, classifica-se o paciente como “saudável”. A fim de garantir a corretude, exclui-se do cálculo do limiar os casos em que o método errou, como nos vídeos 1 e 7 em que a menor velocidade foi atribuída aos olhos saudáveis. Entretanto, ambos os casos são usados no teste, sendo tratados como falsos negativos.

Para validação cruzada, dividiu-se o conjunto de dados em 5 grupos mutuamente exclusivos, onde 3 grupos continham 3 pacientes, e dois grupos continham 2. Os pacientes que fazem parte de cada grupo foram escolhidos de forma aleatória. Durante o teste de cada um dos k (5) limiares obtidos, calcula-se as métricas Sensibilidade, Especificidade, Acurácia e Índice *Kappa*, todas apresentadas no Capítulo 3, Seção 3.7. Como resultado final do teste usando validação cruzada, calcula-se a média de cada uma dessas métricas, enquanto que o limiar final será aquele que apresentou os melhores resultados, levando-se em conta todas as métricas. A Tabela 12 mostra os resultados obtidos por cada *fold*.

Dentre os 5 limiares obtidos, o limiar do *fold* 3 foi o que apresentou o melhor resultado em 3 das 4 métricas apresentadas. Somente a métrica Sensibilidade que apresentou resultado melhor ao usar o limiar do *fold* 4. Portanto, o limiar final escolhido será o do *fold* 3, com valor **19,65%**. Este valor está bem próximo do limite inferior estipulado por Metz (1976), de 20%, o que indica que o limiar escolhido está adequado ao problema.

Tabela 12 – Resultados obtidos com a validação cruzada usando 5 grupos.

Conjunto de Teste	Limiar	Sensibilidade	Especificidade	Acurácia	Kappa
<i>Fold 1</i> + Saudáveis	19,02%	66,66%	94,73%	90,9%	0,907
<i>Fold 2</i> + Saudáveis	22,45%	50%	100%	95,23%	0,952
<i>Fold 3</i> + Saudáveis	19,65%	66,66%	100%	95,45%	0,954
<i>Fold 4</i> + Saudáveis	18,34%	100%	94,73%	95,23%	0,951
<i>Fold 5</i> + Saudáveis	17,14%	66,66%	89,47%	86,36%	0,861
Média		70%	95,78%	92,64%	0,925

Portanto, se ao aplicar a metodologia aqui apresentada, o percentual de diferença da velocidade média entre dois olhos for maior que **19,65%**, classifica-se o paciente como doente, apresentando a paralisia do sexto nervo óptico no olho de menor velocidade média. Caso contrário, o paciente possui ambos os olhos saudáveis. O resultado final da classificação do método proposto ao ser aplicado nos 32 vídeos da base é apresentado na Tabela 13, bem como o limiar final proposto pelo método.

Tabela 13 – Resultado final do diagnóstico dos pacientes em doentes (apresentam a paralisia do sexto nervo óptico) e saudáveis.

	Limiar	Sensibilidade	Especificidade	Acurácia	Índice <i>Kappa</i>
Método Proposto	19,65%	70%	95,78%	92,64%	0,925

De forma geral, os resultados obtidos nesta dissertação são promissores e estão de acordo com as afirmações dos trabalhos base Metz (1976) e Rosenbaum e Santiago (1999), que também usaram a velocidade para avaliar a paralisia do sexto nervo óptico e verificaram que olhos com paralisia se movem de forma mais lenta que olhos saudáveis. Entretanto, estes trabalhos usaram a velocidade sacádica como medida de avaliação, cuja obtenção depende de aparelhos sofisticados e custosos. Nesta dissertação, apesar de se usar a velocidade média como medida de avaliação, cuja obtenção se deu a partir de vídeos gravados com câmeras comuns, verificou-se que a diferença se manteve. Após testes, verificou-se que **92,64%** dos pacientes foram classificados corretamente usando a metodologia desenvolvida. O valor de **0,925** do índice *Kappa*, de acordo com a Tabela 3, indica que os resultados obtidos foram excelentes, e que o método desenvolvido é altamente confiável, o que sugere sua aplicação clínica. Por fim, o limiar de **19,65%** obtido neste trabalho, junto com a velocidade de movimentação de cada olho e o percentual de diferença entre elas podem guiar os oftalmologistas em sua análise, ajudando-os a diferenciar olhos saudáveis de olhos com paralisia de forma acessível e não invasiva.

6 CONCLUSÃO

A paralisia do sexto nervo óptico é uma disfunção rara que impede que alguns dos músculos que controlam o movimento dos olhos funcionem adequadamente, causando dor de cabeça, enxaqueca, visão turva, vertigem e diplopia. O diagnóstico deve ser feito cedo, pois quando tratada, a paralisia geralmente desaparece sem deixar sequelas. O processo de diagnóstico dessa paralisia envolve a realização de um exame muito invasivo, e de difícil aplicação em crianças. A outra forma de diagnóstico depende de equipamentos caros e sofisticados, presentes apenas em grandes laboratórios de pesquisa.

A fim de auxiliar oftalmologistas com um método não invasivo, acessível e de baixo custo, este trabalho apresentou uma metodologia computacional para auxílio na identificação da paralisia do sexto nervo óptico através do processamento de vídeos. Por quantificar a velocidade média com que os olhos se movem, esta metodologia fornece dados que podem guiar os oftalmologistas em sua análise.

Para isso, construiu-se uma base de vídeos com exemplos de pessoas que apresentam a paralisia e pessoas com ambos os olhos saudáveis. Os vídeos foram adquiridos com câmeras comuns, e neles, pede-se que os pacientes movam os olhos de acordo com o movimento persecutório. O método desenvolvido é aplicado sobre esses vídeos, e é composto por 4 etapas principais.

Na primeira etapa da metodologia, localiza-se a região dos olhos através da rede neural YOLOv3, que apresentou 99,02% de precisão na execução desta tarefa. Em seguida, detecta-se o centro dos olhos através do centro da região informada pela YOLOv3, cuja aplicação apresentou erro médio de 5,09%. Essas duas etapas se repetem ao longo do vídeo, a fim de registrar trajetória da movimentação dos olhos. Ao final do processamento do vídeo, é possível calcular a velocidade média da movimentação de cada olho e o percentual de diferença entre elas, seguido pela classificação do paciente, sendo esta a quarta etapa da metodologia.

Ao aplicar o método desenvolvido sobre os 32 vídeos da base de dados, chegou-se ao limiar de 19,65% de diferença mínima entre as velocidades dos olhos para se considerar um paciente como doente (apresenta a paralisia do sexto nervo óptico no olho de menor velocidade média). Este limiar pode auxiliar o oftalmologista a diagnosticar se há paralisia ou não, visto que olhos que apresentam a paralisia se movem de forma mais lenta que olhos

saudáveis. Os testes realizados após a aplicação do método mostraram que 92,64% dos pacientes foram classificados corretamente. O método funcionou especialmente bem nos pacientes saudáveis, com especificidade de 95,78%. Entretanto, a sensibilidade da classificação apresenta margem para melhora em relação aos pacientes que apresentam a paralisia, cujo resultado foi 70%. Em adição, o valor de 0,925 do índice *Kappa* ressalta a confiabilidade dos resultados obtidos e sugere a aplicação clínica do método. Assim, este trabalho fornece dados que podem guiar o oftalmologista no processo de diagnóstico, servindo como uma opinião complementar. Vale ressaltar que apenas um médico especializado pode fornecer um diagnóstico definitivo ao paciente.

De modo geral, os resultados alcançados neste trabalho são promissores e trazem perspectivas favoráveis à metodologia desenvolvida, possibilitando que futuramente, ela seja incorporada em um CAD para auxiliar no diagnóstico da paralisia do sexto nervo óptico. Dentre as principais contribuições deste trabalho lista-se:

- Desenvolvimento de uma metodologia computacional automática, acessível e de baixo custo para auxílio no diagnóstico da paralisia do sexto nervo óptico através de vídeos gravados com equipamentos simples (câmeras fotográficas e câmeras de *smartphones*);
- Construção de uma base de vídeos com exemplos de pacientes que apresentam a paralisia do sexto nervo óptico;
- Construção de uma base com 18.344 imagens de olhos com a marcação da posição da pupila.

6.1 Trabalhos Futuros

Como trabalhos futuros, sugere-se: (1) aumentar a quantidade de vídeos na base de dados, em especial o de pessoas com a paralisia, a fim de obter percentuais de diferença mais representativos. (2) Melhorar a detecção do centro dos olhos em caso de pessoas com desvios de versão. Para isso, sugere-se retreinar a rede neural com mais exemplos de olhos que apresentem esse problema. (3) Aumentar a base de imagens de treino da rede neural, para construir um modelo de aprendizado mais robusto, incluindo imagens de olhos em iluminações variadas. (4) Testar outras abordagens para a detecção do centro do olho, como segmentar a esclera e detectar os cantos dos olhos para reduzir a região

de busca. (5) Testar outras arquiteturas de redes neurais para a localização dos olhos, principalmente as mais leves que podem ser incorporadas em dispositivos móveis. Por fim, em vez de fornecer valores que ajudem o especialista no diagnóstico, pode-se (6) testar o uso de modelos de aprendizagem profunda para a classificação automática da paralisia.

6.2 Produções Científicas

Como resultado desta pesquisa, um artigo científico com resultados preliminares foi publicado, conforme mostra a Tabela 14.

Tabela 14 – Artigo publicado que possui relação com o método proposto.

Anais	Título	Qualis
SBCAS 2019	Identificando a Paralisia do Sexto Nervo Ótico em Vídeos Através de Redes Neurais Convolucionais e Curvatura <i>Isophotes</i> - Menção Honrosa Artigo Completo	B4

REFERÊNCIAS

ABADI, M.; BARHAM, P.; CHEN, J.; CHEN, Z.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; IRVING, G.; ISARD, M. et al. Tensorflow: A system for large-scale machine learning. In: **12th Symposium on Operating Systems Design and Implementation ({OSDI} 16)**. [S.l.: s.n.], 2016. p. 265–283.

ALGORITHMIA. **Introduction to Loss Functions**. 2018. <<https://algorithmia.com/blog/introduction-to-loss-functions>>. "Acesso em 07/04/2020".

ALMEIDA, J. D. S. D.; SILVA, A. C.; TEIXEIRA, J. A. M.; PAIVA, A. C.; GATTASS, M. Computer-aided methodology for syndromic strabismus diagnosis. **Journal of digital imaging**, Springer, v. 28, n. 4, p. 462–473, 2015.

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>>.

ALVARADO, J. **Optometría Ortóptica**. 2013. <<https://www.slideshare.net/drafts10/campo-de-accion-26010427/10>>. "Acesso em 13/03/2020".

ARAÚJO, J. D. L.; SOUZA, J. C.; NETO, O. P. S.; SOUSA, J. A. de; ALMEIDA, J. D. S. de; PAIVA, A. C. de; SILVA, A. C.; JUNIOR, G. B.; GATTASS, M. Glaucoma diagnosis in fundus eye images using diversity indexes. **Multimedia Tools and Applications**, Springer, v. 78, n. 10, p. 12987–13004, 2019.

BELLIDO, J. C.; PIETRO, G. D.; SANNINO, G. A control cycle for the automatic assisted positioning of auscultation sensors. In: . [S.l.: s.n.], 2016. p. 53–60.

BICAS, H. E. Oculomotricidade e seus fundamentos. **Arquivos Brasileiros de Oftalmologia**, SciELO Brasil, v. 66, n. 5, p. 687–700, 2003.

BIOPAC. **Electrooculography**. 2016. <<https://www.youtube.com/watch?v=-QXGiZBDkUw>>. "Acesso em 29/03/2020".

BUCCI, M. P.; KAPOULA, Z.; YANG, Q.; ROUSSAT, B.; BRÉMOND-GIGNAC, D. Binocular coordination of saccades in children with strabismus before and after surgery. **Investigative ophthalmology & visual science**, The Association for Research in Vision and Ophthalmology, v. 43, n. 4, p. 1040–1047, 2002.

CAMPADELLI, P.; LANZAROTTI, R.; LIPORI, G. Eye localization: a survey. **NATO SECURITY THROUGH SCIENCE SERIES E HUMAN AND SOCIETAL DYNAMICS**, IOS PRESS, v. 18, p. 234, 2007.

CASTLEMAN, K. R. **Digital image processing**. [S.l.]: Pearson, 1996.

CBO. **Um olhar sobre o Brasil. Censo Oftalmológico 2014**. 2014. <http://www.cbo.net.br/novo/publicacoes/Olhar_sobre_o_Brasil.pdf>. "Acesso em 15/01/2020".

CHOLLET, F. et al. **Keras**. [S.l.]: GitHub, 2015. <<https://github.com/keras-team/keras>>. "Acesso em 23/01/2019".

- CLARO, M.; SANTOS, L.; SILVA, W.; ARAÚJO, F.; MOURA, N.; MACEDO, A. Automatic glaucoma detection based on optic disc segmentation and texture feature extraction. **CLEI Electronic Journal**, Centro Latinoamericano de Estudios en Informática, v. 19, n. 2, p. 5–5, 2016.
- COELHO, R. P.; WEISSHEIMER, J.; ROMÃO, E.; CRUZ, A. A. V. Comparação entre a dor provocada pela facoemulsificação com anestesia tópica e a pela infiltração peribulbar sem sedação. **Arquivos Brasileiros de Oftalmologia**, SciELO Brasil, v. 68, n. 1, p. 45–48, 2005.
- COHEN, J. A coefficient of agreement for nominal scales. **Educational and psychological measurement**, Sage Publications Sage CA: Thousand Oaks, CA, v. 20, n. 1, p. 37–46, 1960.
- CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, v. 20, n. 3, p. 273–297, Sep 1995. ISSN 1573-0565. Disponível em: <<https://doi.org/10.1007/BF00994018>>.
- CRUZ, L. B. da; SOUZA, J. C.; SOUSA, J. A. de; SANTOS, A. M.; PAIVA, A. C. de; ALMEIDA, J. D. S. de; SILVA, A. C.; JUNIOR, G. B.; GATTASS, M. Interferometer eye image classification for dry eye categorization using phylogenetic diversity indexes for texture analysis. **Computer Methods and Programs in Biomedicine**, Elsevier, v. 188, p. 105269, 2020.
- CUI, J.; CHEN, F.; SHI, D.; LIU, L. Eye detection with faster r-cnn. In: **INSTICC. Proceedings of the International Conference on Advances in Computer Technology, Information Science and Communications - Volume 1: CTISC.**, [S.l.]: SciTePress, 2019. p. 111–116. ISBN 978-989-758-357-5.
- CULLEN, K.; SADEGHI, S. Vestibular system. **Scholarpedia**, v. 3, n. 1, p. 3013, 2008.
- CURI, R. L. N.; COSTA, I. C. B. de O.; BARROSO, T. G. M. Paralisia do vi nervo (abducente). **Rev Bras Oftalmol**, SciELO Brasil, v. 72, n. 1, p. 59–69, 2013.
- DAGAR, M.; MISHRA, N.; RANI, A.; AGARWAL, S.; YADAV, J. Performance comparison of hampel and median filters in removing deep brain stimulation artifact. In: **Innovations in Computational Intelligence**. [S.l.]: Springer, 2018. p. 17–28.
- DAI, W.; YANG, Q.; XUE, G.-R.; YU, Y. Boosting for transfer learning. In: **Proceedings of the 24th international conference on Machine learning**. [S.l.: s.n.], 2007. p. 193–200.
- DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: **IEEE. 2009 IEEE conference on computer vision and pattern recognition**. [S.l.], 2009. p. 248–255.
- DOI, K. Computer-aided diagnosis in medical imaging: historical review, current status and future potential. **Computerized medical imaging and graphics**, Elsevier, v. 31, n. 4-5, p. 198–211, 2007.
- DUCHOWSKI, A. T. Eye tracking methodology. **Theory and practice**, Springer, v. 328, n. 614, 2007.

- EASTON, R. L. *Fundamentals of digital image processing*. November, 2010.
- EHRENHAU, M. P. **Abducens Nerve Palsy (Sixth Cranial Nerve Palsy)**. 2018. <<https://emedicine.medscape.com/article/1198383-overview>>. "Acesso em 25/02/2019".
- EMBALANCE. **Saccades**. 2016. <https://www.youtube.com/watch?v=D_y56PQvZZA>. "Acesso em 17/03/2020".
- GANAN, M. M.; CAOVILO, H. H.; FREITAS, F. Eletronistagmografia versus videonistagmografia. **Brazilian Journal of Otorhinolaryngology**, scielo, v. 76, p. 399 – 403, 06 2010. ISSN 1808-8694. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1808-86942010000300021&nrm=iso>.
- GARD. **Sixth nerve palsy**. 2015. <<https://rarediseases.info.nih.gov/diseases/9482/sixth-nerve-palsy>>. "Acesso em 25/02/2019".
- GONZALEZ, R.; WOODS, R.; COAUT, W. **Digital Image Processing**. Prentice Hall, 2002. ISBN 9780201180756. Disponível em: <<https://books.google.com.br/books?id=738oAQAAMAAJ>>.
- 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>>.
- GRAUMAN, K.; BETKE, M.; GIPS, J.; BRADSKI, G. R. Communication via eye blinks-detection and duration analysis in real time. In: IEEE. **Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001**. [S.l.], 2001. v. 1, p. I-I.
- GRAY, H.; CARTER, H. V. **Anatomy of the human body**. [S.l.]: Lea & Febiger, 1918.
- GUEDES, V. M. P. **Anestesia tópică como alternativa segura e eficaz em esclerectomia profunda**. 2007. <<http://bit.ly/AnestesiaTopica>>. "Acesso em 09/03/2020".
- HAGEMANN, L. F. **Cirurgia de Catarata sob anestesia tópică**. 2020. <<https://hob.med.br/cirurgia-de-catarata-sob-anestesia-topica/>>. "Acesso em 05/03/2020".
- HALLIDAY, D.; RESNICK, R.; WALKER, J. **Fundamentals of physics**. 10. ed. [S.l.]: Wiley and Sons, 2013. ISBN 978-1-118-54787-8.
- HALLINAN, P. W. Recognizing human eyes. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Geometric Methods in Computer Vision**. [S.l.], 1991. v. 1570, p. 214–226.
- HAMPEL, F. R. The influence curve and its role in robust estimation. **Journal of the american statistical association**, Taylor & Francis, v. 69, n. 346, p. 383–393, 1974.
- HANSEN, D. W.; PECE, A. E. Eye tracking in the wild. **Computer Vision and Image Understanding**, Elsevier, v. 98, n. 1, p. 155–181, 2005.

- HARITHA, Z. S. A.; RAVEENA, P.; ARUN, K.; NITHYA, P.; BALAN, M. V.; KRISHNAN, S. Eye tracking system using isophote eye center detection with blink perception. In: IEEE. **2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPE5)**. [S.l.], 2016. p. 467–471.
- HAYKIN, S. **Kalman filtering and neural networks**. [S.l.]: John Wiley & Sons, 2004. v. 47.
- HAYKIN, S. **Neural Networks and Learning Machines**. [S.l.]: Pearson Education India, 2010.
- HUANG, J.; WECHSLER, H. Eye detection using optimal wavelet packets and radial basis functions (rbfs). **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 13, n. 07, p. 1009–1025, 1999.
- HUANG, W.; MARIANI, R. Face detection and precise eyes location. In: IEEE. **Proceedings 15th International Conference on Pattern Recognition. ICPR-2000**. [S.l.], 2000. v. 4, p. 722–727.
- ITSEEZ. **Open Source Computer Vision Library**. 2015. <<https://github.com/itseez/opencv>>.
- IVAMOTO, H. **Movimentos Oculares**. 2014. <http://www.actamedica.org.br/publico/noticia.php?codigo=327&cod_menu=327>. "Acesso em 30/12/2019".
- IVINS, J. P.; PORRILL, J. A deformable model of the human iris for measuring small three-dimensional eye movements. **Machine Vision and Applications**, Springer, v. 11, n. 1, p. 42–51, 1998.
- 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.
- KASINSKI, A.; SCHMIDT, A. The architecture and performance of the face and eyes detection system based on the haar cascade classifiers. **Pattern Analysis and Applications**, Springer, v. 13, n. 2, p. 197–211, 2010.
- KHAIROSFAIZAL, W. W. M.; NOR'AINI, A. Eyes detection in facial images using circular hough transform. In: IEEE. **2009 5th International Colloquium on Signal Processing & Its Applications**. [S.l.], 2009. p. 238–242.
- KIMME, C.; BALLARD, D.; SKLANSKY, J. Finding circles by an array of accumulators. **Communications of the ACM**, ACM New York, NY, USA, v. 18, n. 2, p. 120–122, 1975.
- KOLB, H. Gross anatomy of the eye. In: **Webvision: The Organization of the Retina and Visual System [Internet]**. [S.l.]: University of Utah Health Sciences Center, 2007.
- KOO, T. K.; LI, M. Y. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. **Journal of chiropractic medicine**, Elsevier, v. 15, n. 2, p. 155–163, 2016.

- KUMAR, V.; BHOOSHAN, S.; SOOD, A.; SHAHI, R.; MENDIRATTA, S. A novel technique for automated red eye detection and correction. In: IEEE. **2009 XXII International Symposium on Information, Communication and Automation Technologies**. [S.l.], 2009. p. 1–5.
- LAND, M. F.; FURNEAUX, S. The knowledge base of the oculomotor system. **Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences**, The Royal Society, v. 352, n. 1358, p. 1231–1239, 1997.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015.
- LICHTENAUER, J.; HENDRIKS, E.; REINDERS, M. Isophote properties as features for object detection. In: IEEE. **Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on**. [S.l.], 2005. v. 2, p. 649–654.
- LIM, L.; ROSENBAUM, A. L.; DEMER, J. L. Saccadic velocity analysis in patients with divergence paralysis. **Journal of pediatric ophthalmology and strabismus**, SLACK Incorporated, v. 32, n. 2, p. 76–81, 1995.
- LIMA, M. **Movimento dos Olhos**. 2015. <<http://limatreinamento.blogspot.com/2015/06/movimento-dos-olhos.html>>. "Acesso em 13/03/2020".
- LISBERGER, S. G.; MORRIS, E.; TYCHSEN, L. Visual motion processing and sensory-motor integration for smooth pursuit eye movements. **Annual review of neuroscience**, Annual Reviews 4139 El Camino Way, PO Box 10139, Palo Alto, CA 94303-0139, USA, v. 10, n. 1, p. 97–129, 1987.
- LIU, W.; WANG, Z.; LIU, X.; ZENG, N.; LIU, Y.; ALSAADI, F. E. A survey of deep neural network architectures and their applications. **Neurocomputing**, Elsevier, v. 234, p. 11–26, 2017.
- MACHADO, N. A. G.; STRINI, P. J. S. A.; BRANCO, C. A.; BORGES, R. N.; MAZZETTO, M. de O.; NETO, A. J. F.; FONSECA, R. B. Eletromiografia aplicada à desordens temporomandibulares. **Revista Odontológica do Brasil Central**, v. 19, n. 51, 2010.
- MARCO, T. D.; CAZZATO, D.; LEO, M.; DISTANTE, C. Randomized circle detection with isophotes curvature analysis. **Pattern Recognition**, Elsevier, v. 48, n. 2, p. 411–421, 2015.
- MARTINSEN, O. G.; GRIMNES, S. **Bioimpedance and bioelectricity basics**. [S.l.]: Academic press, 2011.
- MCCONNELL, R. K. **Method of and apparatus for pattern recognition**. [S.l.]: Google Patents, 1986. US Patent 4,567,610.
- MCHUGH, M. L. Interrater reliability: the kappa statistic. **Biochemia medica: Biochemia medica**, Medicinska naklada, v. 22, n. 3, p. 276–282, 2012.

- MELLEN, P.; BILYK, J. **July 2018 Wills Eye Resident Case Series - Diagnosis and Discussion**. 2018. <<https://www.reviewofophthalmology.com/article/july-2018-wills-eye-resident-case-series-diagnosis-and-discussion>>. "Acesso em 22/02/2020".
- METZ, H. S. Forced duction, active force generation, and saccadic velocity tests. **International ophthalmology clinics**, v. 16, n. 3, p. 47–73, 1976.
- METZ, H. S. Saccadic velocity measurements in strabismus. **Transactions of the American Ophthalmological Society**, American Ophthalmological Society, v. 81, p. 630, 1983.
- MONZO, D.; ALBIOL, A.; SASTRE, J.; ALBIOL, A. Precise eye localization using hog descriptors. **Machine Vision and Applications**, Springer, v. 22, n. 3, p. 471–480, 2011.
- MORGENLANDER, J. **Sports Concussion Exam (8 of 9): Pursuits, Saccades, and Convergence**. 2017. <<https://www.youtube.com/channel/UCx8EDkz1OeztrTQfqKALrnw/videos>>. "Acesso em 17/03/2020".
- NI, Y.; SUN, B. A remote free-head pupillometry based on deep learning and binocular system. **IEEE Sensors Journal**, IEEE, v. 19, n. 6, p. 2362–2369, 2018.
- OMS. **Blindness and vision impairment**. 2019. <<https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment>>. "Acesso em 15/01/2020".
- OTTAIANO, J.; ÁVILA, M.; UMBELINO, C.; TALEB, A. **As condições de saúde ocular no Brasil 2019**. 2019. <http://www.cbo.com.br/novo/publicacoes/condicoes_saude_ocular_brasil2019.pdf>. "Acesso em 15/01/2020".
- OUABIDA, E.; ESSADIQUE, A.; BOUZID, A. Vander lugt correlator based active contours for iris segmentation and tracking. **Expert Systems with Applications**, Elsevier, v. 71, p. 383–395, 2017.
- PEARSON, R. K.; NEUVO, Y.; ASTOLA, J.; GABBOUJ, M. Generalized hampel filters. **EURASIP Journal on Advances in Signal Processing**, SpringerOpen, v. 2016, n. 1, p. 1–18, 2016.
- PENTLAND, A.; MOGHADDAM, B.; STARNER, T. et al. View-based and modular eigenspaces for face recognition. **IEEE Conference on Computer Vision Pattern Recognition**, Vision and Modeling Group, Media Laboratory, Massachusetts Institute of Technology, n. 245, 1994.
- PINHEIRO, J. F. **Avaliação automática de versões oculares baseada em imagens**. Dissertação (Mestrado) — Universidade Federal do Maranhão, 2018.
- POESCHL, T. Detecting surface irregularities using isophotes. **Computer Aided Geometric Design**, Elsevier, v. 1, n. 2, p. 163–168, 1984.
- POWERS, D. M. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. Bioinfo Publications, 2011.

QUEIROZ, J.; GOMES, H. Introdução ao processamento digital de imagens. **RITA**, v. 13, p. 11–42, 01 2006.

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)**, 2015. Disponível em: <<http://arxiv.org/abs/1506.02640>>.

REDMON, J.; FARHADI, A. **Yolov3: An incremental improvement**. [S.l.], 2018.

REMYINGTON, L. A. **Clinical anatomy of the visual system**. [S.l.]: Elsevier Health Sciences, 2005.

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.

ROSA, A. **Descubra as principais partes do olho e as suas funções**. 2019. <<https://retinapro.com.br/blog/principais-partes-do-olho/>>. "Acesso em 10/01/2020".

ROSENBAUM, A. L.; SANTIAGO, A. P. Sixth cranial nerve palsy. In: **Clinical strabismus management: principles and surgical techniques**. [S.l.]: David Hunter, 1999. cap. 19, p. 259–261.

ROSSUM, G. V.; DRAKE, F. L. **Python 3 Reference Manual**. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697.

ROSSUM, G. V.; JR, F. L. D. **Python tutorial**. [S.l.]: Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995.

RUBIN, M. **Paralisia do sexto nervo craniano (nervo motor ocular externo)**. 2018. <<http://bit.ly/paralisia-do-sexto-nervo>>. "Acesso em 14/12/2018".

SARAVANAN, C. Color image to grayscale image conversion. In: IEEE. **2010 Second International Conference on Computer Engineering and Applications**. [S.l.], 2010. v. 2, p. 196–199.

SAUNDERS, R. A. Incomitant vertical strabismus: treatment with posterior fixation of the inferior rectus muscle. **Archives of Ophthalmology**, American Medical Association, v. 102, n. 8, p. 1174–1177, 1984.

SHERMAN, C. **The Senses, Vision**. [S.l.]: Dana Foundation, 2019. <<https://www.dana.org/article/the-senses-vision/>>. "Acesso em 10/01/2020".

SIHOTA, R.; TANDON, R. **Parsons' Diseases of the Eye**. [S.l.]: Elsevier India, 2011.

SILVA, I. F. S. da; ALMEIDA, J. D. S. de; TEIXEIRA, J. A. M.; JUNIOR, G. B.; PAIVA, A. C. de. Segmentation of the retinal reflex in brückner test images using u-net convolutional network. In: SPRINGER. **International Conference Image Analysis and Recognition**. [S.l.], 2018. p. 679–686.

SNELL, R. S.; LEMP, M. A. **Clinical anatomy of the eye**. [S.l.]: John Wiley & Sons, 2013.

SONG, J.; CHI, Z.; LIU, J. A robust eye detection method using combined binary edge and intensity information. **Pattern Recognition**, Elsevier, v. 39, n. 6, p. 1110–1125, 2006.

SOUSA, J. A. de; PAIVA, A. C. de; ALMEIDA, J. D. S. D.; SILVA, A. C.; JUNIOR, G. B.; GATTASS, M. Texture based on geostatistic for glaucoma diagnosis from fundus eye image. **Multimedia Tools and Applications**, Springer, v. 76, n. 18, p. 19173–19190, 2017.

SRINIVASAN, P. P.; KIM, L. A.; METTU, P. S.; COUSINS, S. W.; COMER, G. M.; IZATT, J. A.; FARSIU, S. Fully automated detection of diabetic macular edema and dry age-related macular degeneration from optical coherence tomography images. **Biomedical optics express**, Optical Society of America, v. 5, n. 10, p. 3568–3577, 2014.

STONE, M. Cross-validatory choice and assessment of statistical predictions. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 36, n. 2, p. 111–133, 1974.

SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V.; ALEMI, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: **AAAI**. [S.l.: s.n.], 2017. v. 4, p. 12.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to data mining**. [S.l.]: Pearson Education India, 2016.

TAPSELL, S. **Eye Movement Terminology**. 2009. <<http://eyevideos.blogspot.com/2009/03/eye-movement-terminology.html>>. "Acesso em 30/12/2019".

TIAN, Y.-l.; KANADE, T.; COHN, J. F. Dual-state parametric eye tracking. In: **IEEE. Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)**. [S.l.], 2000. p. 110–115.

TRADE, P. **A Keras implementation of YOLOv3**. 2018. <<https://github.com/qqwwee/keras-yolo3>>. "Acesso em 30/07/2018".

TURNER, M. **The sclera and its coverings**. 2012. <<https://eyeanatomyblog.wordpress.com/2012/10/21/the-sclera-and-its-coverings/>>. "Acesso em 21/01/2020".

VALENTE, T. L. A. **Metodologia Computacional para Detecção e Diagnóstico Automáticos de Estrabismo em Vídeos Digitais utilizando o Cover Test**. Dissertação (Mestrado) — Universidade Federal do Maranhão, 2017.

VALENTI, R.; GEVERS, T. Accurate eye center location and tracking using isophote curvature. **IEEE**, 2008.

VALENTI, R.; GEVERS, T. Accurate eye center location through invariant isocentric patterns. **IEEE transactions on pattern analysis and machine intelligence**, **IEEE**, v. 34, n. 9, p. 1785–1798, 2012.

VERA-OLMOS, F.; MALPICA, N. Deconvolutional neural network for pupil detection in real-world environments. In: **SPRINGER. International Work-Conference on the Interplay Between Natural and Artificial Computation**. [S.l.], 2017. p. 223–231.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: IEEE. **Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001.** [S.l.], 2001. v. 1, p. I–I.

VUČKOVIĆ, V. Image and its matrix, matrix and its image. **Review of the National Center for Digitization**, Faculty of Mathematics, n. 12, p. 17–31, 2008.

WIERTS, R.; JANSSEN, M. J.; KINGMA, H. Measuring saccade peak velocity using a low-frequency sampling rate of 50 hz. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 55, n. 12, p. 2840–2842, 2008.

WILLOUGHBY, C. E.; PONZIN, D.; FERRARI, S.; LOBO, A.; LANDAU, K.; OMIDI, Y. Anatomy and physiology of the human eye: effects of mucopolysaccharidoses disease on structure and function—a review. **Clinical & Experimental Ophthalmology**, Wiley Online Library, v. 38, p. 2–11, 2010.

ZEISS. **Why good vision is so important.** 2017. <<https://www.zeiss.com/vision-care/int/better-vision/health-prevention/why-good-vision-is-so-important-.html/>>. "Acesso em 06/01/2020".

ZHOU, Z.-H.; GENG, X. Projection functions for eye detection. **Pattern recognition**, Elsevier, v. 37, n. 5, p. 1049–1056, 2004.

ZIFFER, A. J.; ROSENBAUM, A. L.; DEMER, J. L.; YEE, R. D. Congenital double elevator palsy: vertical saccadic velocity utilizing the scleral search coil technique. **Journal of pediatric ophthalmology and strabismus**, SLACK Incorporated, v. 29, n. 3, p. 142–149, 1992.

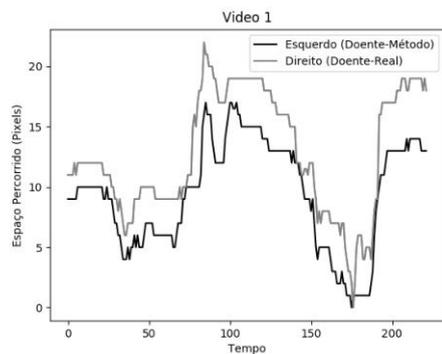
APÊNDICES

APÊNDICE A – REGISTROS

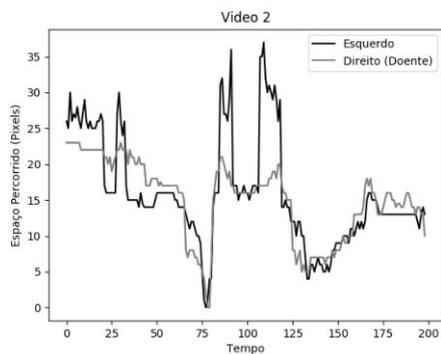
A.1 Trajetória de Movimentação dos Olhos em Vídeos

As Figuras 40 a 41 mostram a trajetória de movimentação dos olhos de pacientes que apresentam a paralisia do sexto nervo óptico, referente aos vídeos 1 a 13.

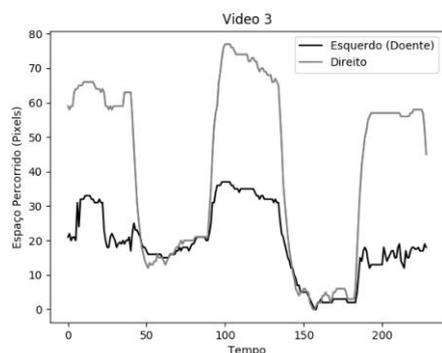
Figura 40 – Trajetória de movimentação dos olhos presentes nos vídeos 1 a 6.



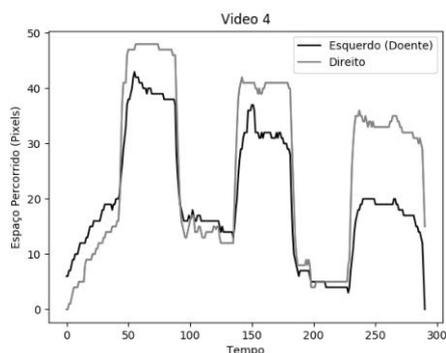
(a) Vídeo 1.



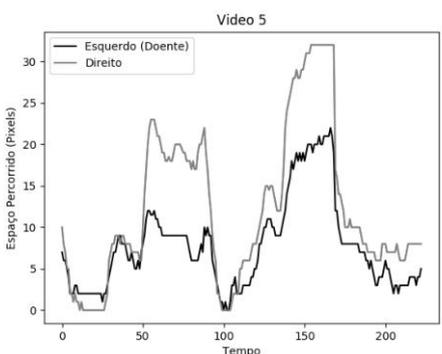
(b) Vídeo 2.



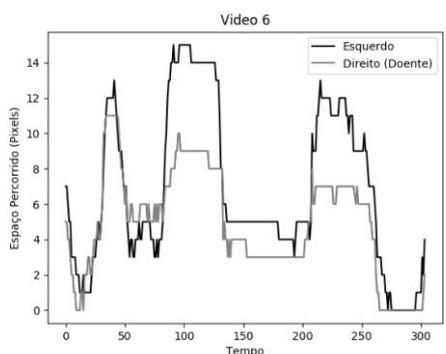
(c) Vídeo 3.



(d) Vídeo 4.

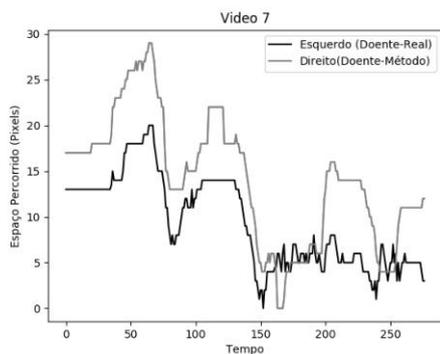


(e) Vídeo 5.

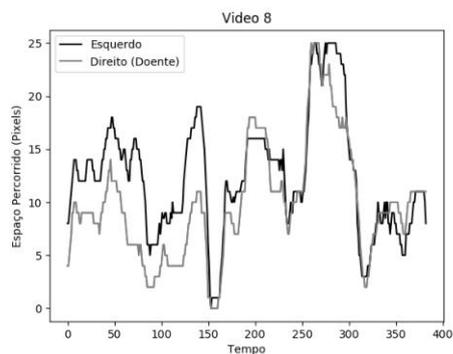


(f) Vídeo 6.

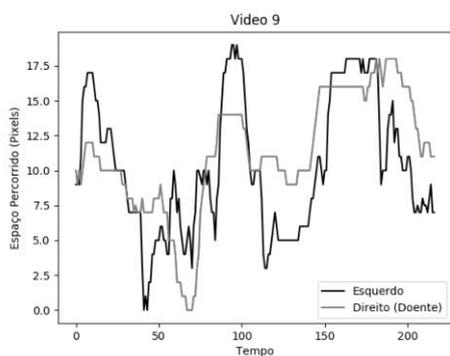
Figura 41 – Trajetória de movimentação dos olhos presentes nos vídeos 7 a 13.



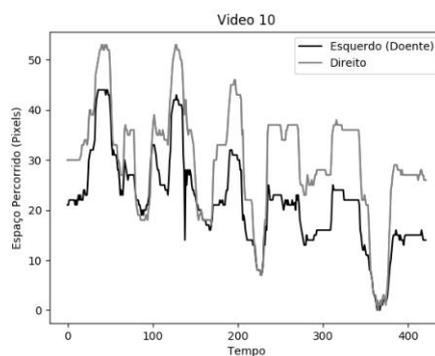
(a) Vídeo 7.



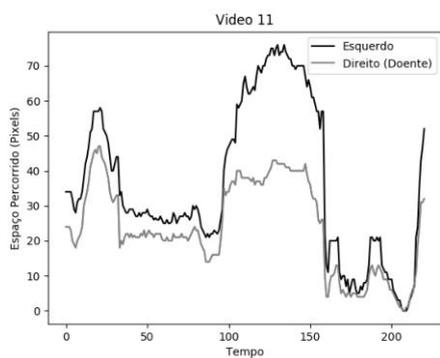
(b) Vídeo 8.



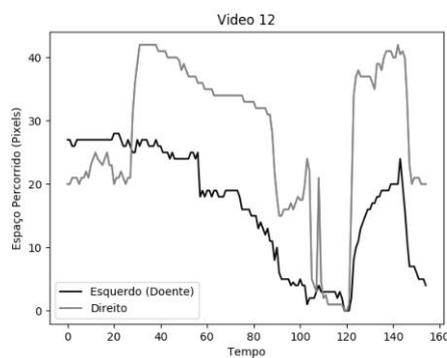
(c) Vídeo 9.



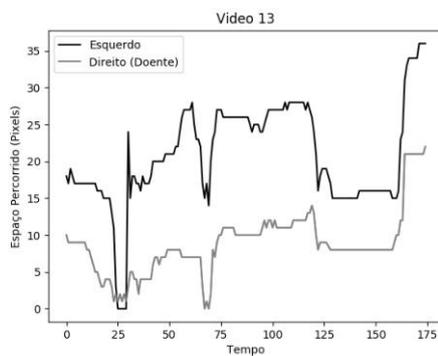
(d) Vídeo 10.



(e) Vídeo 11.



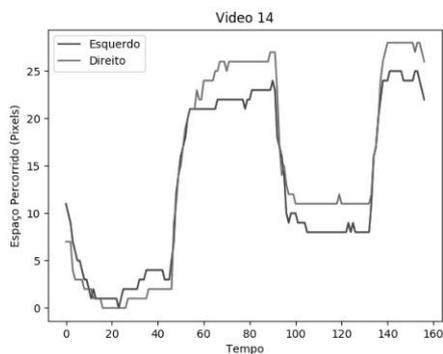
(f) Vídeo 12.



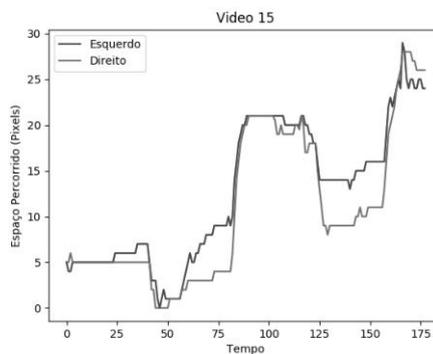
(g) Vídeo 13.

As Figuras 42 a 44 mostram a trajetória de movimentação dos olhos de pacientes que não apresentam paralisia, referente aos vídeos 14 a 32.

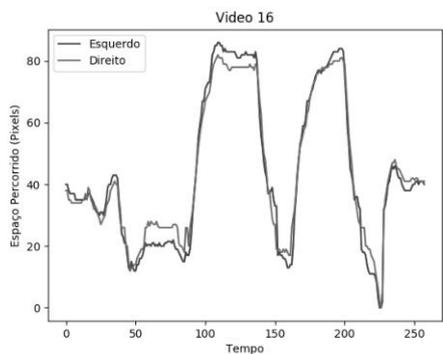
Figura 42 – Trajetória de movimentação dos olhos presentes nos vídeos 14 a 19.



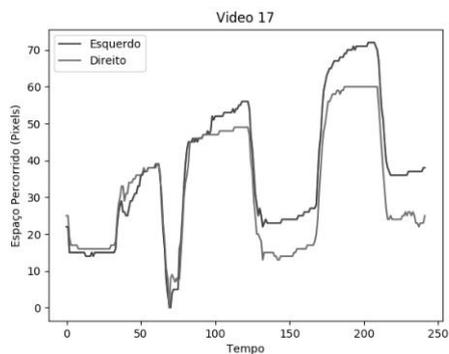
(a) Vídeo 14.



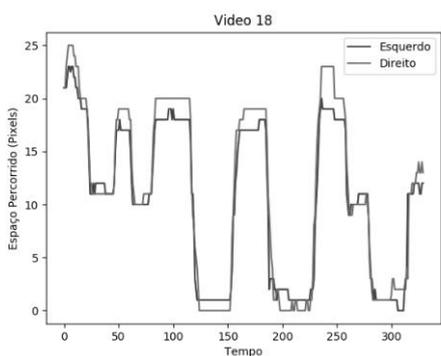
(b) Vídeo 15.



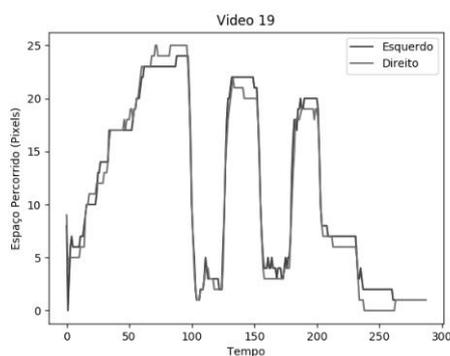
(c) Vídeo 16.



(d) Vídeo 17.

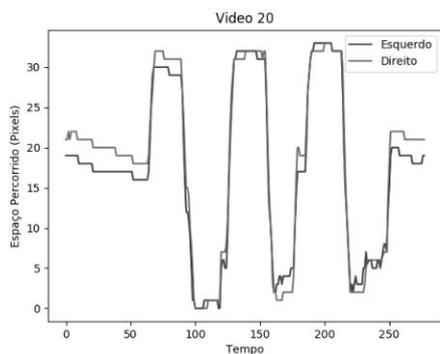


(e) Vídeo 18.

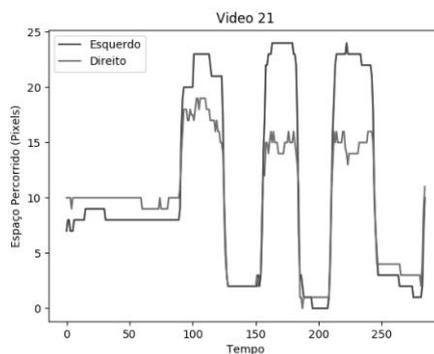


(f) Vídeo 19.

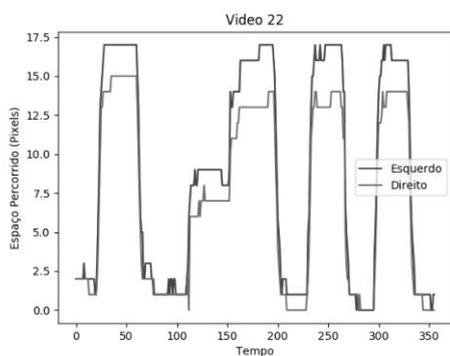
Figura 43 – Trajetória de movimentação dos olhos presentes nos vídeos 20 a 27.



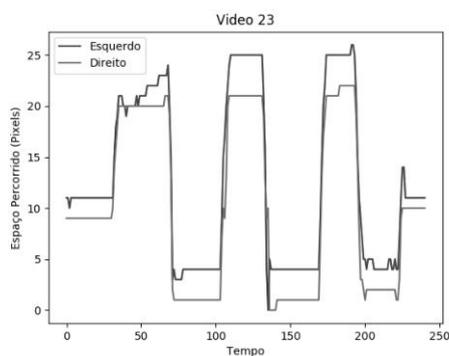
(a) Vídeo 20.



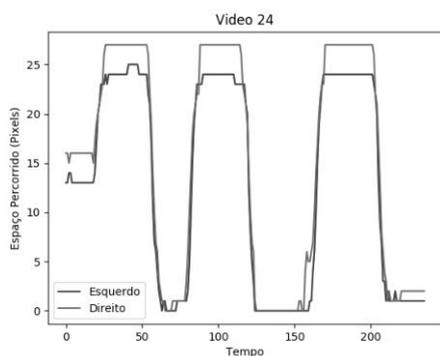
(b) Vídeo 21.



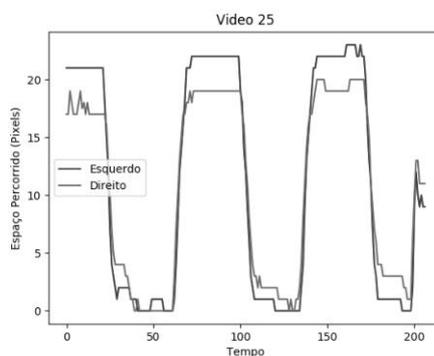
(c) Vídeo 22.



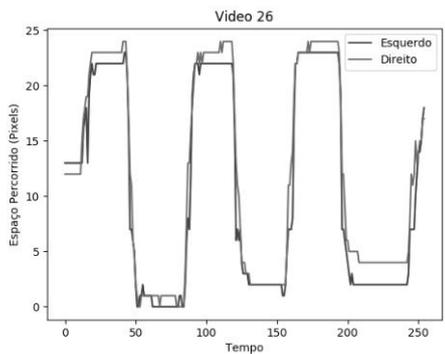
(d) Vídeo 23.



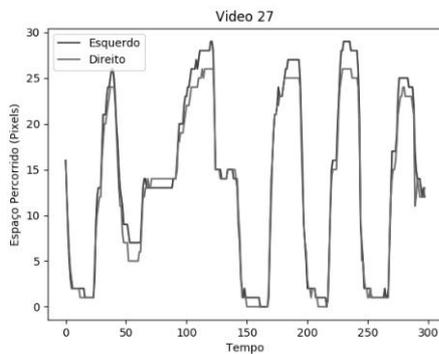
(e) Vídeo 24.



(f) Vídeo 25.

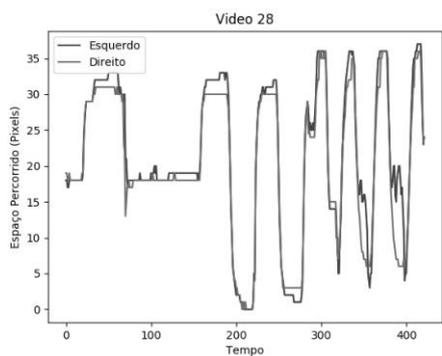


(g) Vídeo 26.

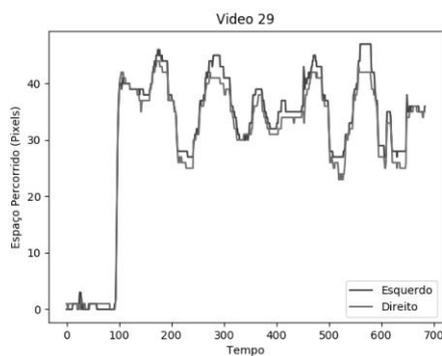


(h) Vídeo 27.

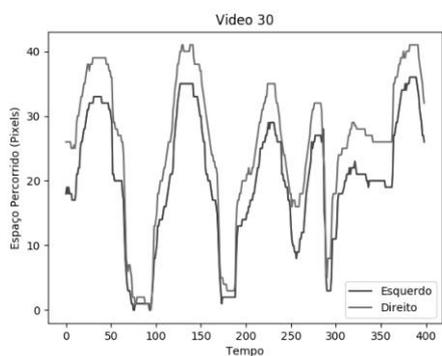
Figura 44 – Trajetória de movimentação dos olhos presentes nos vídeos 28 a 32.



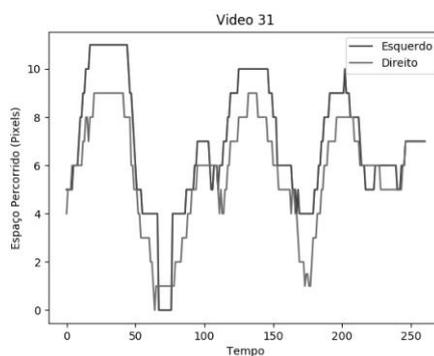
(a) Vídeo 28.



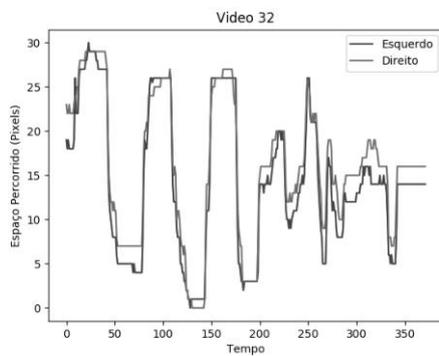
(b) Vídeo 29.



(c) Vídeo 30.



(d) Vídeo 31.



(e) Vídeo 32.