



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

SAULO ENOCK RODRIGUES FERNANDES

***DIAGNÓSTICO AUTOMÁTICO DA PARALISIA DO SEXTO NERVO
EM VÍDEOS DE MOTILIDADE OCULAR POR CLASSIFICAÇÃO DE
SÉRIES TEMPORAIS E APRENDIZADO PROFUNDO***

São Luís
2026

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

SAULO ENOCK RODRIGUES FERNANDES

**DIAGNÓSTICO AUTOMÁTICO DA PARALISIA DO SEXTO NERVO EM
VÍDEOS DE MOTILIDADE OCULAR POR CLASSIFICAÇÃO DE SÉRIES
TEMPORAIS E APRENDIZADO PROFUNDO**

SÃO LUÍS – MA

2026

SAULO ENOCK RODRIGUES FERNANDES

**DIAGNÓSTICO AUTOMÁTICO DA PARALISIA DO SEXTO NERVO EM
VÍDEOS DE MOTILIDADE OCULAR POR CLASSIFICAÇÃO DE SÉRIES
TEMPORAIS E APRENDIZADO PROFUNDO**

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

2026

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

Rodrigues Fernandes, Saulo Enock.

Diagnóstico Automático da Paralisia do Sexto Nervo em Vídeos de Motilidade Ocular por Classificação de Séries Temporais e Aprendizado Profundo / Saulo Enock Rodrigues Fernandes. - 2026.

94 f.

Orientador(a): João Dallyson Sousa de Almeida.

Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, Auditório do Nca, 2026.

1. Sexto Nervo Óptico. 2. Paralisia do Sexto Nervo. 3. Rastreamento Ocular. 4. Classificação. 5. Séries Temporais. I. Sousa de Almeida, João Dallyson. II. Título.

SAULO ENOCK RODRIGUES FERNANDES

**DIAGNÓSTICO AUTOMÁTICO DA PARALISIA DO SEXTO NERVO EM
VÍDEOS DE MOTILIDADE OCULAR POR CLASSIFICAÇÃO DE SÉRIES
TEMPORAIS E APRENDIZADO PROFUNDO**

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.

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

**Prof. Dr. Tiago Bonini Borchartt
(UFMA)**
(Avaliador Interno)

**Profa. Dra. Andrea Gomes Campos
(UFOP)**
(Avaliador Externo)

SÃO LUÍS – MA
2026

Primeiramente a Deus, aos meus pais, a minha namorada, ao meu falecido irmão Carlos...

AGRADECIMENTOS

Gostaria de agradecer, primeiramente, a Deus por ter me abençoado ao longo de toda esta caminhada, por me proteger e guiar em todos os momentos.

Agradeço à minha mãe, que sempre trabalhou e se sacrificou para cuidar de mim com todas as forças que tinha, nunca me deixando faltar nada e me amando incondicionalmente. Agradeço ao meu pai por sempre ter se esforçado para ser um pai melhor a cada dia, por me ensinar e me aconselhar. Obrigado, mãe, obrigado, pai. Eu amo muito vocês do fundo do meu coração.

Gostaria também de agradecer à minha namorada Fernanda Ayla, que me apoiou muito, sempre me ajudou e incentivou, apareceu de surpresa em minha vida e tornou tudo mais leve e alegre, tornando-me um homem muito melhor do que jamais fui. Eu te amo demais.

Agradeço de coração ao meu orientador, João Dallyson. Obrigado por tudo, o apoio, os incentivos, a paciência, o cuidado e por acreditar em mim durante e muito antes desses dois anos de mestrado. Obrigado pelas orientações, pelas cobranças, pelas oportunidades e pela sua compreensão. Sou muito grato por tudo o que fez por mim.

Agradeço ao Prof. Anselmo Paiva, que me convidou para o NCA, me oferecendo oportunidades e um ambiente de crescimento pessoal e profissional que mudaram minha vida e me levaram para onde estou hoje. Sempre serei imensamente grato ao senhor.

Agradeço à Prof. Simara Rocha, que sempre foi um anjo na vida dos alunos de Ciência da Computação da UFMA. Sua leveza, compreensão, carinho e simpatia ajudaram e continuam ajudando todos nós a sermos não só estudantes, mas também pessoas melhores. Agradeço também a todos os professores do NCA, Prof. Geraldo Braz, Prof. Ari, Prof. Ítalo, Prof. Portela, Prof. Riveiro, Prof. Darlan, Prof. Davi, Prof. Vandécia e tantos outros são modelos de inspiração e portos seguros de apoio quando mais precisamos.

Agradeço aos meus amigos Becker e Priscila pelo apoio e pela amizade, que estão comigo desde o início dessa caminhada. Vocês são muito importantes para mim, e sou grato por tê-los na minha vida. Amo muito cada um de vocês.

Agradeço aos meus amigos, Fernando, João Leonardo, Igor e Juliany, Vitor, Andrews, Pedro, Estephane, Alexandre(s) Pessoa e Araújo, João Pedro, Mário, Alan, Gabriel, Alison, Kelson, Daniel, Marcus e Milena, Márcio, Mackele, entre outros, até mesmo os “Podres”, por serem tão especiais, cada um do seu próprio jeito.

Gostaria de agradecer, em especial, ao meu falecido irmão, Carlos André. Que Deus o tenha, meu irmão. As saudades que sinto por ti não têm fim.

“Amarás ao Senhor teu Deus de todo o teu coração, de toda a tua alma, e de todo o teu entendimento. Este é o grande e primeiro mandamento. E o segundo, semelhante a este, é: Amarás ao teu próximo como a ti mesmo.”

Jesus de Nazaré

RESUMO

O sexto nervo craniano está conectado ao músculo reto lateral, responsável pelos movimentos oculares da esquerda para a direita. A paralisia desse nervo impede o funcionamento adequado do músculo reto lateral, podendo causar dores de cabeça, enxaquecas, visão turva, tontura e diplopia (visão dupla) ao tentar mover o olho para o canto externo. Portanto, é essencial diagnosticar essa condição precocemente para prevenir efeitos a longo prazo. Como as técnicas de diagnóstico disponíveis são invasivas ou de alto custo, este estudo propõe um método automatizado para o diagnóstico da paralisia do sexto nervo craniano, baseado na classificação de séries temporais extraídas a partir do rastreamento ocular de pacientes oftalmológicos por meio de modelos de aprendizado profundo, visando auxiliar os especialistas no processo de diagnóstico. O método proposto utiliza a rede YOLO para rastrear o deslocamento ocular com precisão e rapidez em vídeos oftalmológicos, juntamente com o modelo de detecção facial do MediaPipe, em uma estratégia de compensação das trajetórias dos olhos, removendo os movimentos ocasionais da cabeça e isolando o deslocamento ocular real. Em experimento de validação cruzada, o método proposto alcançou 70% de sensibilidade na média da validação cruzada empregada e 75% no melhor *fold* treinado, indicando que nossa metodologia tem potencial para aplicação clínica.

Palavras-chave: Sexto Nervo Óptico, Paralisia do sexto nervo, Rastreamento ocular, Séries temporais, Classificação, Vídeos digitais, Velocidade média.

ABSTRACT

The sixth cranial nerve innervates the lateral rectus muscle, which is responsible for left-to-right eye movements. Paralysis of this nerve impairs the lateral rectus muscle's function, potentially causing headaches, migraines, blurred vision, dizziness, and diplopia (double vision) when attempting to move the eye toward the outer corner. Therefore, early diagnosis of this condition is essential to prevent long-term effects. Since available diagnostic techniques are invasive or expensive, this study proposes an automated method for diagnosing sixth cranial nerve palsy based on classifying time series extracted from eye-tracking data of ophthalmological patients using deep learning models, aiming to assist specialists in the diagnostic process. The proposed method uses the YOLO network to track eye movement accurately and quickly in ophthalmological videos, along with the MediaPipe facial detection model, in a strategy to compensate for eye trajectories, removing occasional head movements and isolating the actual eye movement. In a cross-validation experiment, the proposed method achieved a mean sensitivity of 70% across the folds and 75% in the best-trained fold, indicating that our methodology has potential for clinical application.

Keywords: Sixth Optic Nerve, Sixth nerve palsy, Eye tracking, Time series, Classification, Digital videos, Average speed.

LISTA DE ILUSTRAÇÕES

Figura 1 – Posição Primária do Olhar e Duções.	24
Figura 2 – Exemplos de Versões e Vergências.	25
Figura 3 – Sacadas e fixações.	26
Figura 4 – Movimentos persecutórios.	27
Figura 5 – Paralisia do sexto nervo craniano no olho esquerdo.	28
Figura 6 – Métodos tradicionais para detectar a paralisia do sexto nervo óptico.	29
Figura 7 – Inteligência artificial e suas áreas.	31
Figura 8 – <i>Machine Learning</i> e <i>Deep Learning</i>	32
Figura 9 – <i>Artificial Neural Networks</i>	33
Figura 10 – <i>Convolutional Neural Networks</i>	34
Figura 11 – Etapas da rede YOLO.	36
Figura 12 – Arquitetura YOLOv8.	38
Figura 13 – Atribuição de rótulos duplos.	39
Figura 14 – Principais módulos da arquitetura da YOLO11.	40
Figura 15 – Arquitetura YOLOv11.	41
Figura 16 – Detecção de face do MediaPipe.	42
Figura 17 – Exemplo de Séries Temporais.	43
Figura 18 – Exemplo de Séries Temporais.	44
Figura 19 – Exemplo de classificação por KNN.	46
Figura 20 – Comparação distância euclidiana e DTW	48
Figura 21 – Curva da razão entre precisão e sensibilidade.	52
Figura 22 – Fluxograma do método proposto.	55
Figura 23 – Curva mAP do modelo YOLOv8 416x416.	66
Figura 24 – Matriz de confusão da predição do modelo YOLOv8 416x416.	67
Figura 25 – Pacientes utilizados no estudo de caso.	73
Figura 26 – Série temporal da trajetória dos olhos do paciente 1.	73
Figura 27 – Interpolação da série temporal do paciente 1.	74
Figura 28 – Filtro de Hampel aplicado.	74
Figura 29 – Janelamento por sobreposição.	75
Figura 30 – Série temporal da trajetória dos olhos do paciente 14.	77
Figura 31 – Interpolação da série temporal do paciente 14.	77

Figura 32 – Filtro de Hampel.	78
Figura 33 – Janelamento por sobreposição.	78
Figura 34 – Trajetórias oculares dos pacientes estudados.	79

LISTA DE TABELAS

Tabela 1 – Resultado da classificação com Série Temporal do Movimento Ocular.	64
Tabela 2 – Resultado da classificação com Série Temporal da Posição Ocular.	64
Tabela 3 – Finetune (128 x 128)	65
Tabela 4 – Finetune (416 x 416)	66
Tabela 5 – Classificação baseada na velocidade média	68
Tabela 6 – Resultados por série temporal com séries univariadas.	70
Tabela 7 – Resultados por paciente com séries univariadas.	71
Tabela 8 – Resultados por série temporal com séries multivariadas.	71
Tabela 9 – Resultados por paciente com séries multivariadas.	71
Tabela 10 – Resultados por Fold do melhor modelo	71
Tabela 11 – Comparação de resultados da literatura.	72
Tabela 12 – Artigo publicado relacionado à metodologia proposta.	82

LISTA DE ABREVIATURAS E SIGLAS

AI	<i>Artificial Intelligence</i>
ANN	<i>Artificial Neural Networks</i>
AUC	<i>Area Under the Curve</i>
C2PSA	<i>Cross Stage Partial with Spatial Attention</i>
CBO	Conselho Brasileiro de Oftalmologia
CBS	<i>Convolution-BatchNorm-Silu</i>
CNN	<i>Convolutional Neural Networks</i>
CSP	<i>Cross Stage Partial</i>
DEXTRO	Dextroversão
DMA	Desvio Médio Absoluto
DTW	<i>Dynamic Time Warping</i>
EOG	Eletro-oculografia
FN	Falso-negativo
FP	Falso-Positivo
FPS	<i>Frames Per Second</i>
GAK	<i>Global Alignment Kernel</i>
ICC	<i>Intraclass Correlation Coefficient</i>
INFRA	Infraversão
IOU	<i>Intersection Over Union</i>
KNN	<i>K Nearest Neighbors</i>
LED	<i>Light Emitting Diode</i>
LEVO	Levoversão

MAP	<i>Mean Average Precision</i>
ML	<i>Machine Learning</i>
MLP	<i>Multilayer Perceptron</i>
NMS	Supressão Não Máxima
OI	Oblíquo Inferior
OMS	Organização Mundial de Saúde
OS	Oblíquo Superior
RI	Reto Inferior
RL	Reto Lateral
RM	Reto Medial
RNN	<i>Recurrent Neural Network</i>
RS	Reto Superior
SiLU	<i>Sigmoid Linear Unit</i>
SPPF	<i>Spatial Pyramid Pooling-Fast</i>
STMO	<i>Série Temporal de Movimento Ocular</i>
STPO	<i>Série Temporal da Posição Ocular</i>
SUPRA	Supraversão
SVC	<i>Support Vector Classifier</i>
SVM	<i>Support Vector Machine</i>
TFP	Taxa de Falsos Positivos
TGF	Teste de Geração de Força
TVP	Taxa de Verdadeiros Positivos
VN	Verdadeiro-Negativo

VP	Verdadeiro-Positivo
YOLO	<i>You Only Look Once</i>
YOLOv8	<i>You Only Look Once</i> versão 8
YOLOv10	<i>You Only Look Once</i> versão 10
YOLOv11	<i>You Only Look Once</i> versão 11

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Objetivo	18
1.1.1	Objetivos Específicos	19
1.2	Organização do trabalho	19
2	TRABALHOS RELACIONADOS	20
3	FUNDAMENTOS TEÓRICOS	23
3.1	Oculomotricidade	23
3.1.1	Sacadas	25
3.1.2	Fixações	26
3.1.3	Movimento Persecutório	26
3.2	Paralisia do Sexto Nervo Óptico	27
3.3	Machine Learning	30
3.4	Deep Learning	31
3.5	Artificial Neural Networks	32
3.6	Convolutional Neural Networks	33
3.7	Rede YOLO (You Only Look Once)	35
3.7.1	YOLOv8	37
3.7.2	YOLOv10	38
3.7.3	YOLO11	40
3.8	MediaPipe	42
3.9	Séries Temporais	43
3.9.1	Interpolação	44
3.9.2	Filtro de Hampel	45
3.10	Modelo K Neighbors TsLearn	45
3.10.1	K-Nearest Neighbors	45
3.10.2	Distância Euclidiana	47
3.10.3	Dynamic Time Warping	47
3.11	Métricas de Avaliação	50
4	MATERIAIS E MÉTODO	53

4.1	Bases de vídeos e imagens	53
4.1.1	Base de imagens de Estrabismo	53
4.1.2	Base de vídeos do Sexto Nervo	53
4.2	Método Proposto	54
4.2.1	Rastreamento Ocular	55
4.2.2	Rastreamento Facial	56
4.2.3	Compensação dos movimentos da face	56
4.2.4	Geração de séries temporais	57
4.2.5	Pré-processamento de dados	57
4.2.5.1	Interpolação	58
4.2.5.2	Filtro de Hampel	58
4.2.6	Janelamento	58
4.2.7	Aumento de dados	59
4.2.8	Classificadores de séries temporais	59
5	RESULTADOS E DISCUSSÃO	62
5.1	Experimento de classificação com trajetórias da YOLOv3	62
5.2	Experimento de rastreamento ocular com modelos YOLO	64
5.3	Experimento de classificação por velocidade média	67
5.4	Experimento final de classificação	68
5.5	Estudo de Caso	72
5.5.1	Caso de acerto: Paciente 1	73
5.5.2	Caso de falha: Paciente 14	76
6	CONCLUSÃO	80
6.1	Trabalhos Futuros	82
6.2	Produções Científicas	82
	REFERÊNCIAS	83

1 INTRODUÇÃO

O olho humano é um dos órgãos mais complexos e especializados do corpo, exigindo um extenso processamento pelo cérebro. Portanto, por se tratar de uma região do corpo extremamente sensível, a detecção precoce de doenças oculares é crucial para prevenir danos permanentes. Um estudo de 2023 da Organização Mundial da Saúde (OMS) relatou que pelo menos 2,2 bilhões de pessoas em todo o mundo são afetadas por alguma forma de doença ocular (ORGANIZATION, 2023). Em 2023, o Conselho Brasileiro de Oftalmologia (CBO) estimou que 0,75% da população brasileira, ou seja, aproximadamente 1,6 milhão de pessoas, apresentava algum grau de cegueira (UMBELINO; ÁVILA, 2023). Segundo a OMS, o envelhecimento da população é o principal motivo do aumento do número de pessoas com deficiência visual no mundo, seguido pelas mudanças no estilo de vida e pela urbanização (UMBELINO; ÁVILA, 2023). Dessa forma, o fácil acesso a serviços oftalmológicos de qualidade é a solução para um tratamento eficaz, bem como para o controle do risco de progressão de deficiências visuais e de outras doenças oculares (ORGANIZATION, 2023).

Dentre as diversas condições que afetam o olho humano, destaca-se a paralisia do sexto nervo óptico, também conhecido como nervo abducente, pois controla o músculo reto lateral, responsável pelo movimento do olho para o canto externo. A paralisia desse nervo pode causar dores de cabeça, enxaquecas, visão turva, tontura e visão dupla (diplopia) ao tentar mover o olho para o canto externo.

Existem dois métodos principais para a identificação da paralisia do sexto nervo: o Teste de Geração de Força (TGF) e a Eletrooculografia (EOG). O TGF é invasivo e desconfortável, depende totalmente da experiência do examinador, pois o paciente move rapidamente o olho paralisado em direção ao músculo reto lateral, e o oftalmologista sente a força gerada pelo deslocamento do músculo com uma pinça, após anestesia tópica, para determinar o grau de paralisia (CURI RENATO LUIZ NAHOUM, 2013). Por outro lado, a eletro-oculografia é um exame complexo que depende de equipamento específico e mede as flutuações do potencial corneoretiniano, rastreando os movimentos oculares. A velocidade e a localização dos movimentos oculares são medidas por meio de eletrodos colocados ao redor dos olhos, auxiliando no diagnóstico da paralisia. Esse exame requer tecnologia cara, disponível apenas em grandes laboratórios de pesquisa, e é impraticável

na maioria dos consultórios oftalmológicos (MARTINSEN; HEISKANEN, 2023).

Portanto, com o objetivo de oferecer uma opção alternativa de baixo custo e não invasiva, esta pesquisa propõe um método digital para detectar paralisia do sexto nervo óptico, empregando a classificação, por meio de modelos de aprendizado profundo, a partir de séries temporais da trajetória ocular de pacientes oftalmológicos, auxiliando especialistas no diagnóstico dessa condição.

Nesse contexto, o trabalho é conduzido com base nas seguintes questões de pesquisa: (1) É possível desenvolver uma metodologia digital para diagnóstico da paralisia do sexto nervo que não dependa da definição prévia de um limiar, tal qual na literatura (COSTA et al., 2022), para identificar casos positivos da condição paralítica? E (2) o método proposto, uma abordagem nova baseada na classificação de séries temporais das trajetórias oculares de pacientes oftalmológicos, apresenta desempenho superior ao método baseado na classificação a partir do limiar estimado?

Com esse objetivo, é realizado o rastreamento do deslocamento ocular com a rede neural YOLOv8 (You Only Look Once versão 8) para obter a trajetória de cada olho do paciente, seguido do rastreamento da face com o modelo de detecção facial do MediaPipe, que obtém os movimentos voluntários e involuntários da cabeça durante os vídeos, para que então seja feita a compensação do deslocamento da face na trajetória dos olhos, garantindo a precisão das trajetórias percorridas pelos olhos de cada paciente. Continuando, essas trajetórias são transformadas em séries temporais, passam por uma etapa de pré-processamento, necessária para refinar as séries temporais adquiridas, e, finalmente, são aplicadas arquiteturas de machine learning para classificar a presença de paralisia com base em séries temporais de movimento ocular.

1.1 Objetivo

Diante do contexto apresentado, este trabalho tem por objetivo propor um novo método computacional, automático e de baixo custo para auxiliar no diagnóstico da paralisia do sexto nervo óptico em vídeos de pacientes realizando movimentos oculares. Para isso, utilizam-se redes neurais convolucionais, juntamente com técnicas de processamento de imagens e de séries temporais, para classificar pacientes com ou sem paralisia com base na movimentação ocular obtida.

1.1.1 Objetivos Específicos

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

- Desenvolver um método automatizado de rastreamento ocular em vídeos de pacientes oftalmológicos com paralisia do sexto nervo óptico.
- Detectar a face dos pacientes e compensar a movimentação da cabeça nas trajetórias oculares, garantindo que os dados representem apenas os movimentos dos olhos.
- Gerar séries temporais que representam os padrões de movimento ocular dos pacientes.
- Classificar os padrões de movimento ocular por meio de diferentes algoritmos de aprendizado de máquina voltados a séries temporais.
- Comparar os resultados alcançados pela metodologia proposta com o diagnóstico do oftalmologista, com o objetivo de validá-los;

1.2 Organização do trabalho

Esta é a organização deste trabalho: **Capítulo 2** evidencia os trabalhos relacionados que serviram de referência para esta dissertação. O **Capítulo 3** apresenta a fundamentação teórica desta pesquisa, desenvolvendo os conceitos e as técnicas necessários ao entendimento da metodologia adotada. Aborda os fundamentos da oculomotricidade, explica a paralisia do sexto nervo óptico, a rede neural YOLO e suas versões utilizadas em experimentos, o modelo de rastreamento facial do framework MediaPipe e o modelo de classificação aplicado, o K Neighbors TsLearn. Já o **Capítulo 4** apresenta o método proposto neste trabalho, descrevendo suas etapas, desde a obtenção da base de dados até as métricas de avaliação empregadas. O **Capítulo 5** apresenta os resultados obtidos com a metodologia desenvolvida, discutindo-os e interpretando-os. Por último, o **Capítulo 6** apresenta uma perspectiva geral sobre a pesquisa desenvolvida, analisando sua eficácia, seus defeitos e quais melhorias seriam adequadas, seguida das contribuições do trabalho e de ideias para trabalhos futuros.

2 TRABALHOS RELACIONADOS

Diversos estudos aplicaram métodos computacionais com o objetivo de detectar automaticamente doenças oculares ou apoiar o trabalho do oftalmologista no diagnóstico (ALMEIDA et al., 2015; SILVA et al., 2018; ARAÚJO et al., 2019; CLARO et al., 2019; SOUSA et al., 2021). No entanto, com exceção do trabalho de Costa et al. (2022), até o momento da redação deste trabalho, não foram encontrados métodos voltados especificamente à detecção ou ao diagnóstico da paralisia do sexto nervo óptico por meio de imagens, vídeos digitais ou séries temporais, considerando-se buscas realizadas por métodos tradicionais. Assim, essa condição ainda é geralmente identificada por meio de exames convencionais, como o TGF e a eletro-oculografia.

O trabalho de Costa et al. (2022) propõe uma abordagem computacional automática para auxiliar no diagnóstico da paralisia do sexto nervo óptico. O método sugerido rastreia a trajetória do movimento de cada olho em vídeos de pacientes, utilizando redes neurais convolucionais e algoritmos de processamento de imagem. A velocidade média com que cada olho se move é determinada a partir dessa trajetória, então estima-se um limiar com base nos deslocamentos obtidos, e esse limiar é aplicado para verificar se um olho é parético ou saudável, pois é bem sabido que os olhos paréticos se movem mais lentamente do que os saudáveis. Os oftalmologistas podem se beneficiar desse limite e da velocidade média do movimento ocular ao realizar suas análises. Com um índice Kappa de 0,925 e uma pontuação de precisão de 92,64% no diagnóstico de paralisia do sexto nervo.

No estudo de Prasad (PRASAD; VOLPE, 2010), utilizou-se o sistema de vinculação de registros médicos do Projeto Epidemiológico de Rochester para identificar novos casos de paralisia do sexto nervo craniano no condado de Olmsted ao longo de 15 anos, por meio da análise de prontuários médicos de pacientes, a fim de confirmar os diagnósticos, verificar o local de residência e identificar as condições clínicas associadas. As associações mais frequentes envolveram causas indeterminadas, hipertensão, concomitância de hipertensão e diabetes, trauma e esclerose múltipla, ao passo que neoplasias e aneurismas foram relativamente raros. O estudo concluiu que a paralisia do sexto nervo está mais comumente associada a fatores de risco vasculares, como diabetes e hipertensão, e menos frequentemente a neoplasias, do que o sugerido anteriormente por estudos institucionais.

Por outro lado, além desse estudo, é possível identificar, na literatura, pesquisas que

se alinham com parte da proposta desse trabalho. Por exemplo, na pesquisa de Samadani et al. (2015), foi desenvolvido um algoritmo de rastreamento ocular para detectar paralisias dos nervos oculomotor (III) e abducente (VI) em pacientes com condições neurológicas. O método analisa a razão entre os movimentos verticais e horizontais da pupila enquanto os pacientes assistem a um vídeo, identificando padrões característicos de paralisia. Pacientes com paralisia do sexto nervo apresentaram razões elevadas (até 1,84), enquanto os com paralisia do terceiro apresentaram valores reduzidos (até 0,06). Após a intervenção cirúrgica, os valores normalizaram, demonstrando o potencial diagnóstico do método, automatizado e não invasivo.

Por sua vez, o estudo de Rakhmatulin e Duchowski (2020) desenvolveu um sistema de rastreamento ocular em tempo real, de baixo custo, usando YOLOv3 para detectar a direção do olhar (esquerda, direita e centro), combinando essa informação com a posição do rosto em relação a um monitor, identificada por LEDs infravermelhos via OpenCV. A abordagem demonstrou ser uma alternativa de baixo custo com desempenho comparável a equipamentos laboratoriais, utilizando ferramentas como Haar Cascades (VIOLA; JONES, 2001; MCCULLAGH, 2023) e Dlib (KING, 2009) para auxiliar na delimitação ocular.

Na pesquisa de Jyotsna (JYOTSNA et al., 2023), é proposto um novo modelo personalizado de séries temporais, o PredictEYE, que prevê o estado mental de uma pessoa e identifica, em um vídeo, a cena específica responsável por esse estado. O modelo avalia dados de séries temporais de rastreamento ocular de pessoas enquanto assistem a vídeos relaxantes e estressantes, empregando um modelo de regressão de séries temporais univariado de aprendizado profundo baseado em Memória de Longo Prazo (LSTM) para prever a evolução futura de cada característica e um algoritmo de Floresta Aleatória, fundamentado em aprendizado de máquina, para estimar o estado mental.

Já no estudo de Zhou Zhou et al. (2024), foi proposta uma metodologia baseada em deep learning para a análise automática da abdução ocular em pacientes com paralisia do sexto nervo submetidos à cirurgia de estrabismo. Utilizando uma rede R2AU-Net (ZUO; CHEN; WANG, 2021), o modelo segmenta os olhos e estima parâmetros de abdução com alta repetibilidade, apresentando correlações intraclasse (ICC entre 0,985 e 0,992). Os resultados automáticos mostraram baixo viés e, após a cirurgia, observou-se melhora na abdução em 76,9% dos pacientes.

Há também o trabalho proposto por Angsana (ANGSANA; INDRIASARI;

MAHAYANA, 2025), no qual foi utilizado o sistema GENICULA (Gaze Tracking Artificial Intelligence for Oculomotor Palsy), um método de rastreamento ocular em tempo real baseado em vídeo, acoplado a uma webcam convencional, para a obtenção de medidas do ângulo visual. Foi então realizada uma análise estatística comparativa entre pacientes com paralisia do nervo abducente e indivíduos saudáveis, para avaliar a evolução das medidas antes e após a recuperação dos pacientes acometidos. O estudo contou com 39 participantes e demonstrou diferenças estatisticamente significativas entre os grupos analisados, além de melhora significativa após o tratamento.

Nossa abordagem se diferencia de Costa et al. (2022) por não depender de um limiar pré-definido para identificar casos de paralisia, delegando a tarefa de classificar o sexto nervo a modelos de aprendizado profundo. Além disso, enquanto trabalhos como (RAKHMATULIN; DUCHOWSKI, 2020) e (ANGSANA; INDRIASARI; MAHAYANA, 2025) focam principalmente no rastreamento ocular e na estimativa da direção do olhar, esta pesquisa emprega versões mais recentes da arquitetura YOLO, combinadas a métodos adicionais de rastreamento ocular, visando maior robustez na extração das trajetórias oculares. Em contraste com (PRASAD; VOLPE, 2010), que tem foco epidemiológico, e com (ZHOU et al., 2024) e (JYOTSNA et al., 2023), que focam em predição, nosso trabalho concentra-se especificamente na classificação de séries temporais relacionadas à paralisia do sexto nervo, explorando representações univariadas e multivariadas. Adicionalmente, ao contrário de (SAMADANI et al., 2015), que investiga simultaneamente diferentes paralisias oculomotoras, esta pesquisa dedica-se exclusivamente à análise da paralisia do sexto nervo, o que permite uma investigação mais direcionada das características temporais associadas à patologia.

3 FUNDAMENTOS TEÓRICOS

Este capítulo descreve os fundamentos teóricos necessários ao entendimento do problema abordado neste trabalho e as técnicas que embasam o método proposto (Seção 4). Serão abordadas noções de oculomotricidade, a definição da paralisia do sexto nervo óptico, explicações sobre a família de redes neurais YOLO (You Only Look Once) e suas versões empregadas, informações sobre o framework MediaPipe e fundamentos de séries temporais, juntamente com as técnicas de pré-processamento aplicadas. E, por último, a explicação do modelo classificador K Neighbors (TsLearn) utilizado.

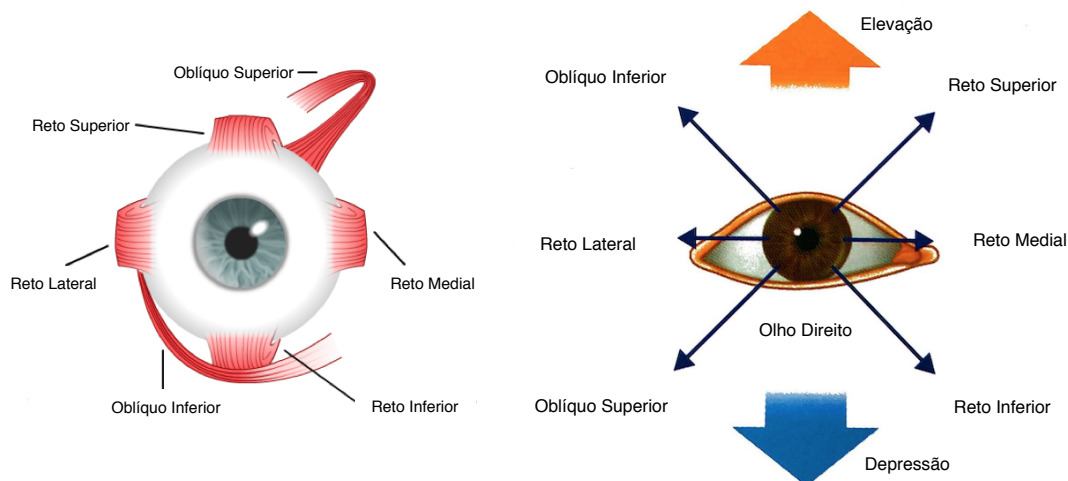
3.1 Oculomotricidade

Os movimentos dos olhos são fundamentais para a captação de informações na vida cotidiana e para ações motoras simples do corpo humano. Logo, o olho humano está sempre em movimento para antecipar ações ou acontecimentos, em vez de responder a eles (LAND; FURNEAUX, 1997). O posicionamento dos olhos é garantido por doze músculos oculares externos, seis em cada olho, acionados por três pares de nervos. Esses nervos são estimulados pela cooperação entre os comandos volitivos ¹, reativos a essas movimentações e às reações automáticas complementares, formando, assim, uma sinergia metódica com as funções sensoriais oculares (BICAS, 2003). Esse conjunto de funções responsáveis pelos movimentos dos olhos é denominado oculomotricidade.

Existem seis músculos extraoculares responsáveis por mover os olhos: os músculos reto superior e oblíquo inferior elevam o olho; já o reto inferior e o oblíquo superior executam o movimento para baixo; o músculo reto medial é responsável pelo deslocamento do olho em direção ao nariz; e o reto lateral movimenta o olho em direção ao lado da têmpora (SNELL; LEMP, 2013). Esses músculos, com seus respectivos movimentos, são mostrados na Figura ??, com o olho direito como referência.

Dos músculos apresentados anteriormente na Figura ??, os movimentos executados por eles podem ser divididos em três grupos: Duções, Versões e Vergências (SNELL; LEMP, 2013). As duções são deslocamentos isolados de um único olho, sendo elas: **Adução**, que se refere ao movimento do olho em direção ao nariz; **Abdução**, que é o deslocamento do

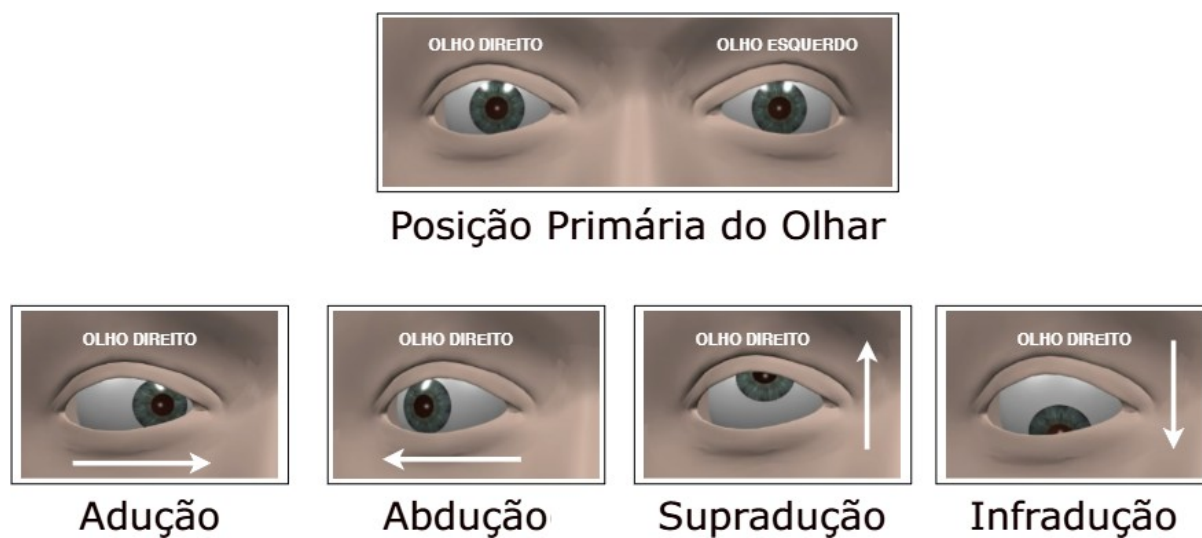
¹ movimentos voluntários e conscientes realizados pelos olhos, orquestrados pelo sistema nervoso central para focar, rastrear objetos e explorar o ambiente



Fonte: Imagem de (COSTA et al., 2022).

olho em direção à região temporal; a **Supradução**, definida pela rotação do olho para cima; e a **Infradução**, que ocorre quando o olho rotaciona para baixo. Há também a posição primária do olhar, caracterizada pelos olhos estarem voltados diretamente para a frente. Cada uma dessas posições é exibida na Figura 1.

Figura 1 – Posição Primária do Olhar e Duções.

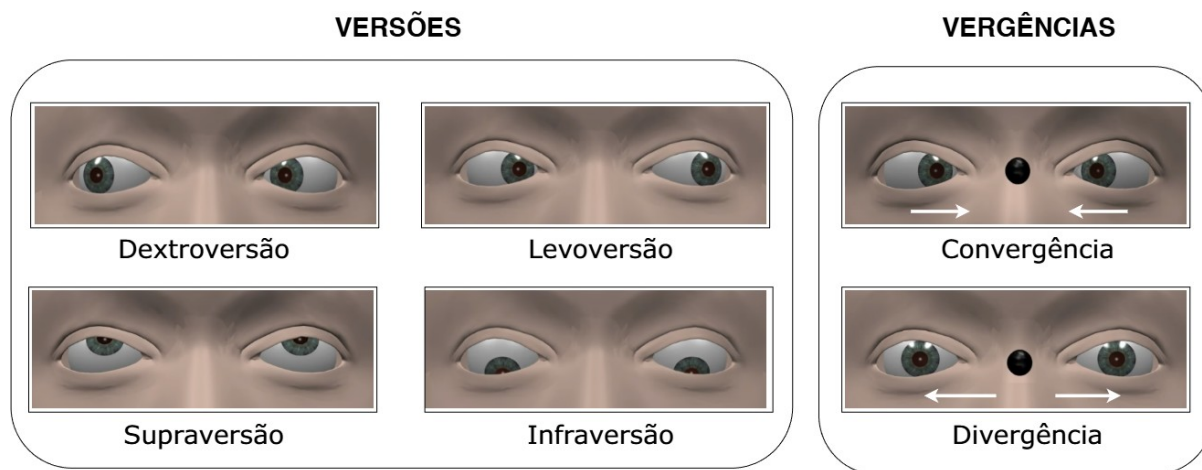


Fonte: Adaptado de (COSTA et al., 2022).

Por outro lado, as versões são movimentos conjuntos dos olhos na mesma direção. Elas são: a **Dextroversão** refere-se ao caso em que os olhos se movem para a direita; já a **Levoversão** ocorre quando ambos se movem para a esquerda; há também a **Supra-versão**, caracterizada pelo deslocamento conjunto dos olhos para cima; e, por último, a **Infraversão**, que ocorre no ato de olhar para baixo.

O último tipo de movimento ocular é a vergência, que envolve o movimento simultâneo dos olhos em direções opostas. Elas são: a **Convergência**, em que os olhos realizam o movimento de adução, ou seja, ambos se movem na direção da região nasal; e a **Divergência**, que se refere ao movimento de abdução dos olhos, voltando à posição primária do olhar. Portanto, as versões e vergências são combinações de duções nos dois olhos. Os movimentos de versões e vergências podem ser observados na Figura 2.

Figura 2 – Exemplos de Versões e Vergências.



Fonte: Adaptado de (COSTA et al., 2022).

3.1.1 Sacadas

Os movimentos sacádicos são movimentos oculares rápidos, voluntários e reflexos que reposicionam a fóvea ocular ² em uma nova posição no ambiente visual. Esses movimentos podem ser realizados voluntariamente ou induzidos como medida corretiva optocinética ³ ou vestibular (LIVERSEDGE; GILCHRIST; EVERLING, 2011). Um movimento de sacada é seguido por uma fixação, um período durante o qual o olho permanece relativamente imóvel e coletam-se informações visuais úteis. Como a acuidade visual diminui rapidamente ao se afastar da direção atual do olhar, os movimentos sacádicos são necessários para direcionar o olhar para regiões de interesse (DUCHOWSKI, 2017).

Durante uma fixação, a visão periférica é utilizada para determinar a localização da próxima fixação. Em uma cena com múltiplos alvos possíveis, a seleção do alvo para

² pequena depressão na mácula, região central da retina, especializada na visão de alta resolução, cores e detalhes, como ler ou dirigir

³ movimento ocular rápido que ocorre para reposicionar o olhar após uma fase lenta de rastreamento de um estímulo visual em movimento

o próximo movimento sacádico envolve uma interação entre as propriedades visuais das localizações no ambiente e o objetivo do observador.

3.1.2 Fixações

As fixações são movimentos oculares que estabilizam a retina sobre um objeto de interesse estacionário. Mudanças na orientação da cabeça em relação à gravidade desencadeiam reflexos otolíticos-oculares de baixa frequência que controlam a orientação dos olhos no espaço físico, contrarrotacionando em torno do eixo da cabeça (LIVERSEDGE; GILCHRIST; EVERLING, 2011).

Do ponto de vista do controle oculomotor, a estabilização da imagem retiniana requer duas respostas funcionalmente distintas que se sobrepõem temporalmente: uma consiste em emitir um comando motor rápido, proporcional à velocidade da cabeça, que contrarrotaciona o olho, com a menor latência possível, para compensar perturbações na posição da cabeça. A outra ação, ligeiramente atrasada, consiste em gerar comandos de posição que mantêm a nova orientação estável dos olhos (DUCHOWSKI, 2017).

As fixações são caracterizadas por movimentos oculares sutis, como tremor, deriva e microssacadas. As microssacadas são sinais de movimento ocular, geralmente aleatórios no espaço, com amplitude variando entre 1 e 2 minutos de arco. Na Figura 3 são exemplificadas as Fixações e Sacadas.

Figura 3 – Sacadas e fixações.



Fonte: Adaptado de (BATTY; PERKINS; SITA, 2015).

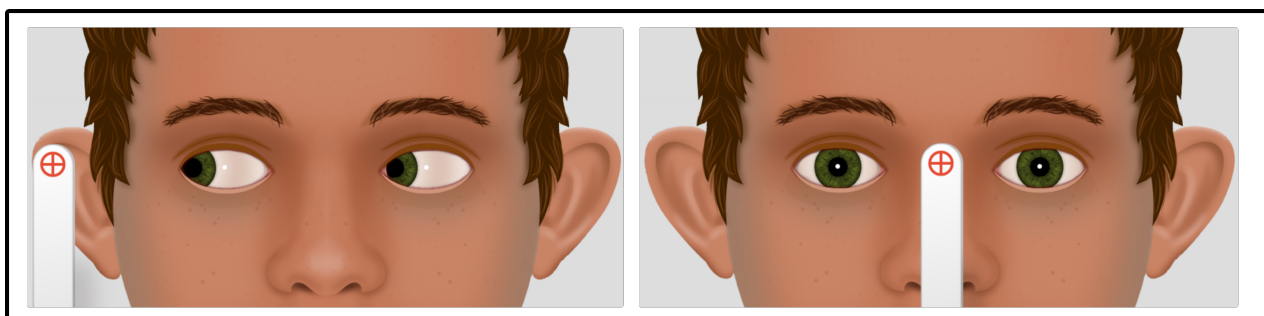
3.1.3 Movimento Persecutório

Os movimentos de perseguição envolvem o rastreamento visual de um alvo em movimento. Dependendo da amplitude do movimento do alvo, os olhos podem acompanhar sua velocidade.

Os movimentos oculares de perseguição permitem rastrear objetos em movimento, combinando movimentos suaves e sacádicos. O principal objetivo é manter uma velocidade ocular suave, próxima à do objeto, minimizando o movimento da imagem na retina e preservando a acuidade visual. As sacadas realinham a imagem quando ela se desloca para fora do centro do campo de visão, que é a área de maior acuidade visual. Os movimentos de perseguição são frequentemente descritos como voluntários, mas processos subjacentes detectam o movimento na retina e podem induzir movimentos oculares sem participação ativa (DUCHOWSKI, 2017).

O fator que distingue a perseguição de tais movimentos reflexos é a capacidade de selecionar e rastrear um único objeto entre múltiplos estímulos. O processo seletivo requer atenção, que parece aumentar a recompensa associada ao objeto selecionado e/ou suprimir a recompensa associada a outros estímulos. A competição resultante geralmente reduz a velocidade de perseguição (LIVERSEDGE; GILCHRIST; EVERLING, 2011). É possível ver um exemplo da execução do movimento persecutório na Figura 4.

Figura 4 – Movimentos persecutórios.



Fonte: Adaptado de (ORGE MD; K. DAVID EPLEY, 2019).

3.2 Paralisia do Sexto Nervo Óptico

O sexto nervo é o nervo óptico mais afligido por paralisias motoras oculares, ele é diretamente ligado ao músculo reto lateral e, portanto, encarregado pela sua contração (ROSENBAUM; SANTIAGO, 1999). Esse músculo é responsável pela abdução (Figura 1). Logo, um trauma no sexto nervo causa paralisia parcial ou total do músculo reto lateral, limitando ou até mesmo impedindo o movimento de abdução. A paralisia do sexto nervo craniano ocorre com mais frequência em pacientes adultos com mais de 40 anos, podendo ser causada por diversos fatores, enquanto, em crianças, a paralisia por doenças vasculares é menos frequente (ROSENBAUM; SANTIAGO, 1999). Entre os diversos casos em que

uma causa pode ser identificada, as vasculopatias ⁴ decorrentes de hipertensão, diabetes ou aterosclerose são as mais comuns. Além dessas deficiências, a paralisia desse nervo pode causar dores de cabeça, enxaquecas, visão turva, tontura e visão dupla ao tentar mover o olho para o canto externo (CURI; COSTA; BARROSO, 2013). Na Figura 5, é possível observar a disparidade entre um olho saudável e um paralítico. É perceptível que o olho esquerdo do paciente não consegue completar o movimento de abdução, parando na posição primária do olhar.

Figura 5 – Paralisia do sexto nervo craniano no olho esquerdo.



Fonte: Adaptado de (ORGE MD; K. DAVID EPLEY, 2019).

Apesar de ser uma condição atípica, a paralisia do sexto nervo deve ser diagnosticada e tratada precocemente para evitar danos permanentes no olho afetado, por se tratar de uma região do corpo extremamente frágil e sensível (CURI; COSTA; BARROSO, 2013). Existem dois métodos principais para detectar a paralisia do sexto nervo: o teste de geração de força e a eletro-oculografia.

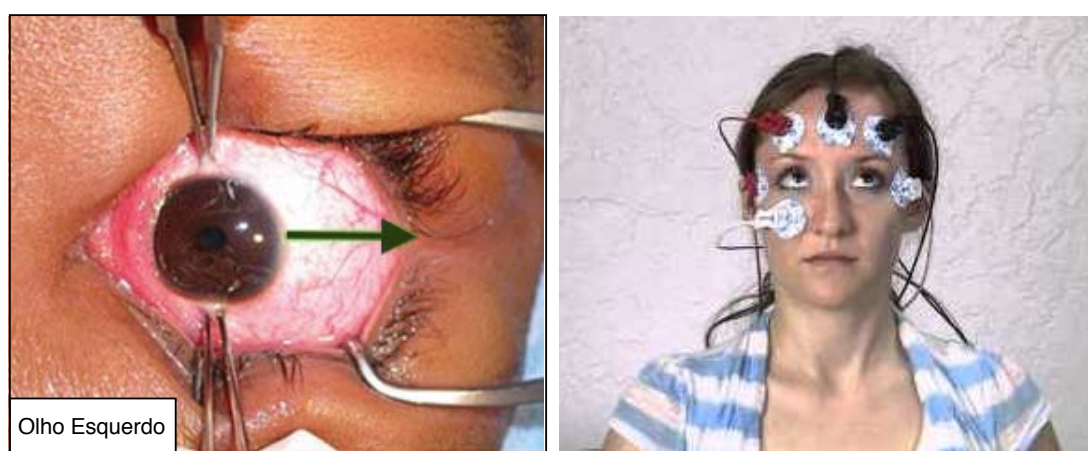
O teste de geração de força é um exame simples, porém invasivo e desconfortável, empregado para avaliar a fragilidade ou a inanição do potencial muscular, seja por lesão

⁴ vasculopatias são doenças do sistema circulatório

ou por paralisia do nervo. Ele, porém, requer que o paciente não esteja sedado e que coopere no exame, pois, após a aplicação de anestesia tópica no olho, é preciso que o paciente mova o olho na direção contrária ao músculo examinado, em direção ao nariz. A seguir, é colocada uma pinça no limbo temporal do olho (Figura 6) para segurá-lo e mantê-lo imobilizado. É preciso então que o paciente mova o olho em direção ao músculo reto lateral, dessa forma, o oftalmologista analisa a força sentida no fórceps. Ao tentar mover o olho enquanto está imobilizado, uma contração será sentida se o músculo estiver saudável, mas, caso esteja paralisado, ele não apresentará resistência ao fórceps. Ainda que seja simples, a eficácia desse teste depende da experiência do examinador. Além do mais, esse exame não pode ser aplicado em crianças, visto que a cooperação delas é muitas vezes inviável e a anestesia geral não é uma opção (METZ, 1976).

O outro método de detectar a presença de paralisia do sexto nervo é por meio da eletro-oculografia, que é um teste que avalia os movimentos oculares com base em sinais elétricos emitidos pelos músculos extraoculares (MACHADO et al., 2010). Esse exame registra os movimentos oculares a partir da diferença de potencial córneo-retiniano durante o deslocamento dos olhos, sendo eficaz para detectar não só a paralisia do sexto nervo, mas também diversas outras patologias oculares, como estrabismo e outras paralisias oculomotoras, entre outras (ZIFFER et al., 1992; BUCCI et al., 2002). O teste é realizado fixando eletrodos ao redor dos olhos do paciente, que registram a atividade muscular ocular durante sua movimentação, conforme mostra a Figura 6 (b), gerando, assim, a trajetória ocular e a velocidade sacádica.

Figura 6 – Métodos tradicionais para detectar a paralisia do sexto nervo óptico.



(a) Teste de Geração de Força

(b) Eletro-oculografia

Fonte: Imagem de (COSTA et al., 2022).

3.3 Machine Learning

Machine Learning (ML) é uma área específica da inteligência artificial (AI) que emprega algoritmos treinados com conjuntos de dados para desenvolver modelos aptos a executar tarefas que, tradicionalmente, exigiriam intervenção humana, como a categorização de imagens, a análise de dados ou a previsão de variações de preços (DU; SWAMY, 2013; SADIKU; AJAYI; SADIKU, 2025). A aprendizagem de máquina é atualmente a vertente da inteligência artificial mais bem-sucedida e disseminada. Trata-se da abordagem mais amplamente estudada e aplicada em praticamente todas as disciplinas e domínios. Existem inúmeras abordagens e modelos de aprendizagem de máquina para seleção e integração, e as amplas aplicações da aprendizagem de máquina têm gerado impactos de longo alcance na sociedade (DU et al., 2025).

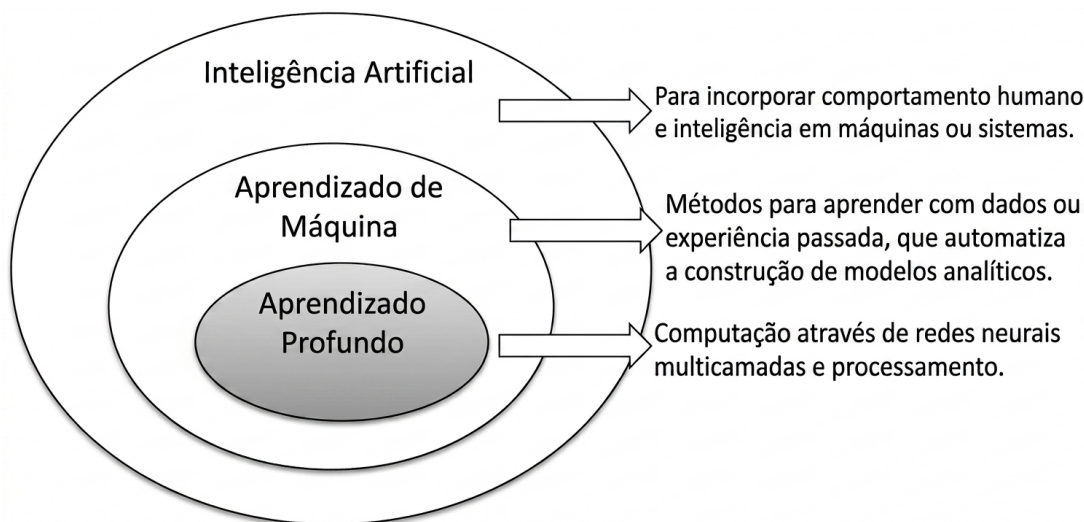
Ela utiliza algoritmos treinados em conjuntos de dados para criar modelos de autoaprendizagem capazes de prever resultados e classificar informações sem intervenção humana. O foco está em algoritmos que conseguem "aprender" os padrões dos dados de treinamento e, conseqüentemente, fazer inferências precisas sobre novos dados (SADIKU; AJAYI; SADIKU, 2025). Essa habilidade de identificar padrões permite que os modelos de aprendizado de máquina tomem decisões ou façam previsões sem depender de instruções rígidas e explícitas. Contudo, para que esses algoritmos operem de maneira realmente eficaz, geralmente precisam passar por diversos ciclos de refinamento até acumularem um conjunto abrangente de instruções que lhes permita funcionar adequadamente (SHINDE; SHAH, 2018).

O objetivo é garantir que o modelo possa generalizar para além do conjunto de dados de treinamento, com base na ideia de que existe uma relação matemática entre qualquer combinação de dados de entrada e de saída. O modelo de aprendizado de máquina não conhece essa relação antecipadamente, mas pode inferi-la se forem fornecidos exemplos suficientes de conjuntos de entrada e de saída (SADIKU; AJAYI; SADIKU, 2025). Em casos mais avançados, vários algoritmos são combinados para formar sistemas complexos conhecidos como redes de aprendizado profundo.

3.4 Deep Learning

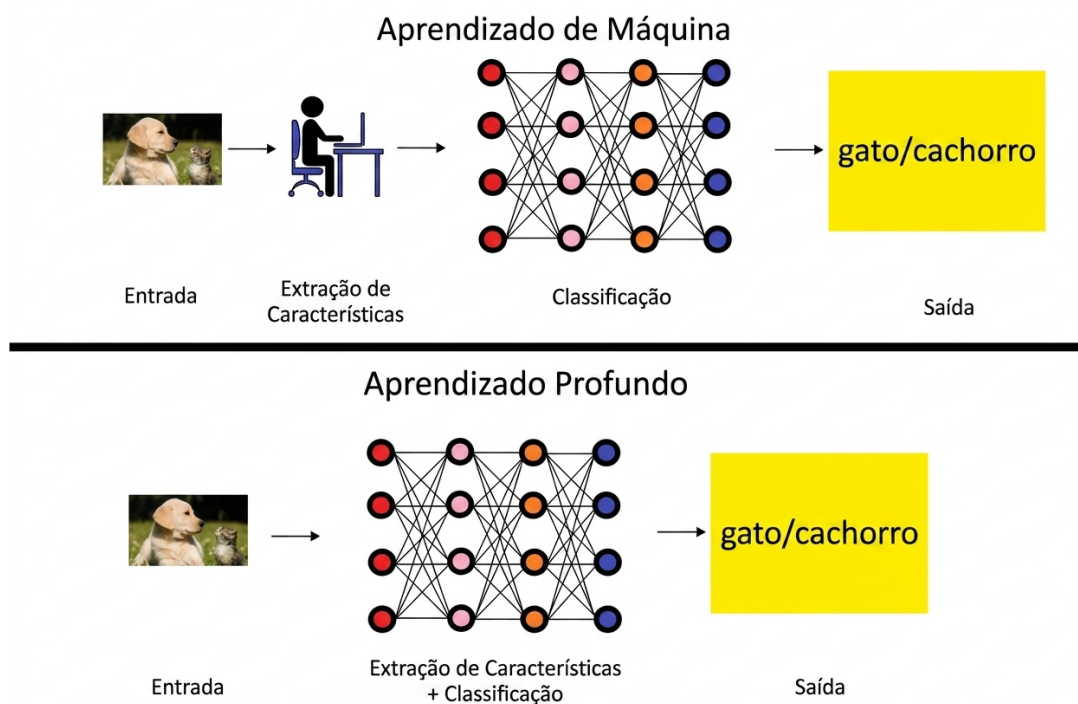
Deep Learning (DL) é um ramo da aprendizagem de máquina e da inteligência artificial (Figura 7), atualmente considerada uma tecnologia central da Quarta Revolução Industrial (4IR) (PIRES, 2025). Devido à sua capacidade de aprendizado, originária de redes neurais artificiais, tornou-se um tema em voga na computação e é amplamente aplicada em áreas como saúde, reconhecimento visual, análise de texto, segurança cibernética e muito mais (SARKER, 2021).

Figura 7 – Inteligência artificial e suas áreas.



Fonte: Imagem de (SARKER, 2021).

A aprendizagem profunda distingue-se da aprendizagem de máquina convencional em termos de eficiência à medida que a quantidade de dados cresce (Figura 8). Essa tecnologia utiliza diversas camadas para representar abstrações dos dados e construir modelos computacionais (SCHMIDHUBER, 2015). Embora o treinamento de um modelo de aprendizagem profunda seja demorado devido ao grande número de parâmetros, seu tempo de execução nos testes é muito menor do que o de outros algoritmos de aprendizagem de máquina (XIN et al., 2018).

Figura 8 – *Machine Learning e Deep Learning*.

Fonte: Imagem de (HALDER et al., 2019).

A teoria do aprendizado profundo demonstra que as redes profundas possuem duas vantagens exponenciais em relação aos algoritmos de aprendizado clássicos que não utilizam representações distribuídas. Primeiro, o aprendizado de representações distribuídas permite generalizar para novas combinações de valores de características aprendidos, além das observadas durante o treinamento. Em segundo lugar, a composição de camadas de representação em uma rede profunda traz o potencial de uma outra vantagem exponencial (LECUN; BENGIO; HINTON, 2015).

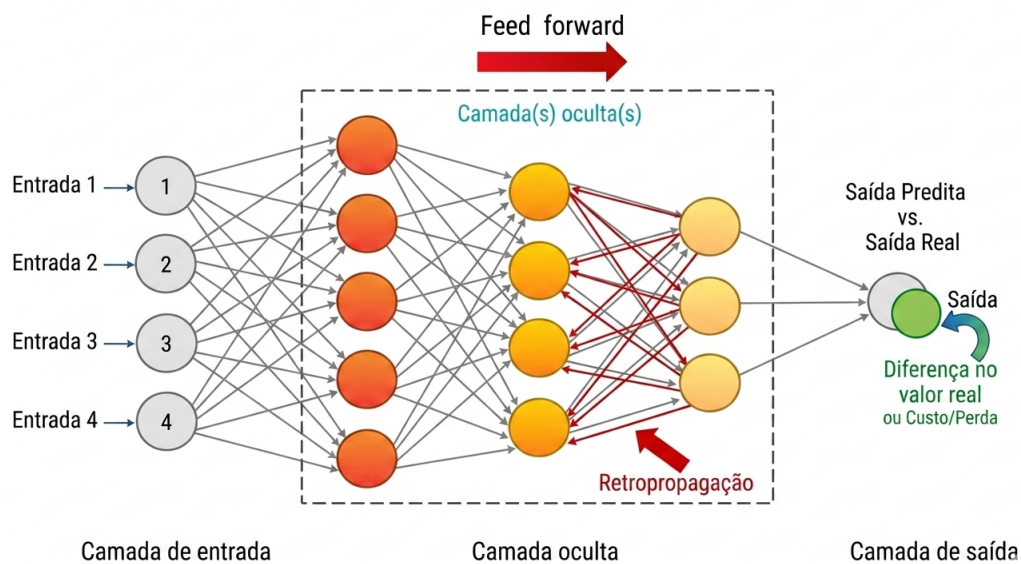
3.5 Artificial Neural Networks

No final da década de 1980, as Artificial Neural Networks (ANNs) tornaram-se um tópico proeminente em aprendizado de máquina e inteligência artificial (IA) devido ao desenvolvimento de métodos de aprendizado eficientes e de arquiteturas de rede (KARHUNEN; RAIKO; CHO, 2015). Redes de *Multilayer Perceptron* (MLP) (POPESCU et al., 2009) treinadas por algoritmos de *backpropagation*, mapas auto-organizáveis e redes de função de base radial estavam entre os métodos inovadores (SARKER, 2021).

Redes neurais são modelos estatísticos adaptativos baseados em uma analogia com a estrutura do cérebro. Elas são adaptativas porque conseguem aprender a estimar os

parâmetros de uma população usando um pequeno número de exemplos (um ou poucos) de cada vez. Elas não se distinguem, de forma fundamental, dos modelos estatísticos tradicionais. Em essência, as redes neurais são formadas por unidades simples, muitas vezes chamadas de neurônios por analogia (WU; FENG, 2018). Essas unidades são conectadas entre si por um conjunto de ligações com pesos associados (Figura 9). O aprendizado geralmente ocorre por meio da modificação dos pesos das conexões. Cada código de unidade corresponde a uma característica ou propriedade de um padrão que queremos analisar ou utilizar como preditor. As unidades são organizadas em camadas (ABDI; VALENTIN; EDELMAN, 1999).

Figura 9 – *Artificial Neural Networks*.



Fonte: Imagem de (THAMCHAROEN et al., 2020).

3.6 Convolutional Neural Networks

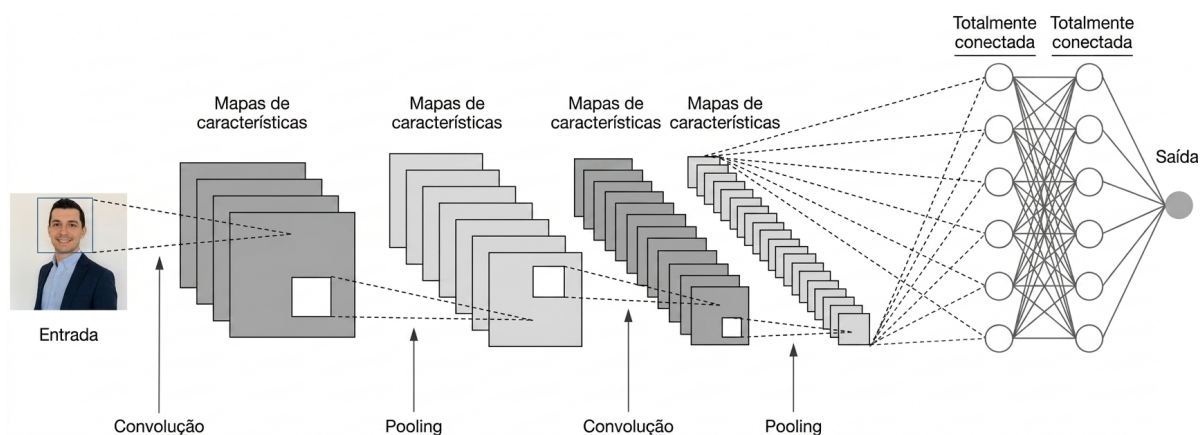
As redes neurais convolucionais (CNNs) são análogas às redes neurais artificiais (ANNs) tradicionais, pois ambas são compostas por neurônios que se auto-otimizam por meio da aprendizagem. Cada neurônio ainda recebe uma entrada e realiza uma operação (como um produto escalar seguido de uma função não linear) – a base de inúmeros ANNs. Desde os vetores brutos da imagem de entrada até a pontuação final da classe de saída, toda a rede expressa uma única função de pontuação perceptual (o peso). A última camada contém as funções de perda para as classes, e todas as dicas e truques desenvolvidos para

ANNs tradicionais ainda se aplicam (O'SHEA; NASH, 2015).

A única diferença notável entre CNNs e ANNs tradicionais é que as CNNs são usadas principalmente para reconhecimento de padrões em imagens. Isso nos permite codificar características específicas da imagem na arquitetura, tornando a rede mais adequada para tarefas focadas em imagens, além de reduzir ainda mais o número de parâmetros necessários para configurar o modelo (BEZDAN, 2019).

As CNNs partem do princípio de que a entrada será composta por imagens, isso direciona a arquitetura para ser configurada da maneira que melhor atenda à necessidade de lidar com o tipo específico de dados. Uma das principais diferenças é que as camadas da CNN são compostas por neurônios organizados em três dimensões: as dimensões espaciais da entrada (altura e largura) e a profundidade. A profundidade não se refere ao número total de camadas na ANN, mas sim à terceira dimensão de um volume de ativação. Ao contrário das ANNs padrão, os neurônios de qualquer camada específica se conectam apenas a uma pequena região da camada anterior (O'SHEA; NASH, 2015). As CNNs são compostas por três tipos de camadas: convolucionais, de pooling e totalmente conectadas. Quando essas camadas são empilhadas, forma-se uma arquitetura de CNN, como pode ser observado na Figura 10.

Figura 10 – *Convolutional Neural Networks*.



A operação de convolução (YAMASHITA et al., 2018) é um dos blocos de construção fundamentais de uma rede neural convolucional. Os parâmetros da camada convolucional consistem em um conjunto de filtros treináveis (kernels). Cada filtro tem uma pequena extensão espacial (em largura e altura), mas se estende por toda a profundidade do volume de entrada. Durante a propagação direta, cada filtro realiza uma convolução

no volume de entrada ao longo de sua largura e altura, calculando os produtos escalares entre o filtro e a entrada em cada posição (BEZDAN, 2019). Essa operação é seguida de uma função de ativação não linear (sigmoide, ReLU etc.), e as saídas obtidas são chamadas de mapas de características. O mapa de características (também chamado de mapa de ativação) representa a resposta do filtro em cada posição no espaço. Em seguida, empilhamos esses mapas de ativação ao longo da dimensão de profundidade, gerando o volume de saída (O'SHEA; NASH, 2015; BEZDAN, 2019).

As CNNs frequentemente utilizam uma camada de pooling após as camadas convolucionais. Essa camada realiza uma subamostragem nas dimensões espaciais da entrada, reduzindo ainda mais o número de parâmetros nessa etapa. Dois tipos comuns de camadas de pooling são o max pooling e o average pooling, que utilizam os valores máximos e médios, respectivamente (BEZDAN, 2019).

Os mapas de características produzidos pela última camada convolucional ou de pooling são geralmente convertidos em um vetor unidimensional e ligados a uma ou mais camadas totalmente conectadas (densas), nas quais cada entrada se conecta a cada saída por meio de pesos ajustáveis durante o treinamento. Essas camadas desempenham a mesma função que as de redes neurais artificiais (ANNs) padrão, mapeando as características extraídas e subamostradas para as saídas finais da rede, como probabilidades de classe em tarefas de classificação. A camada totalmente conectada final geralmente contém o mesmo número de nós de saída que o de classes, e funções de ativação não lineares, como a ReLU, são comumente aplicadas entre as camadas para melhorar o desempenho (O'SHEA; NASH, 2015).

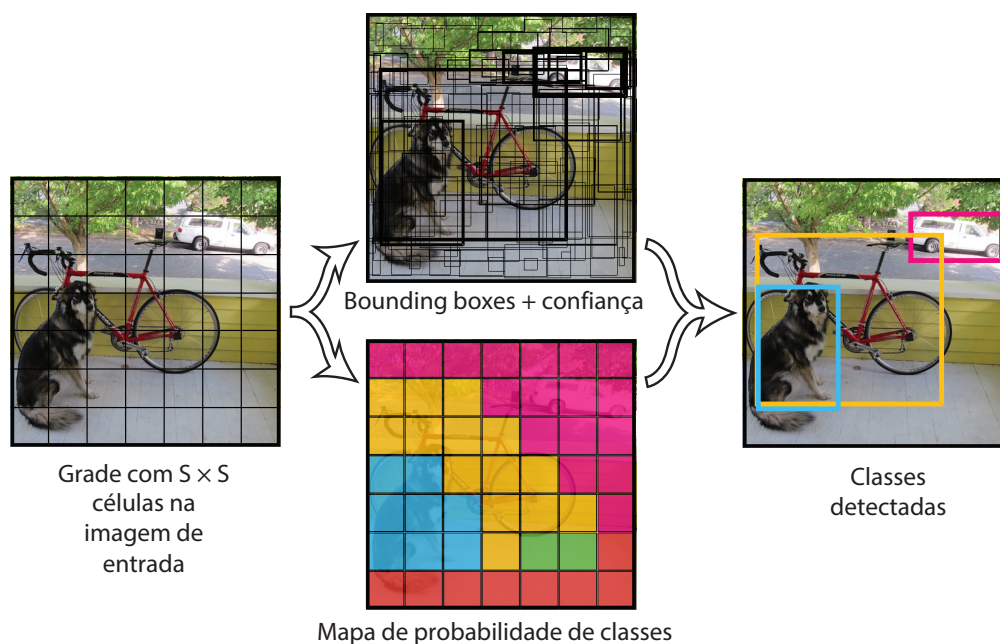
3.7 Rede YOLO (You Only Look Once)

O YOLO é uma família de modelos de redes neurais convolucionais projetada para detecção de objetos em tempo real. Ao contrário das abordagens tradicionais de detecção de imagens que operam em diferentes estágios, o YOLO aborda a detecção como um problema de regressão e prevê diretamente as classes e as localizações dos objetos em uma única iteração por imagem. Esse método permite detecção em alta velocidade, tornando o YOLO adequado para aplicações em tempo real.

A rede YOLO divide a imagem de entrada em uma grade de células, em que cada célula é responsável por prever diversas *bounding boxes* e suas respectivas probabilidades

de conter um objeto. Se houver um objeto ou não, o valor de confiança (probabilidade) da *bounding box* obtida deve refletir o grau de correspondência com a região de interesse. A confiança em si é calculada pela equação $Pr(Object) \times IOU$, em que $Pr(Object)$ é a probabilidade de existir um objeto naquela célula e IOU é a interseção sobre a união entre a *bounding box* obtida na predição e a *bounding box* verdadeira (TAN; STEINBACH; KUMAR, 2016). Se o valor de confiança for acima de um limiar estipulado, a rede preverá qual tipo de objeto existe naquela célula, gerando, assim, um mapa de probabilidade de classes. Além disso, em cada célula, são previstas C probabilidades condicionais de classe, calculadas por $Pr(Classe_i|Objeto)$, onde i varia de 1 a C . Por último, as probabilidades de classe são multiplicadas pelo valor de confiança de cada *bounding box* para obter os valores de confiança de uma só classe para cada *bounding box*. Esse valor representa tanto a probabilidade da existência de tal classe naquela região quanto o quão bem a região prevista se encaixa no objeto. As etapas do processamento da YOLO podem ser observadas na Figura 11.

Figura 11 – Etapas da rede YOLO.



Fonte: Imagem de (COSTA et al., 2022).

A YOLO é conhecida por sua capacidade de generalizar melhor para novos cenários, graças ao processamento de toda a imagem durante a predição, em vez de apenas de regiões de interesse. Isso melhora a precisão ao capturar o contexto global e reduz

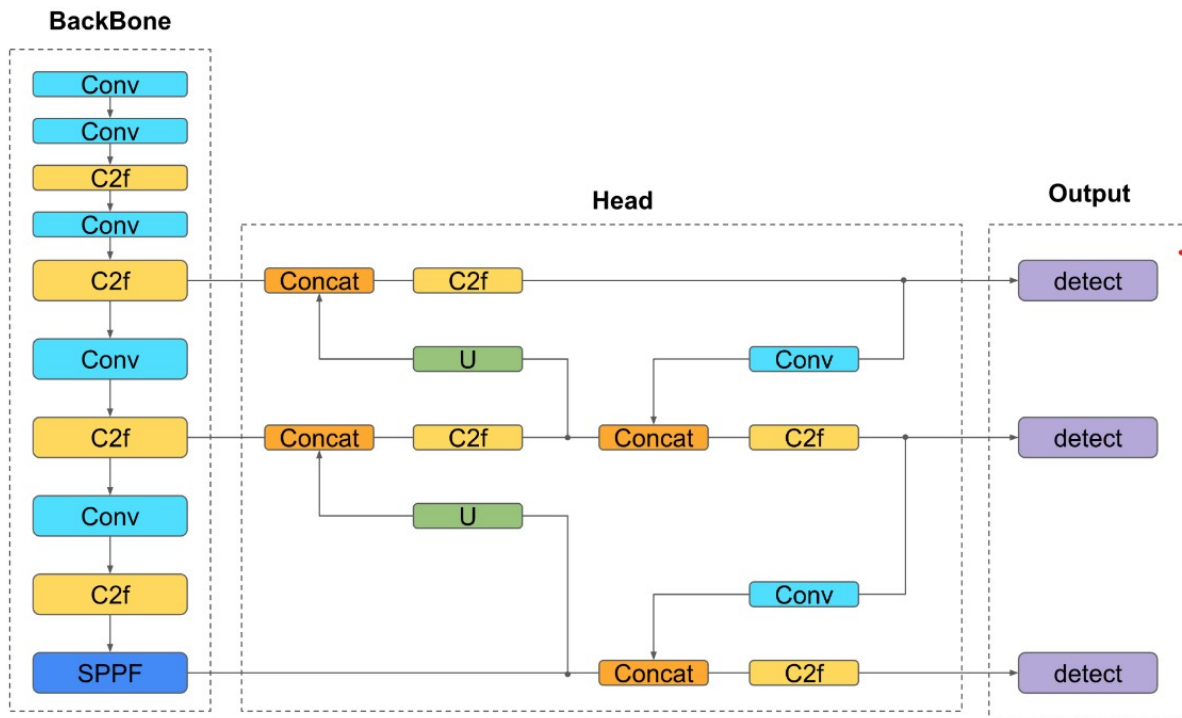
falsos positivos. As versões utilizadas na pesquisa foram a 8^a, a 10^a e a 11^a, escolhidas por serem versões mais atuais e robustas da YOLO, bem como por apresentarem diferenças suficientes entre si para possibilitar a comparação.

3.7.1 YOLOv8

A arquitetura YOLOv8 é composta por duas partes principais, o backbone e a cabeça, em que ambas utilizam uma rede neural convolucional completa. A nova backbone é uma versão modificada da arquitetura CSPDarknet53 (BOCHKOVSKIY; WANG; LIAO, 2020), constituída por 53 camadas convolucionais, que emprega uma técnica chamada de conexões parciais entre estágios para aprimorar a transmissão de informações entre os vários níveis da rede (SOHAN; RAM; REDDY, 2024). Esse backbone consiste em múltiplas camadas convolucionais, organizadas em sequência, que extraem características relevantes da imagem de entrada, reduzindo gradualmente sua resolução espacial e, ao mesmo tempo, aumentando o número de canais de características. Isso resulta na criação de mapas de características capazes de detectar objetos em diferentes escalas e resoluções (REIS et al., 2023; SOHAN; RAM; REDDY, 2024).

Já a cabeça recebe os mapas de características produzidos pela backbone e os processa ainda mais, fornecendo a saída final do modelo em forma de caixas delimitadoras e classes de objetos. No YOLOv8, a camada da cabeça é separável e capaz de gerenciar detecção, classificação e regressão de forma independente. Essa abordagem permite que cada ramo se concentre em sua própria tarefa, ao mesmo tempo em que melhora a precisão geral do modelo (SOHAN; RAM; REDDY, 2024). As camadas U (camadas de upsampling) na Figura 12 aumentam a resolução dos mapas de características (RANGEKING; JOCHER, 2023), agregando características de diferentes níveis da rede por meio de *skip connections* (HE et al., 2016). Ao fazer isso, a rede consegue capturar melhor as características em múltiplas escalas e resoluções, o que é crucial para detectar com precisão objetos de diferentes tamanhos e formatos (REIS et al., 2023).

Figura 12 – Arquitetura YOLOv8.



Fonte: Imagem de (REIS et al., 2023).

Por fim, o módulo de detecção utiliza um conjunto de camadas convolucionais e lineares para mapear as características de alta dimensionalidade das bounding boxes e das classes de objetos de saída. Toda a estrutura foi projetada para ser rápida e eficaz, mantendo alta precisão na detecção de objetos (SOHAN; RAM; REDDY, 2024).

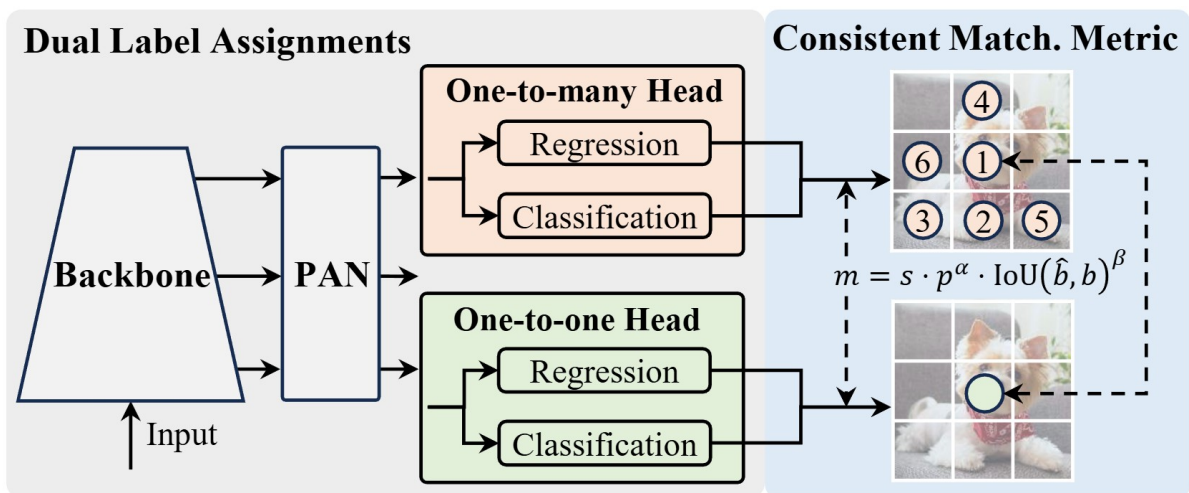
Além disso, a YOLOv8 é um modelo que não depende de âncoras. Isso significa que ele prevê o centro de um objeto diretamente, em vez de deslocar-se a partir de uma caixa de ancoragem conhecida. O uso de detecção sem âncoras reduz o número de previsões de bounding boxes, o que acelera a Supressão Não Máxima (NMS) (REIS et al., 2023), uma etapa complexa de pós-processamento que examina as detecções candidatas após a inferência (LIU; HASAN; LIAO, 2023).

3.7.2 YOLOv10

A arquitetura YOLOv10 trouxe duas principais novidades, a atribuição de rótulos duplos para treinamento sem NMS e a modelagem orientada à eficiência e precisão (ALIF; HUSSAIN, 2024). O primeiro consiste na otimização conjunta de duas cabeças durante o treinamento, uma com atribuição um para muitos e outra com atribuição um para um. A correspondência um para um atribui apenas uma previsão a cada *ground truth*, evitando

o pós-processamento de NMS, mas, no entanto, leva a uma supervisão fraca, o que reduz a precisão e a velocidade de convergência (WANG et al., 2024). Essa deficiência pode ser compensada por meio da atribuição um para muitos (CHEN et al., 2026), que permite múltiplas previsões a partir de um *ground truth*, melhorando a sensibilidade, enquanto a atribuição um para um garante a precisão, selecionando a melhor previsão (HUSSAIN, 2024), combinando o melhor das duas estratégias, sendo assim denominada atribuição de rótulos duplos (Figura 13).

Figura 13 – Atribuição de rótulos duplos.



Fonte: Imagem de (WANG et al., 2024).

Por outro lado, a modelagem orientada à eficiência e à precisão se resume à otimização de diversos componentes da arquitetura para minimizar o custo computacional e, ao mesmo tempo, melhorar o desempenho. Isso se faz possível mediante diversas atualizações, como a adoção de uma arquitetura leve para a cabeça de classificação, que consiste em duas convoluções separáveis em profundidade (CHOLLET, 2017) com tamanho de kernel de 3×3 seguidas por uma convolução 1×1 (WANG et al., 2024). Além disso, foi realizada a separação das informações espaciais e de canal durante o *downsampling*, otimizando o processo de extração de características, pois o modelo consegue processar os dados de entrada com maior eficiência, resultando em melhor desempenho e menor custo computacional.

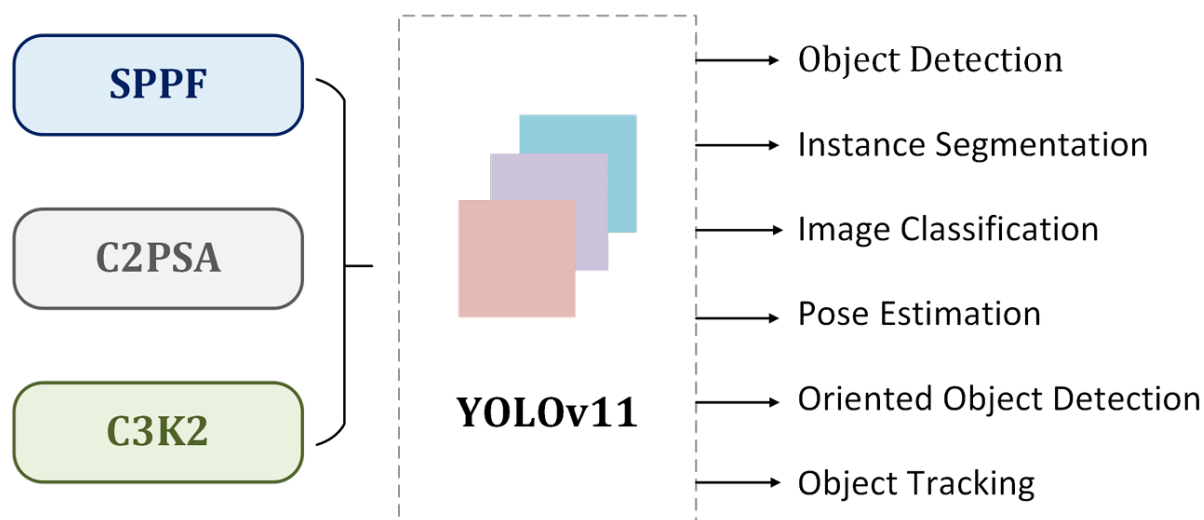
O design de blocos guiado por classificação também simplifica a arquitetura geral, utilizando informações da classificação para orientar a seleção de características importantes, garantindo que o modelo se concentre nos aspectos mais relevantes dos dados de entrada (HUSSAIN, 2024). Adicionalmente, a YOLOv10 incorpora convoluções de

kernel grande para ampliar a capacidade do modelo de captar características detalhadas em regiões espaciais maiores, permitindo que o modelo compreenda melhor o contexto dos objetos em uma imagem e aprimorando a precisão da detecção. Por último, o módulo de autoatenção parcial aumenta a precisão com um custo computacional mínimo, pois ajuda o modelo a concentrar-se nas características relevantes dos dados de entrada, melhorando a sua capacidade de detectar e classificar objetos com maior precisão.

3.7.3 YOLO11

A YOLO11 amplia e aprimora a base estabelecida pela YOLOv8, introduzindo inovações na arquitetura e otimizações de parâmetros para alcançar um desempenho de detecção superior, conforme ilustrado na Figura 14, tornando-se uma escolha versátil para uma ampla gama de tarefas de visão computacional (ULTRALYTICS, 2024; KOTTHAPALLI; RAVIPATI; BHATIA, 2025).

Figura 14 – Principais módulos da arquitetura da YOLO11.



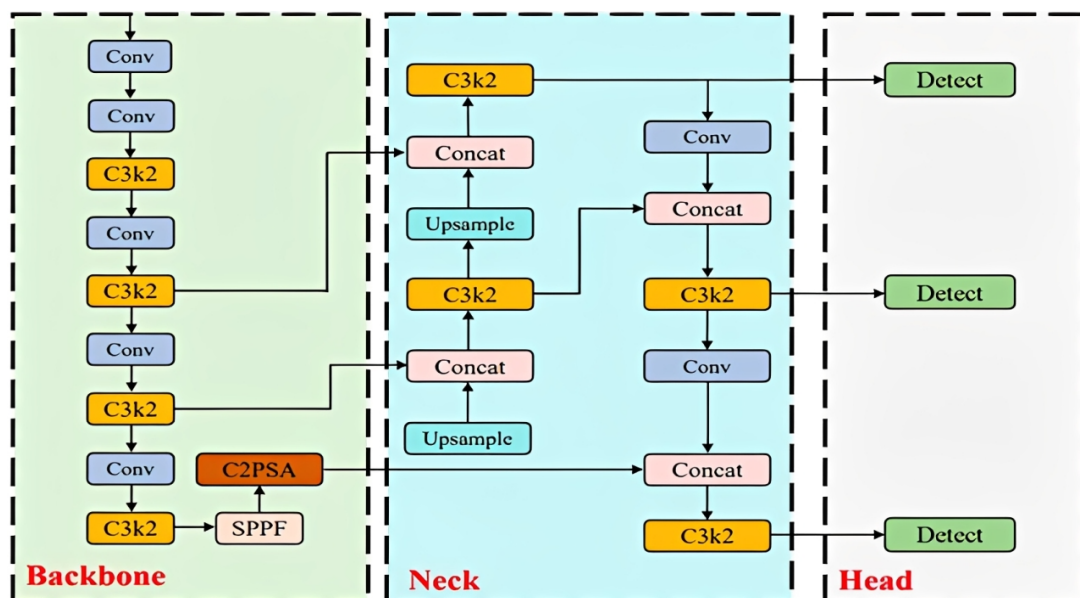
Fonte: Imagem de (KHANAM; HUSSAIN, 2024).

Uma dessas inovações é a extração de características aprimorada, que incorpora uma arquitetura de backbone e pescoço, incluindo componentes como o bloco C3k2, o Spatial Pyramid Pooling-Fast (SPPF) e um novo Cross Stage Partial with Spatial Attention (C2PSA), como pode ser observado na Figura 15, o que facilita a extração de características para tarefas complexas (ULTRALYTICS, 2024; KOTTHAPALLI; RAVIPATI; BHATIA, 2025). O bloco C3k2 é uma implementação computacionalmente mais eficiente do gargalo Cross Stage Partial (CSP). Ele emprega duas convoluções menores em vez de uma única

convolução grande, como observado no YOLOv8 (REIS et al., 2023). O “k2” em C3k2 indica um tamanho de kernel menor, o que contribui para um processamento mais rápido, mantendo o desempenho.

Já o bloco SPPF é mantido das versões anteriores, mas introduz um novo bloco, C2PSA, após ele (ULTRALYTICS, 2024). O bloco C2PSA é uma adição notável que aprimora a atenção espacial nos mapas de características. Esse mecanismo de atenção espacial permite que o modelo se concentre com maior eficácia em regiões importantes da imagem. Ao agrupar características espacialmente, o bloco C2PSA permite que o YOLO11 se concentre em áreas específicas de interesse, o que potencialmente melhora a precisão na detecção de objetos de tamanhos e posições variados (KHANAM; HUSSAIN, 2024).

Figura 15 – Arquitetura YOLOv11.



Fonte: Imagem de (ELIWA; EL-HAFEEZ, 2025).

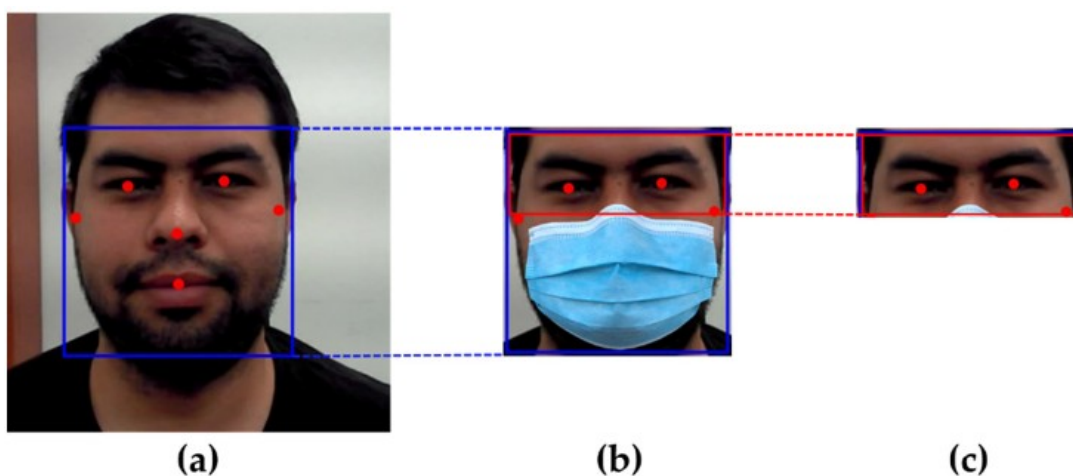
Já a cabeça da arquitetura é formada por várias camadas CBS (Convolution-BatchNorm-Silu) [19] após os blocos C3k2. Essas camadas refinam ainda mais os mapas de características, extraíndo características relevantes para a detecção precisa de objetos, estabilizando e normalizando o fluxo de dados por meio da normalização em lote e do uso da função de ativação Sigmoid Linear Unit (SiLU) para reduzir a não linearidade, o que melhora o desempenho do modelo. Os blocos CBS servem como componentes fundamentais tanto na extração de características quanto no processo de detecção, garantindo que os mapas de características refinados sejam passados às camadas subsequentes para previsões de caixas delimitadoras e de classificação (KHANAM; HUSSAIN, 2024).

3.8 MediaPipe

O MediaPipe é um framework desenvolvido para resolver dois grandes desafios do desenvolvimento de aplicações que empregam entradas perceptivas (vídeos, áudio ou imagens): a forte dependência entre as etapas de processamento e a dificuldade de desenvolvimento multiplataforma (LUGARESI et al., 2019a; LUGARESI et al., 2019b). Ele consiste em uma biblioteca multiplataforma que oferece soluções personalizáveis de aprendizado de máquina para o processamento de imagens e vídeo em tempo real. O MediaPipe inclui diversos modelos de aprendizado de máquina, cada um abordando um problema específico de segmentação de imagens ou de detecção de objetos (THAMAN; CAO; CAPORUSSO, 2022; JAKHETE; KULKARNI, 2024).

Em nosso trabalho, utilizou-se o modelo de detecção facial (GOOGLE, 2023), que oferece uma solução para a detecção de 6 pontos de referência faciais (Figura 16) e de suas respectivas bounding boxes, com suporte a múltiplos rostos simultâneos (MUKHIDDINOV et al., 2023). É baseado no BlazeFace, um detector de rostos leve e de alto desempenho, desenvolvido especificamente para inferência em dispositivos móveis. O BlazeFace (BAZAREVSKY et al., 2019) utiliza uma rede de extração de características leve, inspirada no MobileNetV1 e V2 (HOWARD et al., 2017; SANDLER et al., 2018), com um esquema de ancoragem otimizado para GPUs, adaptado do Single Shot MultiBox Detector (SSD) (LIU et al., 2016), e uma estratégia aprimorada de resolução de empates como alternativa à supressão não máxima.

Figura 16 – Detecção de face do MediaPipe.



Fonte: Imagem de (MUKHIDDINOV et al., 2023).

3.9 Séries Temporais

Uma série temporal é frequentemente o resultado da observação de um processo subjacente, no qual os valores são coletados a partir de medições realizadas em instantes de tempo uniformemente espaçados, conforme uma taxa de amostragem determinada. Uma série temporal pode, portanto, ser definida como um conjunto de instantes de tempo contíguos. A série pode ser univariada, como na Equação 3.1, ou multivariada, quando várias séries abrangem simultaneamente múltiplas dimensões no mesmo intervalo de tempo (ESLING; AGON, 2012). Exemplos dessas séries podem ser observados na Figura 17.

$$T = (t_1, \dots, t_n), \quad t_i \in \mathbb{R} \quad (3.1)$$

Uma série temporal pode ser contínua ou discreta (Figura 18). Em uma série temporal contínua, as observações são medidas a cada instante, enquanto uma série temporal discreta contém observações medidas em pontos discretos no tempo. Por exemplo, leituras de temperatura, vazão de um rio, concentração em um processo químico, etc., podem ser registradas como uma série temporal contínua. Por outro lado, a população de uma determinada cidade, a produção de uma empresa e as taxas de câmbio entre duas moedas podem ser representadas por séries temporais discretas. Normalmente, em uma série temporal discreta, as observações consecutivas são registradas em intervalos de tempo igualmente espaçados, como de uma hora, um dia, uma semana, um mês ou um ano (ADHIKARI; AGRAWAL, 2013). Conforme mencionado por Hipel e McLeod (1994), assume-se que a variável observada em uma série temporal discreta seja medida como variável contínua na escala dos números reais. Além disso, uma série temporal contínua pode ser facilmente transformada em uma discreta por meio da fusão de dados em um intervalo de tempo especificado.

Figura 17 – Exemplo de Séries Temporais.

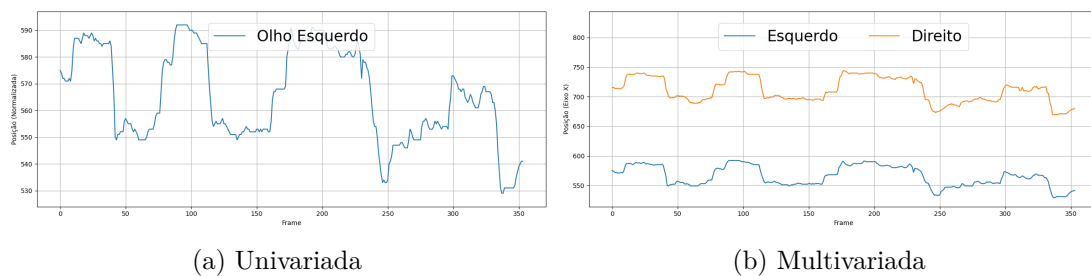
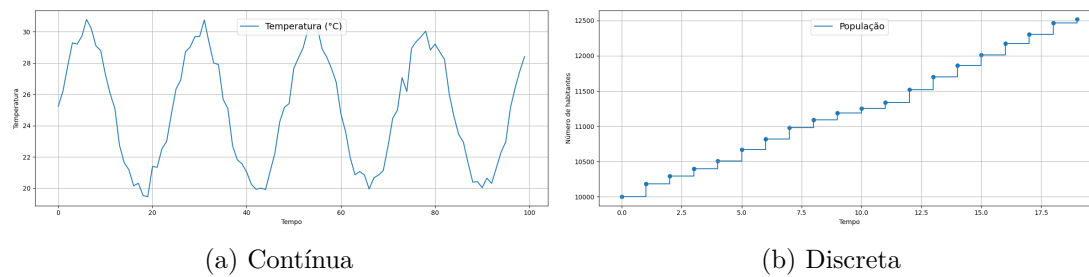


Figura 18 – Exemplo de Séries Temporais.



3.9.1 Interpolação

Geralmente, na prática, séries incompletas são frequentes (BEVERIDGE, 1992). Por diversos motivos, geralmente existem lacunas nos dados de séries temporais, intervalos de tempo irregulares nas gravações ou pontos de dados removidos que frequentemente precisam ser preenchidos para análise de dados, calibração de modelos ou para dados com intervalos de tempo regulares. Para preencher lacunas em séries temporais, há inúmeros métodos, entre eles a interpolação (LEPOT; AUBIN; CLEMENS, 2017).

De acordo com (BEVERIDGE, 1992), uma técnica de interpolação útil deve atender a quatro critérios: (1) não é necessário um grande volume de dados para preencher os valores faltantes; (2) a estimação dos parâmetros do modelo e dos valores faltantes deve ser permitida simultaneamente; (3) o cálculo de séries temporais extensas deve ser eficiente e rápido; e (4) a técnica deve ser aplicável a séries temporais estacionárias⁵ e não estacionárias⁶. O método selecionado também deve ser preciso e robusto.

Dentre as diversas técnicas de interpolação, a escolhida foi a interpolação linear, por ser uma versão simples e confiável (CHEN; FAN; SUN, 2012). Este método procura uma linha reta que passe pelos pontos finais x_A e x_B . Existem várias equações equivalentes a este método, como (GNAUCK, 2004), apresentada na Equação 3.2, e (SCHLEGEL; KORN; SCHEUERMANN, 2012), apresentada na Equação 3.3, em que α é o fator de interpolação, que varia de 0 a 1. Gnauck (2004) demonstrou que este método é eficiente e, na maioria das vezes, é melhor do que as interpolações não lineares para prever valores ausentes em fenômenos ambientais com taxas constantes.

$$X_i = \frac{x_A - x_B}{a - b}(i - b) + x_B \quad (3.2)$$

⁵ são séries cujas propriedades estatísticas, como média, variância e autocovariância, permanecem constantes ao longo do tempo

⁶ série cujas propriedades mudam ao longo do tempo

$$X_i = (1 - \alpha)x_B + \alpha x_A \quad (3.3)$$

3.9.2 Filtro de Hampel

Criado por Hampel (1974), o filtro de Hampel é aplicado a uma série temporal para identificar ruídos e substituí-los por valores mais consistentes. O filtro aplica uma janela deslizante de tamanho k , na qual calcula a mediana m_k e o desvio padrão ajustado S_k . Quando o valor central x_k se desvia da mediana em mais do que um limiar definido pelo parâmetro T , o filtro o classifica como ruído e o substitui pela mediana. O valor atualizado y_k no centro da janela é determinado pela Equação 3.4.

$$y_k = \begin{cases} x_k, & \text{se } |x_k - m_k| \leq T \times S_k \\ m_k, & \text{se } |x_k - m_k| > T \times S_k \end{cases} \quad (3.4)$$

Para este filtro, o desvio padrão S_k de cada janela é estimado de forma diferente, usando o desvio médio absoluto (DMA) (PEARSON et al., 2016). O DMA é o módulo da diferença entre a mediana m_k e cada elemento da janela deslizante, resultando em um novo subconjunto. A mediana deste novo subconjunto é calculada multiplicando-a por um fator predefinido, como se observa na Equação 3.5.

$$S_k = 1.4826 \times \text{mediana}_{j \in [-K, K]} \{|x_{k-j} - m_k|\}, \quad K = \frac{k}{2} \quad (3.5)$$

3.10 Modelo K Neighbors TsLearn

Nesta seção, será descrito o modelo de classificação do método proposto, explicando os conceitos básicos do *K-Nearest Neighbors*, as métricas de distância aplicadas e as diferenças em relação à versão do KNN TsLearn.

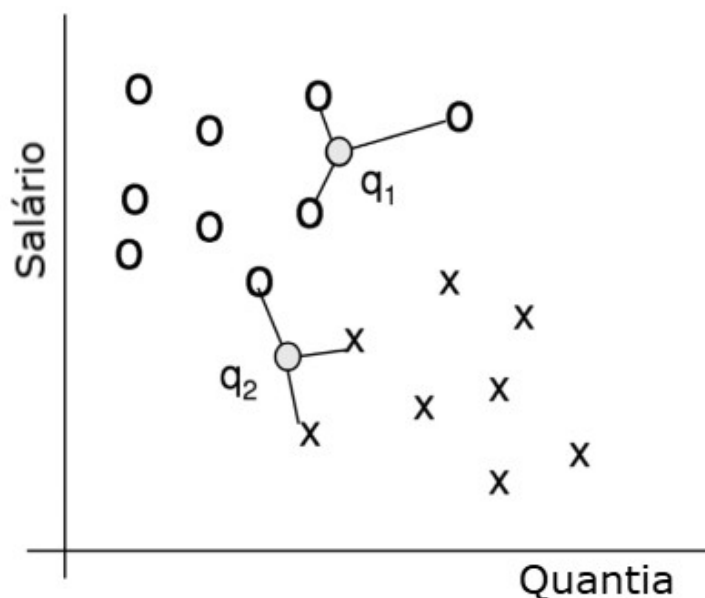
3.10.1 K-Nearest Neighbors

No ano de 1968, Cover e Hart (1967) propôs o algoritmo K-Nearest Neighbors (KNN), que pode ser calculado com base na distância euclidiana (DASARATHY, 1991). O algoritmo classifica cada novo objeto avaliando os rótulos de seus k vizinhos mais próximos no espaço de características. Em termos simples, se uma instância de dados x possui k

vizinhos mais próximos e a maioria desses vizinhos compartilha o mesmo rótulo y , então x é atribuído à classe y (KATARIA; SINGH, 2013).

Essa ideia básica é ilustrada na Figura 19, na qual se emprega um classificador KNN em um problema de duas classes em um espaço de características bidimensional. Neste exemplo, a decisão para q_1 é direta: todos os três vizinhos mais próximos são da classe O, portanto, é classificado como O. A situação para q_2 é um pouco mais complexa, pois possui dois vizinhos da classe X e um da classe O. Isso pode ser resolvido por votação majoritária simples ou por votação ponderada por distância. Assim, a classificação KNN possui duas etapas: a primeira é a determinação dos k vizinhos mais próximos e a segunda é a determinação da classe com base nesses vizinhos (CUNNINGHAM; DELANY, 2021).

Figura 19 – Exemplo de classificação por KNN.



Fonte: Adaptado de (CUNNINGHAM; DELANY, 2021).

Este método é simples de implementar e não requer uma fase de treinamento explícita, pois armazena todas as instâncias de treinamento e toma decisões apenas no momento da predição. O desempenho do KNN depende de fatores como a escolha de k , a métrica de distância e a estrutura dos dados. De uma perspectiva teórica, valores menores de k tendem a criar limites de decisão mais flexíveis, capturando padrões sutis, mas tornando-os mais sensíveis a ruídos. No geral, valores maiores de k produzem limites de decisão mais suaves e melhor generalização, mas podem negligenciar padrões locais (CUNNINGHAM; DELANY, 2020). Como o KNN baseia sua decisão em amostras próximas, em vez de um modelo predefinido, é particularmente eficaz em situações em que os limites de classe

são irregulares ou sobrepostos. No entanto, pode apresentar dificuldades com conjuntos de dados desbalanceados e de alta dimensionalidade se não for devidamente ajustado (KATARIA; SINGH, 2013).

Sabe-se que a classificação ou regressão de um novo ponto no algoritmo KNN baseia-se nos seus vizinhos mais próximos. Portanto, as métricas de distância são essenciais para seu funcionamento, pois definem como a similaridade entre uma nova instância de dados e os exemplos do conjunto de treinamento é calculada (CHOMBOON et al., 2015). Assim, a escolha da métrica de distância influencia diretamente quais pontos serão considerados mais próximos e, conseqüentemente, o resultado final do modelo. Logo, nas próximas seções serão explicadas as distâncias empregadas nos modelos KNN utilizados.

3.10.2 Distância Euclidiana

A distância euclidiana é uma medida que determina a distância entre dois pontos (CHOMBOON et al., 2015), é o comprimento do segmento de reta que os conecta (KATARIA; SINGH, 2013), como pode ser observado na Equação 3.6.

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.6)$$

Portanto, a métrica de distância euclidiana pode ser calculada da seguinte forma, se dois vetores x_i e x_j forem dados, onde $x_i = (x_i^1, x_i^2, x_i^3, \dots, x_i^n)$ e $x_j = (x_j^1, x_j^2, x_j^3, \dots, x_j^n)$, a diferença entre x_i e x_j é dada pela Equação 3.7. Dessa forma, medindo a similaridade ponto a ponto ao longo do tempo entre as séries temporais, ou seja, quanto mais similares forem duas séries, menor será a distância euclidiana entre elas.

$$D(x_i, x_j) = \sqrt{\sum_{k=1}^n (x_i^k - x_j^k)^2} \quad (3.7)$$

3.10.3 Dynamic Time Warping

O algoritmo de *Dynamic Time Warping* (DTW) é uma das funções de similaridade (ou distância) na análise de séries temporais (KEOGH; RATANAMAHATANA, 2005). Além disso, o DTW pode ser usado para medir a distância entre séries temporais de diferentes comprimentos (TAO et al., 2023). Devido ao fato de duas séries temporais poderem ser fundamentalmente semelhantes, mas defasadas ou ligeiramente distorcidas, a

ideia é permitir que o eixo do tempo seja deformado para identificar semelhanças subjacentes e obter o melhor mapeamento (CUNNINGHAM; DELANY, 2020; CUNNINGHAM; DELANY, 2021).

A distância DTW é definida pela Equação 3.8, em que $\pi = [\pi_1, \dots, \pi_l, \dots, \pi_L]$ é o melhor caminho encontrado que liga os pontos das duas sequências, medindo a distância entre elas. Além disso, essa equação obedece a três propriedades.

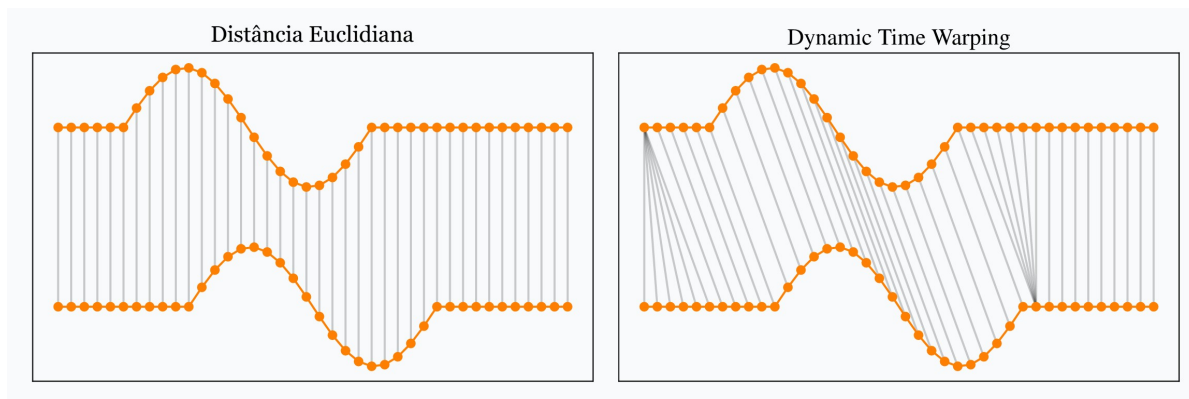
$$DTW(q, x) = \min_{\pi} \sqrt{\sum_{(i,j) \in \pi} d(q_i, x_j)^2} \quad (3.8)$$

- $m = |q|, n = |x|$
- $\pi_1 = (1, 1), \pi_L = (m, n)$
- $\pi_{l+1} - \pi_l \in \{(1, 0), (0, 1), (1, 1)\}$

Primeiramente, obtém-se o tamanho de cada sequência por meio de $m = |q|$ e $n = |x|$, ou seja, o número de pontos de cada série. A análise começa no início das duas sequências $(1, 1)$ e termina no final (m, n) , e, em cada passo, só pode andar em três direções: direita $(1, 0)$, vertical $(0, 1)$ ou diagonal $(1, 1)$ (CUNNINGHAM; DELANY, 2020). Isso garante o cálculo contínuo em frente sem voltar.

Imaginando um caminho que conecte os pontos de q aos de x , cada ligação (i, j) compara q_i com x_j , e a função $d(q_i, x_j)$ mede a distância entre eles. Somam-se, então, todas essas diferenças ao longo do caminho e, depois, extrai-se a raiz, escolhendo o que apresenta a menor diferença total (CUNNINGHAM; DELANY, 2021). É possível observar a diferença entre a distância euclidiana e o *Dynamic Time Warping* na Figura 20.

Figura 20 – Comparação distância euclidiana e DTW



Fonte: Imagem de (TAVENARD, 2021)

Porém, segundo Cunningham e Delany (2020), o DTW não é uma métrica adequada para um KNN padrão, pois não atende a dois critérios: Identidade, em que a distância é 0 apenas se os objetos forem exatamente iguais, e Desigualdade triangular, em que um caminho direto ($A \rightarrow C$) é sempre menor ou igual a um caminho que dê uma volta ($A \rightarrow B \rightarrow C$). O DTW quebra esses critérios, pois há a possibilidade de atribuir distância 0 a pontos distintos, pois ele “entorta” (*warping*) uma sequência para encaixá-la em outra mais adequada e, assim, quebra o segundo critério, em decorrência da variação de distâncias entre pontos distintos. Portanto, o DTW não é adequado para uma aplicação padrão de um modelo KNN.

O modelo K Neighbors do TsLearn (TAVENARD et al., 2020) é aplicado em nossa pesquisa devido à importância de oferecer métodos específicos para séries temporais no aprendizado de máquina. Segundo os experimentos realizados por Petitjean, Ketterlin e Gançarski (2011), a métrica tradicional de distância euclidiana não é uma medida de similaridade precisa para séries temporais, basicamente, onde a medida euclidiana padrão falha, os métodos baseados em DTW conseguem distinguir as séries temporais (SAKOE; CHIBA, 2003; CUTURI; BLONDEL, 2017). Por esse motivo, o TsLearn é utilizado em pesquisas que aplicam machine learning a séries temporais, tanto em trabalhos que abordam agrupamento, como Kuromiya (2021) e Glavan e Croitoru (2022), quanto em classificação, como, por exemplo, Nemer et al. (2024).

Assim, a implementação de K Neighbors do Tslearn é estruturada em torno de um ponto central: a operação direta sobre matrizes de distância entre séries, ao invés de trabalhar no espaço vetorial dos dados originais. No treinamento, as séries temporais são armazenadas em seu formato original, e o modelo gera uma estrutura compatível com a interface interna, sem utilizar os dados diretamente no cálculo dos vizinhos (TAVENARD et al., 2018). Quando uma métrica temporal específica (como DTW ou Soft-DTW) é aplicada, o modelo modifica seu modo de operação para empregar distâncias pré-computadas. Isso implica que, tanto no treino quanto na predição, há uma etapa para calcular a matriz de distâncias entre os dados de entrada e o conjunto de treinamento, utilizando funções otimizadas que suportam paralelização e parâmetros específicos da métrica (TAVENARD et al., 2018).

Uma vez adquirida essa matriz, o algoritmo de vizinhança trabalha diretamente nela para identificar os k vizinhos mais próximos em complexidade linear, seguido de

uma ordenação local para priorizar as menores distâncias. Esse processo é repetido a cada predição, sem a construção de estruturas auxiliares de busca, como árvores espaciais, nem a reutilização de cálculos anteriores. Além disso, o modelo armazena internamente os dois formatos de dados, a forma temporal original e sua forma auxiliar, permitindo alternar entre métricas temporais e tradicionais. Essa metodologia torna o K Neighbors TsLearn altamente especializado para séries temporais, com forte controle sobre o cálculo de distâncias e foco no desempenho numérico, ao custo de maior uso de memória e de computações frequentes (TAVENARD et al., 2018).

3.11 Métricas de Avaliação

Para analisar o desempenho dos experimentos de rastreamento ocular e de classificação de séries temporais, usaram-se as métricas de precisão, acurácia, sensibilidade, F1-Score, AUC (Area Under the Curve), especificidade e mAP (POWERS, 2020; MARIANO, 2021; LI, 2024). Certos termos utilizados no cálculo do desempenho de modelos de classificação são: Verdadeiro Positivo (VP), que ocorre quando o modelo prediz corretamente um caso positivo; Verdadeiro Negativo (VN), que denota a predição correta de um caso negativo; o Falso Positivo (FP), que acontece quando o modelo prediz positivo, mas o caso é negativo; e o Falso Negativo (FN), que significa que o modelo prediz negativo, mas o caso é positivo.

A acurácia quantifica o número de pacientes corretamente classificados. A acurácia é dada em função dos Verdadeiros Positivos **VP**, Verdadeiros Negativos **VN**, Falsos Positivos **FP** e Falsos Negativos **FN**, conforme mostra a Equação 3.9:

$$Acurácia = \frac{VP + VN}{VP + FP + VN + FN} \quad (3.9)$$

Já a precisão indica quantos dos positivos previstos são realmente positivos, conforme a Equação 3.10: é dada em função de Verdadeiros Positivos (**VP**) e Falsos Positivos (**FP**) (POWERS, 2020).

$$Precisão = \frac{VP}{VP + FP} \quad (3.10)$$

A sensibilidade avalia quão bem os resultados positivos foram obtidos, ou seja, em relação à classificação da rede neural, indica quais dentre todos os olhos existentes

na base de dados a YOLO conseguiu identificar corretamente (MARIANO, 2021). Ela é dada em função dos Verdadeiros Positivos (**VP**) e dos Falsos Negativos (**FN**), conforme mostra a Equação 3.11.

$$Sensibilidade = \frac{VP}{VP + FN} \quad (3.11)$$

O F1-Score é uma métrica usada para avaliar o desempenho de um modelo de classificação. Ela é dada em função de Verdadeiros Positivos (**VP**), Falsos Positivos (**FP**) e Falsos Negativos (**FN**), combinando as medidas de precisão e sensibilidade em uma única métrica (MARIANO, 2021). Ela é definida como a média harmônica dessas métricas, conforme mostra a Equação 3.12.

$$F1 \text{ Score} = \frac{2 * Precisão * Sensibilidade}{Precisão + Sensibilidade} = \frac{2 * VP}{2 * VP + FP + FN} \quad (3.12)$$

Por outro lado, a especificidade avalia os pacientes saudáveis, quantificando o número deles classificados como tais. 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**), referentes aos pacientes saudáveis que o método classificou como doentes (POWERS, 2020). A Equação 3.13 mostra como a especificidade é calculada:

$$Especificidade = \frac{VN}{VN + FP} \quad (3.13)$$

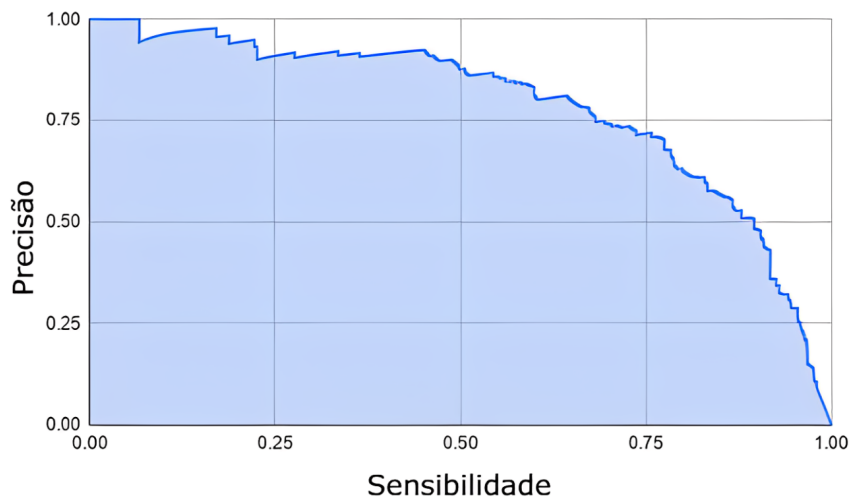
Por sua vez, o AUC (Area Under the Curve) representa a probabilidade do modelo classificar um exemplo positivo escolhido aleatoriamente com pontuação maior do que um exemplo negativo (LI, 2024). É definida em função da Taxa de Verdadeiros Positivos (**TVP**), da Taxa de Falsos Positivos (**TFP**), da Sensibilidade e da Especificidade, conforme mostra a Equação 3.14:

$$AUC = Sensibilidade (TVP) - (1 - Especificidade) (TFP). \quad (3.14)$$

Por último, a Mean Average Precision (mAP) (EVERINGHAM et al., 2015; RUSSAKOVSKY et al., 2015) é a média das precisões médias interpoladas de cada classe (CHOWDHURY, 2010). Estas precisões médias por classe são obtidas a partir da área sob

a curva da razão entre precisão e sensibilidade das detecções (HENDERSON; FERRARI, 2016), como se observa na Figura 21. Para um conjunto de dados com k classes de objetos, o mAP de um detector em um limiar de interseção sobre união (IoU) de $@t$ é definido pela Equação 3.15 (WANG, 2022). O $@t$ pode, por exemplo, ser $mAP@0.5$, que corresponde ao mAP com $\text{IoU} \geq 0.5$.

Figura 21 – Curva da razão entre precisão e sensibilidade.



Fonte: Adaptado de (WANG, 2022).

$$\text{mAP}@t = \frac{1}{k} \sum_{i=0}^{k-1} \text{AP}_i@t \quad (3.15)$$

4 MATERIAIS E MÉTODO

Esta seção descreve os materiais empregados e o método adotado na pesquisa. No texto, desenvolvem-se as bases de dados utilizadas na pesquisa, assim como é detalhado o método proposto no trabalho, seguido da explicação sobre cada uma de suas etapas.

4.1 Bases de vídeos e imagens

Para obter resultados relacionados ao objetivo estabelecido neste trabalho, foram utilizadas duas bases de dados de vídeos: uma com imagens de pacientes com estrabismo e outra com vídeos de pacientes oftalmológicos com paralisia do sexto nervo óptico.

4.1.1 Base de imagens de Estrabismo

Utilizou-se um banco de imagens privado descrito em (ALMEIDA et al., 2015), que contém 277 imagens de rostos inteiros de 45 pacientes, orientadas em direções distintas, totalizando 554 olhos. O conjunto de imagens foi então ampliado, com cada imagem espelhada e rotacionada em 90°, 180° e 270°, resultando em quatro novas imagens. Após o aumento de dados, o conjunto de imagens passou a conter 1.385 imagens. As caixas delimitadoras correspondentes à posição de cada olho foram definidas com base na marcação da íris de cada olho. Esse banco de imagens foi empregado, pois as imagens exibem o rosto inteiro do paciente, possuem as posições dos olhos anotadas e também contêm pacientes com estrabismo.

4.1.2 Base de vídeos do Sexto Nervo

O banco de dados de vídeos inclui vídeos de pacientes com paralisia do sexto nervo e de indivíduos com visão normal em ambos os olhos. Os vídeos foram gravados em uma clínica oftalmológica em São Luís, no Maranhão, Brasil, a partir de 2017 e ao longo de mais de três anos. O número reduzido de vídeos deve-se à raridade da doença entre os pacientes atendidos pelo oftalmologista. Os pacientes que concordaram em colaborar com esta pesquisa autorizaram a gravação de vídeos e assinaram um termo de consentimento livre e esclarecido. Este projeto foi autorizado pelo Comitê de Ética em Pesquisa do Hospital Universitário Federal do Maranhão, sob o parecer nº 331/09.

Além disso, Costa (COSTA et al., 2022) definiu um protocolo de aquisição de vídeo para facilitar a aplicação do método desenvolvido. Os vídeos foram gravados com a câmera de um smartphone padrão, apoiada em um tripé para garantir a estabilidade. Cada vídeo começa a ser registrado com o paciente com os olhos na posição primária do olhar (PPO) (Figura 1), ou seja, olhando para a frente. Os olhos do paciente devem estar visíveis e alinhados horizontalmente, sem a necessidade de exibir o rosto inteiro. Em seguida, solicita-se que o paciente olhe para a esquerda e, depois, para a direita, de forma semelhante a um movimento persecutório (Figura 4), repetindo-o algumas vezes, com o objetivo de analisar a movimentação horizontal dos olhos.

Existem duas versões desse conjunto de dados; a primeira, criada por Costa et al. (2022), continha 35 vídeos, dos quais dezesseis foram adquiridos antes da definição do protocolo de aquisição, dos quais 13 pertenciam a pacientes com paralisia. No entanto, como não havia um protocolo definido, o rosto do paciente não estava totalmente voltado para a frente em alguns vídeos, o que resultou em visibilidade parcial de um dos olhos.

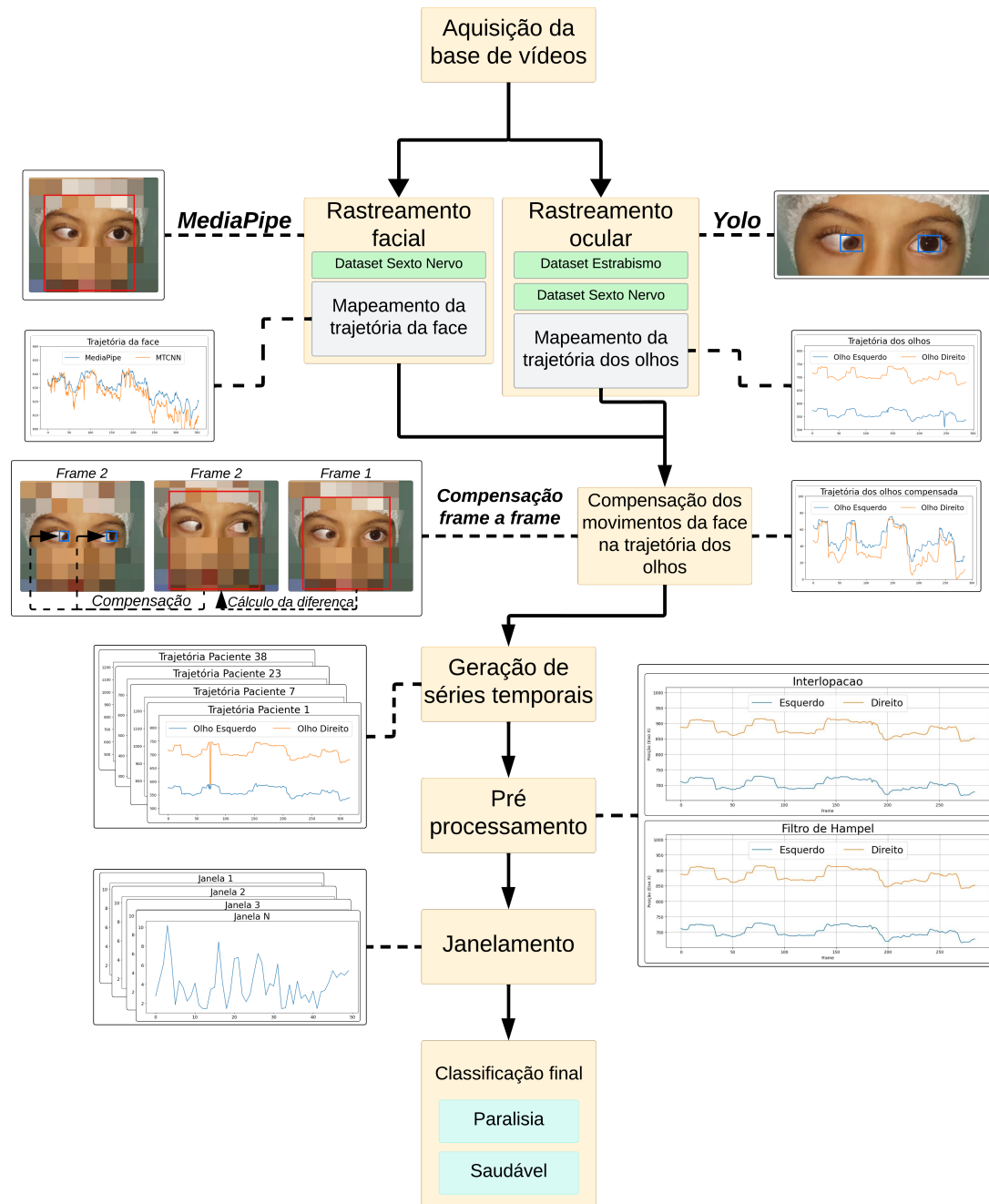
A segunda versão, criada posteriormente por nós, expandiu a base com a inclusão de quatro novos pacientes, fornecidos pelo especialista, e os adicionou ao conjunto de dados deste estudo: três com paralisia ocular e um indivíduo saudável. Essa versão do conjunto de dados contém 39 vídeos, dos quais 17 são de pacientes com paralisia e 22, de pacientes saudáveis.

4.2 Método Proposto

Nesta seção, detalham-se as etapas do método empregado na nossa pesquisa.

O método proposto (Figura 22) consiste em sete etapas principais: (1) aquisição do conjunto de dados de vídeo; (2) rastreamento ocular; (3) rastreamento facial; (4) compensação dos movimentos da face na trajetória ocular; (5) geração de séries temporais; (6) pré-processamento das séries temporais; (7) janelamento destas séries; e (8) classificação das séries temporais.

Figura 22 – Fluxograma do método proposto.



4.2.1 Rastreamento Ocular

Para realizar o rastreamento ocular em pacientes, após diversos experimentos, optou-se pela YOLOv8 (REIS et al., 2023), por ter obtido os melhores resultados gerais. A rede YOLO foi escolhida para a detecção ocular em imagens oftalmológicas devido à sua alta precisão e velocidade. A YOLO foi projetada para detecção em tempo real, o que é essencial em ambientes médicos onde a velocidade de análise pode influenciar diagnósticos e tratamentos. Além disso, a arquitetura do YOLO permite a detecção eficiente de objetos

de diferentes tamanhos e em diversos cenários, tornando-o ideal para lidar com as condições presentes em imagens oftalmológicas. A saída da rede é então utilizada para gerar as séries temporais empregadas em nossa pesquisa para a detecção da paralisia do sexto nervo.

4.2.2 Rastreamento Facial

O rastreamento facial foi realizado por meio do framework MediaPipe (LUGARESI et al., 2019a; GOOGLE, 2023). O modelo de detecção facial do MediaPipe foi escolhido, pois, semelhante a YOLO, oferece detecção em tempo real com acurácia, mesmo em situações com alta variação de iluminação, oclusão e posições da face, característica essencial para aplicações que envolvem análise contínua de sinais visuais. O ponto central da bounding box de saída do modelo, com limiar de confiança de 50%, é utilizado para gerar a série temporal do deslocamento facial do paciente, que é posteriormente empregada para compensar a trajetória dos olhos.

Essa etapa é necessária, pois alguns dos vídeos da base de dados de paralisia do sexto nervo (Seção 4.1.2) não seguem o protocolo criado por Costa (COSTA et al., 2022), logo, um certo número de vídeos apresenta pacientes que movimentam suas cabeças, tanto voluntariamente quanto involuntariamente, como, por exemplo, os pacientes 3, 6, 35 e 38. Isso afeta diretamente a informação sobre a movimentação dos olhos, pois, nos instantes do deslocamento da face, o modelo de rastreamento ocular acaba por captar tais locomoções como provenientes dos olhos, quando, na verdade, não são.

4.2.3 Compensação dos movimentos da face

A compensação baseia-se na dissociação do movimento geral da face do movimento relativo dos olhos. Primeiramente, extraem-se as coordenadas centrais dos olhos e da face, frame a frame. Em seguida, interpretando as variações sucessivas na posição horizontal do centro facial devido à locomoção da cabeça, e não aos movimentos oculares voluntários, calculou-se a disparidade entre as coordenadas faciais em frames consecutivos. O método avalia o deslocamento horizontal da cabeça ao longo do tempo.

O algoritmo compensa essa diferença subtraindo o deslocamento da cabeça calculado com base nas coordenadas dos olhos no frame atual. Esse método normaliza as posições dos olhos, removendo o impacto dos movimentos da cabeça e mantendo apenas o movimento ocular imediato. Assim, o algoritmo permite uma análise mais precisa da

locomoção ocular, pois os deslocamentos resultantes correspondem aproximadamente aos movimentos dos olhos.

4.2.4 Geração de séries temporais

As séries temporais utilizadas nesta pesquisa foram extraídas de dados de movimento ocular rastreados em vídeos de pacientes oftalmológicos, tanto saudáveis quanto com paralisia do sexto nervo óptico. Como as séries temporais consistem em dados ordenados cronologicamente, utilizados para analisar e prever o comportamento ao longo do tempo, definimos o tempo e a posição do eixo x de cada olho como referência para a coleta de dados. Foi rastreado apenas este eixo nos vídeos, uma vez que o sexto nervo afeta apenas o movimento ocular no eixo horizontal.

Os dados das séries temporais foram extraídos por meio de dois métodos. No primeiro método, foi utilizado o valor do eixo x em cada quadro como dado, denomina-se então a série resultante como Série Temporal da Posição Ocular (STPO). No segundo método, calculou-se a diferença posicional entre quadros consecutivos, considerando a diferença absoluta $abs(x_{f_n} - x_{f_{n-1}})$, em que x representa a localização do olho no n -ésimo quadro (f_n). Este método produz uma série temporal que representa o número de pixels que o olho move a cada instante, denominada Série Temporal de Movimento Ocular (STMO). Todas as séries temporais extraídas são univariadas, pois mediu-se uma única variável em pontos sequenciais no tempo, ou seja, a posição de cada olho no eixo x.

Consideraram-se os olhos esquerdo e direito de cada paciente como séries temporais distintas. Assim, para cada tipo de extração, foram obtidas 78 séries temporais de todos os pacientes. Entre os pacientes com paralisia do nervo óptico disponíveis, apenas um olho foi afetado, resultando em 17 séries temporais de nervos ópticos paralisados no conjunto de 78 pacientes. Essa questão será explorada com mais detalhes nas seções subsequentes.

4.2.5 Pré-processamento de dados

Neste trabalho, são utilizados diversos métodos de pré-processamento aplicados progressivamente. Nesta seção, cada um deles é explicado, juntamente com a justificativa de sua aplicação.

4.2.5.1 Interpolação

Após a aquisição, é necessária a interpolação de dados, pois cada vídeo tem duração diferente. Portanto, para utilizar as séries temporais adquiridas, é essencial equalizar suas durações. Para padronizar as durações, foi aplicada a interpolação linear, ajustando todas as séries para 285 frames (a média dos pacientes do banco de dados de sexto nervo), com esse valor utilizado como parâmetro na função de interpolação, juntamente com o tamanho da janela. Contudo, ao interpolar as séries temporais, são estimados valores para novos intervalos de tempo, os quais podem descaracterizar a série resultante em relação à original. Assim, nosso método de interpolação resolve esse problema ao calcular a diferença entre a soma dos dados originais e a soma interpolada. Essa diferença é então distribuída uniformemente entre os valores interpolados, ajustando as séries de modo que a soma dos valores ajustados seja igual à soma original.

4.2.5.2 Filtro de Hampel

Conforme observado por (COSTA et al., 2022), o centro da pupila nem sempre é detectado corretamente em alguns vídeos, portanto, os dados foram pré-processados para eliminar valores discrepantes. Utilizou-se, então, o filtro de Hampel, uma técnica de filtragem de séries temporais que detecta valores discrepantes e os substitui por valores mais consistentes. Conforme explicado na Seção 3.9.2, dos parâmetros necessários para a aplicação desse filtro, nos experimentos realizados, utilizaram-se os valores $k = 5$ e $T = 2$.

4.2.6 Janelamento

Devido à escassez de dados, o uso de janelas é essencial para os experimentos, o que aumenta o número de instâncias analisadas. Na análise de séries temporais, a técnica de janelamento por Sobreposição divide uma série temporal primária em múltiplas sub-séries menores. Utilizando essa abordagem, uma janela de tamanho fixo é deslizada sobre os dados da série temporal inicial, criando uma nova janela a cada passo que se sobrepõe aos dados originais por um determinado número de pontos. Utiliza-se um comprimento de janela de 50 e um intervalo de sobreposição de 40, resultando em 24 novas séries a partir de cada série original.

4.2.7 Aumento de dados

Os experimentos revelaram um desequilíbrio nas séries temporais devido ao número limitado de pacientes com paralisia (apenas 17), o que dificultaria qualquer análise devido à diferença significativa de volume em relação aos casos não patológicos. Para resolver isso, aplicou-se um aumento de dados à série de olhos paralisados, equilibrando as instâncias de dados.

Porém, os dados de séries temporais são particularmente vulneráveis a transformações decorrentes do aumento de dados (OH; HAN; JEONG, 2020). Portanto, para evitar distorções nos dados da série temporal, os métodos selecionados foram: distorção temporal (time warping), distorção de magnitude (magnitude warping), ruído gaussiano e permutação (WEN et al., 2020). A distorção temporal estica e comprime os dados ao longo de cada ponto temporal da série; de forma semelhante, a distorção de magnitude altera a amplitude dos dados em cada instante. O aumento gaussiano adiciona ruído aleatório aos dados originais, melhorando a robustez e a generalização do classificador. Além disso, a permutação aplicada inverte a série em torno do eixo x, de modo que, na nova série, o último quadro anterior passa a ser o primeiro, e assim por diante.

Com exceção da permutação, todas essas técnicas operam em torno de um desvio padrão previamente definido. Neste caso, escolhemos um valor de 0,1 para garantir que as modificações tornem cada instância suficientemente distinta, sem comprometer a integridade da série original, evitando, assim, a classificação incorreta de um paciente saudável como paralisado. Também é importante ressaltar que o aumento de dados foi aplicado exclusivamente aos dados de treinamento.

4.2.8 Classificadores de séries temporais

A biblioteca SKTime foi escolhida por sua robustez e pela ampla variedade de algoritmos para análise de séries temporais (LÖNING et al., 2019). A biblioteca oferece uma interface unificada e intuitiva para métodos de classificação avançados e simples, incluindo shapelets e modelos de aprendizado profundo. Ela se integra a outras ferramentas, como o scikit-learn, facilitando a experimentação e a implementação de diferentes modelos. Utilizaram-se seis classificadores de séries temporais dessa biblioteca: RNN, ResNet, KNeighbors, KNeighbors do TSlearn, Time Series SVC e o Shapelet Learning TSlearn. Esses classificadores foram selecionados com base em sua confiabilidade e adequação às

nossas aplicações em séries temporais.

O classificador RNN (SALEHINEJAD et al., 2017; SCHMIDT, 2019) foi escolhido por sua capacidade de capturar dependências temporais e padrões sequenciais, essenciais à análise de séries temporais. O modelo mantém um estado oculto que armazena informações de etapas anteriores, o que o torna adequado para aplicações em que a ordem dos dados é importante.

O aprendizado residual profundo do ResNet (HE et al., 2016) aborda o problema do desaparecimento do gradiente. Ao usar conexões de salto, que permitem treinar redes profundas sem sacrificar a velocidade, ele pode identificar características complexas em séries temporais.

O algoritmo K-Neighbors é eficaz quando a similaridade entre os dados pode ser medida pela proximidade, com base na ideia de que dados semelhantes tendem a pertencer à mesma classe. Por ser não paramétrico, adapta-se bem a distribuições complexas sem exigir ajustes extensivos.

O K-Neighbors TsLearn (TAVENARD et al., 2020) estende a abordagem K-Nearest Neighbors para atender às necessidades específicas de dados de séries temporais. Este classificador é útil para tarefas em que a forma e a sequência dos pontos de dados são cruciais, pois pode capturar de forma confiável similaridades entre padrões de séries temporais, usando métricas de distância especificamente projetadas para esse tipo de dados, como o Dynamic Time Warping (DTW) e o Global Alignment Kernel (GAK).

O TimeSeriesSVC (CUTURI, 2011) estende o *Support Vector Machine* (SVM) tradicional, permitindo que ele trabalhe diretamente com dados de séries temporais. Isso é alcançado pela incorporação de kernels especializados para análise de pares, como Dynamic Time Warping (DTW) ou Soft-DTW, que capturam a estrutura temporal dos dados de forma mais eficaz do que kernels vetoriais padrão. Ao contrário de um SVC convencional, o TimeSeriesSVC processa séries temporais em seu formato original, multiíndice ou 3D, e suporta sinais de comprimento variável. Ele também pode herdar as capacidades do kernel escolhido, o que o torna particularmente eficaz para problemas de classificação em que a ordem dos dados e padrões dinâmicos são cruciais para distinguir entre as classes.

O Shapelet Learning (GRABOCKA et al., 2014; YANG et al., 2016) é eficaz na classificação de séries temporais porque aprende shapelets discriminativos, pequenas subsequências que melhor separam as classes. Ele identifica tanto a forma quanto a

localização de padrões temporais relevantes diretamente a partir dos dados, tornando o modelo mais interpretável e adequado para tarefas em que a forma do padrão é importante. Seu processo de otimização por aprendizado profundo aprende, de forma conjunta, os shapelets mais informativos, suas posições e suas contribuições para a classificação, permitindo que ele supere métodos que dependem de distâncias predefinidas, como o DTW. Além disso, sua capacidade de lidar com dados multivariados e múltiplos comprimentos de shapelet a torna uma abordagem poderosa e especializada para análise de séries temporais.

5 RESULTADOS E DISCUSSÃO

Nesta seção, são relatados os resultados obtidos pelo método proposto de classificação de séries temporais de trajetórias oculares oftalmológicas, utilizando modelos de aprendizado profundo para a detecção de paralisia do sexto nervo óptico craniano. Os experimentos de cada etapa da pesquisa são descritos, e seus resultados são interpretados e discutidos, culminando no método final do trabalho.

Inicialmente, os resultados dos classificadores de séries temporais foram analisados exclusivamente com base nas trajetórias capturadas pelo modelo YOLOv3 de Costa et al. (2022), com o objetivo de determinar qual das séries temporais, a de Movimento Ocular ou a de Posição Ocular, apresentaria desempenho superior. Avaliou-se, então, a precisão das diferentes versões da rede YOLO empregadas no rastreamento dos olhos, bem como as trajetórias obtidas, utilizando o método de cálculo da velocidade média de (COSTA et al., 2022), para a escolha da melhor versão. Por último, realiza-se um experimento de classificação, aplicando validação cruzada com 5 folds, com base nas trajetórias geradas pelos modelos YOLO treinados e compensadas pelo rastreamento facial do MediaPipe, variando entre a classificação de séries temporais univariadas e multivariadas, bem como entre a classificação por série temporal e por paciente.

5.1 Experimento de classificação com trajetórias da YOLOv3

A base de dados do sexto nervo, na sua versão inicial, foi empregada neste experimento e foi proposta originalmente por (COSTA et al., 2022), e inclui vídeos de 35 pacientes oftalmológicos: 21 saudáveis e 14 acometidos por paralisia do sexto nervo. Aplicou-se a classificação por série temporal, logo, cada série é classificada individualmente, independentemente do paciente de quem ela provém. Por isso, após a geração de séries temporais conforme a Seção 4.2.4, foram obtidas 70 séries temporais no total. Após a aplicação do janelamento por sobreposição, com uma janela de tamanho 50 e um intervalo de sobreposição de 40, geraram-se 24 novas séries a partir de cada série original, resultando em um total de 1.680 séries temporais, sendo 1.008 de pacientes sem paralisia e 672 de casos positivos.

Esse conjunto de séries foi então dividido, de maneira estratificada ¹, em 70% para

¹ técnica estatística e de aprendizado de máquina que divide um conjunto de dados ou amostra em

treinamento e 30% para testes. Assim, o conjunto de 1.680 séries temporais foi dividido em 1.176 para treinamento e 504 para testes. Dessa forma, observa-se que, antes do processo de aumento de dados, o número de séries temporais criadas a partir de olhos com paralisia do nervo óptico por meio de janelamento era de 336 no total, devido à divisão estratificada e à limitação do conjunto de dados descritas na Seção 4.2.7. Dessas 336 séries, 240 são para treinamento e 96 para testes. No entanto, após a aplicação do aumento de dados, foram geradas quatro séries adicionais para cada série dos pacientes com paralisia do grupo de treinamento, resultando em um conjunto de treinamento muito mais equilibrado, com 2.136 séries temporais, das quais 1.200 provenientes de casos positivos e 936 de casos saudáveis.

Para os modelos RNN e Resnet, foram utilizados 100 épocas, tamanho do lote de 20, taxa de aprendizado de 1×10^{-4} , função de perda binary cross-entropy e otimizador Adam. Já para os modelos K Neighbors e K Neighbors TsLearn, foram usadas as distâncias euclidianas e o DTW (Dynamic Time Warping), e a otimização em grade de 5 folds para o valor de K, no espaço de busca de 1, 3, 5, 7 e 9.

Pode-se observar, com base na Tabela 1, que os modelos avaliados com séries temporais do movimento ocular apresentaram dificuldade na tarefa de classificação, o que se reflete principalmente nos baixos valores de sensibilidade e de F1-Score. De modo geral, os resultados revelam baixa capacidade discriminativa dos modelos para esse conjunto de atributos. Ainda assim, o modelo KN TsLearn, baseado em DTW, destacou-se por apresentar o melhor compromisso entre as métricas avaliadas nesse cenário, obtendo maior sensibilidade e F1-Score, o que sugere maior capacidade de identificar corretamente os casos positivos, ainda que às custas de uma redução significativa na especificidade.

Ao comparar com os resultados da Tabela 2, nota-se que, de forma geral, os experimentos realizados com as séries temporais da posição ocular (STPO) apresentaram desempenho significativamente superior. Em especial, os modelos baseados em K-Nearest Neighbors registraram ganhos expressivos nas métricas de acurácia, precisão, AUC e especificidade, indicando que as informações relacionadas à posição ocular fornecem padrões mais discriminativos para a tarefa de classificação. Além disso, os valores elevados de especificidade revelam maior capacidade dos modelos de identificar corretamente casos negativos, reduzindo a incidência de falsos positivos. O melhor resultado foi obtido

subgrupos homogêneos, garantindo que cada estrato seja representado proporcionalmente na amostra final ou nos conjuntos de treinamento e teste

pelo modelo K Neighbors TsLearn utilizando distância euclidiana, alcançando 80% de acurácia, 95% de precisão, 60% de sensibilidade, 74,5% de F1-Score, 80% de AUC e 97% de especificidade. Esses resultados demonstram não apenas um ganho de desempenho global, mas também maior estabilidade nas métricas avaliadas, reforçando o potencial das séries temporais da posição ocular como representação mais adequada para o problema investigado.

Tabela 1 – Resultado da classificação com Série Temporal do Movimento Ocular.

Classificadores	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	48%	47%	33%	39%	48%	62.5%
ResNet	52%	53%	48%	50%	52%	57%
K Neighbors (Euclidiano)	50%	50%	49%	50%	50%	51%
K Neighbors (DTW)	50%	50%	49%	50%	50%	51%
KN TsLearn (Euclidiano)	50%	50%	48%	50%	50%	51%
KN TsLearn (DTW)	58%	56%	77%	65%	58%	38%

Tabela 2 – Resultado da classificação com Série Temporal da Posição Ocular.

Classificadores	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	52%	52%	57%	55%	52%	50%
ResNet	42%	43%	52%	47%	42%	30%
K Neighbors (Euclidiano)	76%	85%	63%	73%	76%	88%
K Neighbors (DTW)	76%	85%	63%	73%	76%	88%
KN TsLearn (Euclidiano)	80%	95%	60%	74.5%	80%	97%
KN TsLearn (DTW)	75%	80%	67%	72%	75%	83%

5.2 Experimento de rastreamento ocular com modelos YOLO

Este experimento também foi realizado utilizando o dataset de (COSTA et al., 2022), porém, para realizar o rastreamento frame a frame com os modelos YOLO, cada vídeo foi transformado em um conjunto de imagens para cada paciente, aplicando-se a taxa de 30 FPS (Frames per Second), dessa forma, adquiriram-se 9.983 imagens no total.

Os modelos YOLOv8, YOLOv10 e YOLOv11 foram treinados com pesos COCO (LIN et al., 2014) por 100 épocas, mantendo as camadas entre o primeiro bloco residual da arquitetura e o bloco de output congeladas durante as primeiras 50 épocas. Essas camadas foram descongeladas a partir da época 51, o que acelerou o processo ao evitar atualizações.

Realizou-se um ajuste fino por mais 50 épocas com todas as camadas descongeladas. Essa etapa foi necessária porque os modelos YOLO empregados, treinados apenas no conjunto de dados de estrabismo, não detectaram adequadamente os olhos no segundo conjunto de dados da paralisia do sexto nervo. Para contornar essa limitação, utilizou-se

uma divisão estratificada de 30% das imagens da base de paralisia do sexto nervo para ajustar o modelo e adaptá-lo à nova condição clínica. Os 70% restantes foram utilizados para testar o desempenho do modelo ajustado.

Além disso, utilizam-se as configurações padrão de hiperparâmetros dos modelos pré-treinados (ULTRALYTICS, 2024), incluindo o otimizador Adam, uma taxa de aprendizado de 1×10^{-3} e um tamanho de lote de 16. A única exceção foi o tamanho da imagem de entrada, ajustado para múltiplos de 32, próximos às dimensões utilizadas por (COSTA et al., 2022), resultando em tamanhos de 416×416 e 128×128 pixels.

Avaliou-se o desempenho com a métrica de Precisão Média (mAP), que serve como métrica de detecção nativa para os modelos YOLO (ULTRALYTICS, 2024). Por padrão, foi relatado o mAP em três níveis de confiança: 50%, 75% e 50%-95%. Este último representa a precisão média (AP) calculada em vários limiares de Interseção sobre União (IoU), variando de 50% a 95% em incrementos de 5

As Tabelas 3 e 4 mostram que os modelos ajustados com 30% do conjunto de dados do sexto nervo apresentaram métricas promissoras. O modelo YOLOv8 treinado com uma entrada de 416×416 obteve o melhor resultado tanto de mAP50% quanto de mAP75%, enquanto o modelo YOLOv11 treinado com as mesmas dimensões de entrada alcançou a melhor métrica de mAP50-95%.

É evidente, portanto, que o ajuste fino com uma porcentagem maior beneficiou mais os modelos durante o treinamento para a detecção de olhos em pacientes. Da mesma forma, imagens de entrada de maior dimensão se mostraram mais eficientes para a extração de características, permitindo que o modelo detectasse melhor o objeto de interesse. Além disso, o modelo YOLOv8 se destacou pelas melhores métricas atingidas, apontando-o como mais promissor para as etapas seguintes.

Tabela 3 – Finetune (128 x 128)

Versão YOLO	mAP50%	mAP75%	mAP50-95%
Yolov8	91.5%	76.5%	65.2%
Yolov10	91.1%	75.8%	65.5%
Yolov11	92.8%	78.8%	67.5%

Tabela 4 – Finetune (416 x 416)

Versão YOLO	mAP50%	mAP75%	mAP50-95%
Yolov8	94.0%	86.4%	73.6%
Yolov10	89.9%	82.7%	72.9%
Yolov11	90.9%	84.7%	74.6%

Na Figura 23, verifica-se a curva de mAP, indicando boa capacidade de detecção, com bom equilíbrio geral entre precisão e sensibilidade, bem como baixo índice de falsos positivos. Porém, a queda no final da curva indica uma explosão de falsos positivos.

Já na Figura 24, observa-se a matriz de confusão, com base nela, podemos constatar que o modelo está localizando os olhos eficientemente, tendo identificado 12329 olhos corretamente, enquanto errou apenas 325. Além disso, o modelo detectou 2811 olhos onde não havia.

Figura 23 – Curva mAP do modelo YOLOv8 416x416.

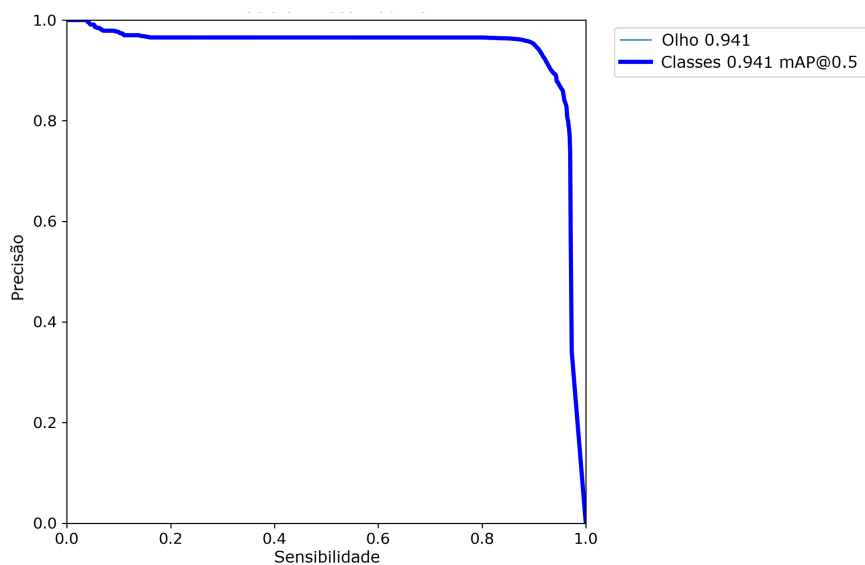
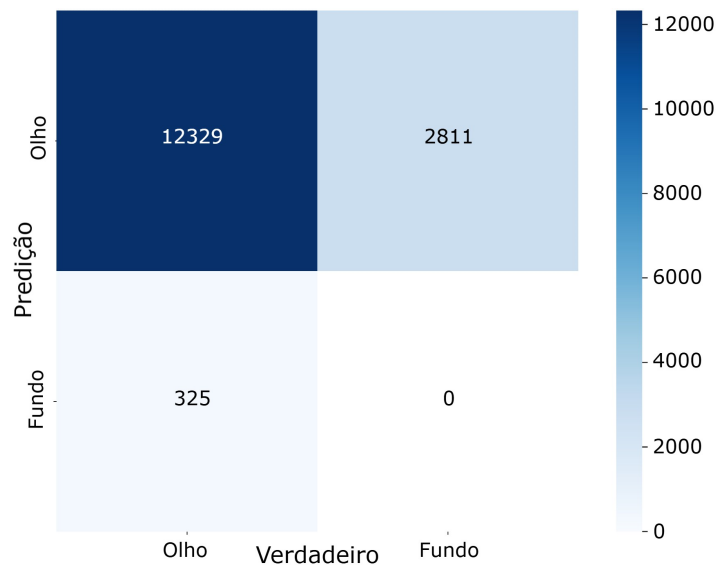


Figura 24 – Matriz de confusão da predição do modelo YOLOv8 416x416.



5.3 Experimento de classificação por velocidade média

Após a avaliação do rastreamento com a YOLO, cada um dos modelos treinados foi utilizado para rastrear os 35 pacientes da primeira versão da base de dados de paralisia do sexto nervo. Dessa forma, com as trajetórias adquiridas, calculou-se a velocidade média das trajetórias oculares dos pacientes. Ela é a razão entre o deslocamento (ΔS) e a variação do tempo (ΔT), como apresentado na Equação 5.1. Nessa metodologia, a velocidade foi calculada apenas ao longo do eixo X, uma vez que o sexto nervo craniano inerva o músculo reto lateral, responsável por um dos movimentos horizontais do olho.

$$\text{Velocidade}_{\text{média}} = \frac{\Delta S}{\Delta T} = \frac{\text{Posição}_{\text{Final}} - \text{Posição}_{\text{Inicial}}}{\text{Tempo}_{\text{Final}} - \text{Tempo}_{\text{Inicial}}} \quad (5.1)$$

Após obter a velocidade média dos olhos esquerdo e direito, calcula-se a diferença percentual entre elas, conforme mostrado na equação 5.2. Se a diferença percentual entre as velocidades dos olhos do paciente exceder o limite estimado anteriormente, relatado em (COSTA et al., 2022), considera-se que o paciente apresenta paralisia do sexto nervo ocular, com a menor velocidade média.

$$\text{Diferença} = \left(1 - \frac{\min(\text{velocidade média}_{\text{esquerdo}}, \text{velocidade média}_{\text{direito}})}{\max(\text{velocidade média}_{\text{esquerdo}}, \text{velocidade média}_{\text{direito}})} \right) \times 100 \quad (5.2)$$

Os resultados apresentados na Tabela 5, assim como em tabelas subsequentes,

foram avaliados com base em acurácia, precisão, sensibilidade, F1-Score, AUC-ROC e especificidade (POWERS, 2020; MARIANO, 2021; LI, 2024).

Os resultados na Tabela 5 mostram que a YOLOv8 superou o modelo de Costa et al. (2022) na métrica de sensibilidade e se mostrou equivalente na de especificidade. Além disso, alcançou bons resultados em acurácia, precisão, F1-Score e AUC. Dessa forma, corroborando o experimento anterior (Seção 5.2), que indica a precisão e a capacidade de rastreamento ocular do modelo YOLOv8, validando sua escolha para os experimentos posteriores.

Tabela 5 – Classificação baseada na velocidade média

Modelos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
Costa (COSTA et al., 2022)	92,64%	-	70%	-	-	95,78%
Yolov8	85,7%	91%	71,4%	80%	83,3%	95%
Yolov10	74,3%	69%	64,3%	66,7%	72,6%	81%
Yolov11	71,4%	62,5%	71,4%	66,7%	71,4%	71,4%

5.4 Experimento final de classificação

Antes de tudo, é importante salientar que este experimento emprega a validação cruzada ao longo de toda a metodologia proposta no conjunto de dados do sexto nervo. Durante o ajuste fino no treinamento da YOLOv8, os pacientes do conjunto de dados de paralisia do sexto nervo são divididos em cinco subconjuntos de tamanho igual. Essa divisão é então aplicada tanto no treinamento da YOLOv8 quanto no treinamento dos classificadores. Os modelos são então treinados cinco vezes, executando todas as etapas da nossa metodologia, cada vez utilizando quatro subconjuntos para treinamento e um para teste. O resultado final é a média dos cinco resultados. Essa abordagem avalia a capacidade de generalização do modelo, pois o modelo é avaliado em múltiplos subconjuntos de dados, em vez de depender de uma única divisão treino-teste.

Nesse experimento, utilizou-se a segunda versão da base de dados de paralisia, com 39 pacientes: 22 saudáveis e 17 paralíticos. Dessa forma, na etapa de treino da YOLOv8, os vídeos foram novamente convertidos em conjuntos de imagens a 30 quadros por segundo, totalizando 11.289 imagens. Já na etapa de classificação de séries temporais, após o janelamento, são adquiridas 1.872 séries temporais divididas em 1.498 para treino e 374 para teste. Antes do aumento de dados, o janelamento gerou 408 séries temporais de olhos com paralisia do sexto nervo, sendo 326 alocadas para treinamento e 82 para

teste. Mas após o aumento de dados, o conjunto de treinamento é composto por 2.802 séries temporais: 1.630 de casos de paralisia e 1.172 de casos saudáveis.

As imagens foram então utilizadas para treinar o modelo YOLOv8 com as mesmas configurações do experimento na Seção 5.2, pesos COCO por 100 épocas, congelamento das camadas durante as primeiras 50 épocas, que são descongeladas a partir da época 51, seguido de um ajuste fino por mais 50 épocas com todas as camadas descongeladas. A divisão é 20% das imagens do conjunto de dados do sexto nervo para ajustar o modelo e 80% para testes. Mantiveram-se as configurações padrão do modelo pré-treinado (ULTRALYTICS, 2024), o otimizador Adam, a taxa de aprendizado de 1×10^{-3} e o tamanho do lote de 16. A única exceção foi o tamanho da imagem de entrada, ajustado para 416×416 .

Os modelos treinados são, por sua vez, utilizados para rastrear os olhos de todos os pacientes da segunda versão da base de vídeos de pacientes com paralisia do sexto nervo, obtendo-se, assim, a trajetória ocular de 39 pacientes. Paralelamente, utiliza-se o modelo de detecção facial do MediaPipe (Seção 3.8) no mesmo conjunto de imagens para localizar a face do paciente e obter seu deslocamento. Dessa forma, combinando ambas as trajetórias, ocular e facial, é realizada a compensação dos movimentos da face na trajetória ocular, conforme explicitado na Seção 4.2.3. As trajetórias resultantes da compensação são, então, encaminhadas às próximas etapas da metodologia.

Em seguida, os testes foram divididos em dois segmentos principais, cada um com sua própria subdivisão. Antes desse experimento, o método de classificação havia sido aplicado de forma independente a cada série temporal, tratando-as como casos distintos. Porém, neste experimento, foi adotada uma nova estratégia: as métricas também são calculadas no nível do paciente, classificando-se primeiro a série temporal de cada sujeito e, em seguida, atribuindo-se uma classificação final ao paciente, desde que ao menos uma série temporal seja classificada como um caso de paralisia. O segundo segmento aborda o uso de séries temporais tanto univariadas quanto multivariadas.

Com os resultados por paciente das Tabelas 7 e 9, observa-se forte evidência de overfitting e de viés de classificação na métrica de sensibilidade, indicando que diversos modelos estão atribuindo praticamente todos os pacientes como positivos, ou seja, com paralisia. Isso corroborado com a métrica de especificidade obtida nesses mesmos modelos, que apresenta valores extremamente baixos, confirmando que, embora identifiquem quase

todos os pacientes positivos, raramente classificam corretamente os casos negativos, correspondentes a indivíduos saudáveis. Além disso, essa tendência é reforçada pelas métricas de acurácia e precisão, que permanecem próximas de 43%, sugerindo que os modelos não estão aprendendo padrões discriminativos relevantes, mas apenas reproduzindo a distribuição predominante das classes. Levando em consideração o banco de dados do sexto nervo, composto por 39 pacientes, dos quais 17 possuem a patologia e 22 são saudáveis, percebe-se que 17 correspondem aproximadamente a 43% do total, o que explica os valores obtidos para essas métricas. Além disso, os desvios-padrão altos observados em modelos multivariados sugerem instabilidade nos experimentos, indicando baixa capacidade de generalização.

Por outro lado, constata-se que os resultados das séries temporais das Tabelas 6 e 8 apresentam um equilíbrio geral mais elevado entre as métricas avaliadas. Por outro lado, nos experimentos por paciente os modelos avaliados apresentam maior estabilidade de identificação entre casos positivos e negativos, o que se traduz em valores de sensibilidade e especificidade mais equilibrados. Nota-se ainda que os modelos multivariados superaram os univariados na maioria das métricas, sugerindo que o uso simultâneo de múltiplas variáveis temporais fornece informações mais ricas e discriminativas para a tarefa de classificação. O melhor desempenho foi obtido pelo modelo K Neighbors TsLearn aplicado a séries multivariadas, atingindo 64% de acurácia, 57% de precisão, 70% de sensibilidade, 61% de F1-Score, 64% de AUC e 59% de especificidade. Embora esses resultados ainda não sejam competitivos em relação aos estudos da literatura, indicam um comportamento mais estável e menos enviesado do que o observado nos experimentos por paciente, afastando-se do padrão de overfitting previamente identificado. Além disso, os valores de AUC superiores a 60% sugerem que os modelos conseguem capturar parcialmente características relevantes das séries temporais, indicando potencial para aprimoramentos futuros por meio de estratégias como aumento do conjunto de dados, balanceamento das classes e otimização de hiperparâmetros.

Tabela 6 – Resultados por série temporal com séries univariadas.

Modelos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	41% ± 2%	23% ± 2%	76% ± 10%	36% ± 3%	55% ± 4%	32% ± 4%
ResNet	53% ± 7%	24% ± 5%	54% ± 17%	33% ± 7%	50% ± 9%	53% ± 12%
K Neighbors	63% ± 5%	30% ± 6%	52% ± 16%	37% ± 8%	59% ± 8%	66% ± 8%
K Neighbors TsLearn	62% ± 6%	30% ± 5%	54% ± 17%	37% ± 7%	58% ± 7%	64% ± 11%
Time Series SVC	61% ± 5%	31% ± 6%	63% ± 13%	41% ± 7%	66% ± 11%	61% ± 5%
Shapelet Learning Tslearn	32% ± 26%	15% ± 10%	66% ± 48%	24% ± 17%	44% ± 13%	23% ± 44%

Tabela 7 – Resultados por paciente com séries univariadas.

Modelos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	43% ± 1%	43% ± 1%	100% ± 1%	60% ± 1%	50% ± 1%	10% ± 1%
ResNet	45% ± 5%	44% ± 3%	95% ± 10%	60% ± 4%	50% ± 4%	8% ± 10%
K Neighbors	43% ± 1%	43% ± 1%	100% ± 1%	60% ± 1%	50% ± 1%	10% ± 1%
K Neighbors TsLearn	43% ± 1%	43% ± 1%	100% ± 1%	60% ± 1%	50% ± 1%	10% ± 1%
Time Series SVC	43% ± 12%	42% ± 7%	83% ± 15%	56% ± 8%	48% ± 12%	13% ± 18%
Shapelet Learning Tslearn	45% ± 5%	44% ± 3%	100% ± 1%	60% ± 2%	52% ± 4%	4% ± 8%

Tabela 8 – Resultados por série temporal com séries multivariadas.

Modelos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	43% ± 3%	43% ± 1%	89% ± 8%	58% ± 2%	49% ± 3%	9% ± 9%
ResNet	61% ± 16%	55% ± 15%	63% ± 17%	59% ± 16%	61% ± 16%	60% ± 17%
K Neighbors	63% ± 9%	57% ± 10%	69% ± 24%	61% ± 13%	64% ± 9%	59% ± 19%
K Neighbors TsLearn	64% ± 9%	57% ± 9%	70% ± 24%	61% ± 14%	64% ± 9%	59% ± 17%
Time Series SVC	58% ± 11%	52% ± 9%	73% ± 11%	60% ± 8%	60% ± 10%	47% ± 17%
Shapelet Learning Tslearn	47% ± 9%	45% ± 6%	85% ± 14%	58% ± 5%	51% ± 8%	17% ± 24%

Tabela 9 – Resultados por paciente com séries multivariadas.

Modelos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
RNN	46% ± 6%	45% ± 2%	100% ± 1%	61% ± 2%	52% ± 5%	5% ± 11%
ResNet	43% ± 20%	41% ± 14%	70% ± 30%	51% ± 19%	46% ± 21%	23% ± 23%
K Neighbors	41% ± 16%	37% ± 11%	60% ± 31%	45% ± 18%	43% ± 17%	26% ± 22%
K Neighbors TsLearn	41% ± 16%	37% ± 11%	60% ± 31%	45% ± 18%	43% ± 17%	26% ± 22%
Time Series SVC	46% ± 6%	35% ± 20%	62% ± 37%	44% ± 25%	48% ± 7%	35% ± 31%
Shapelet Learning Tslearn	44% ± 20%	38% ± 17%	53% ± 33%	44% ± 23%	45% ± 21%	37% ± 14%

A Tabela 10 mostra que o Fold 1 obteve o melhor resultado, seguido de perto pelo Fold 4, enquanto os Folds 2 e 3 obtiveram métricas abaixo do esperado, e o Fold 4 apresentou overfitting na métrica de Sensibilidade, embora não se encaixe no caso de overfitting presente nas Tabelas 7 e 9. Apesar destes resultados não se mostrarem proeminentes, eles ainda indicam potencial positivo para a pesquisa, em meio ao problema complexo de identificar a paralisia do sexto nervo utilizando séries temporais, que será desenvolvido na Seção 5.5.

Tabela 10 – Resultados por Fold do melhor modelo

Folds	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
Fold 1	73%	67%	75%	71%	73%	71%
Fold 2	56%	50%	64.5%	56.6%	57%	49%
Fold 3	55%	47%	33%	39%	52%	71%
Fold 4	61%	52%	100%	69%	66%	33%
Fold 5	72%	64%	76.3%	70%	72%	68%
Média	64% ± 9%	57% ± 9%	70% ± 24%	61% ± 14%	64% ± 9%	59% ± 17%

Dos resultados apresentados na Tabela 11, é possível verificar que o melhor resultado foi o do nosso artigo publicado, que alcançou Precisão e Especificidade superiores às do trabalho de Costa et al. (2022), o que reforça a relevância de seus resultados e a

capacidade do modelo de identificar pacientes saudáveis. Além de que, embora o resultado da média dos folds do melhor resultado obtido (Tabela 10) esteja abaixo do esperado, ele apresenta uma métrica de sensibilidade comparável à obtida por Costa et al. (2022). Além disso, a sensibilidade do nosso melhor fold demonstra uma melhora em relação à reportada por Costa et al. (2022).

Tabela 11 – Comparação de resultados da literatura.

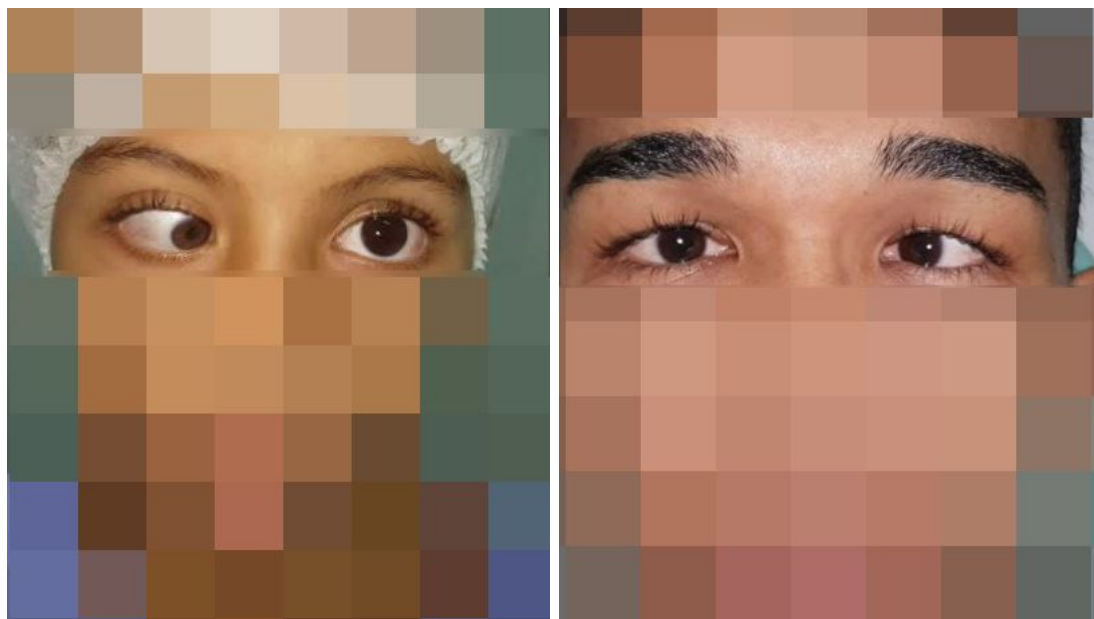
Trabalhos	Acurácia	Precisão	Sensibilidade	F1-Score	AUC	Especificidade
Costa et al. (2022)	92,6%	-	70%	-	-	95.7%
Melhor Fold Costa et al. (2022)	95,4%	-	66.7%	-	-	100%
Artigo publicado (Rodrigues Fernandes et al., 2025)	80%	95%	60%	74.5%	80%	97%
K Neighbors TsLearn Multivariado	64%	57%	70%	61%	64%	59%
K Neighbors TsLearn Multivariado (Melhor Fold)	73%	67%	75%	71%	73%	71%

Dessa forma, é pertinente salientar que os melhores modelos obtidos como resultados finais do nosso estudo, se provam relevantes para essa linha de estudo que ainda se mantém escassa e pouco desenvolvida. Levando em consideração os poucos trabalhos desenvolvidos na área, esta pesquisa se afirma não só como referência como também incentivo para trabalhos futuros da área, tanto na problemática de diagnóstico da paralisia do sexto nervo craniano quanto na metodologia de emprego de séries temporais na detecção de patologias oculares.

5.5 Estudo de Caso

Para evidenciar a complexidade do problema investigado nesta pesquisa, serão discutidos dois casos, um de acerto e outro de erro na detecção de paralisia do sexto nervo, sendo eles, respectivamente, o paciente 1 e o paciente 14 (Figura 25), ambos com paralisia. Será considerado o modelo K Neighbors TsLearn multivariado do melhor Fold, obtido com o melhor resultado da pesquisa (Tabela 10), detalhando o passo a passo das séries ao longo da metodologia proposta.

Figura 25 – Pacientes utilizados no estudo de caso.



(a) Paciente 1

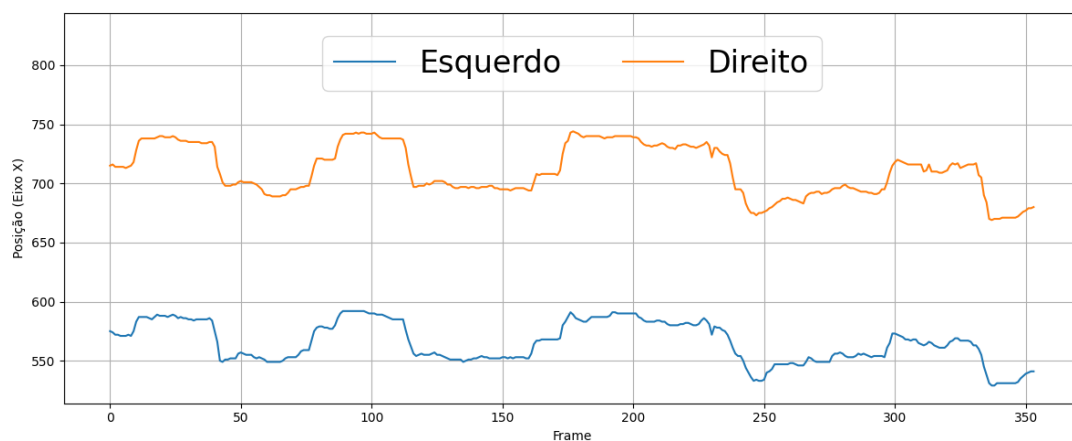
(b) Paciente 14

5.5.1 Caso de acerto: Paciente 1

Este paciente foi corretamente classificado como um caso positivo de paralisia, dessa forma, será apresentado o passo a passo pelo qual as séries temporais desse paciente foram submetidas.

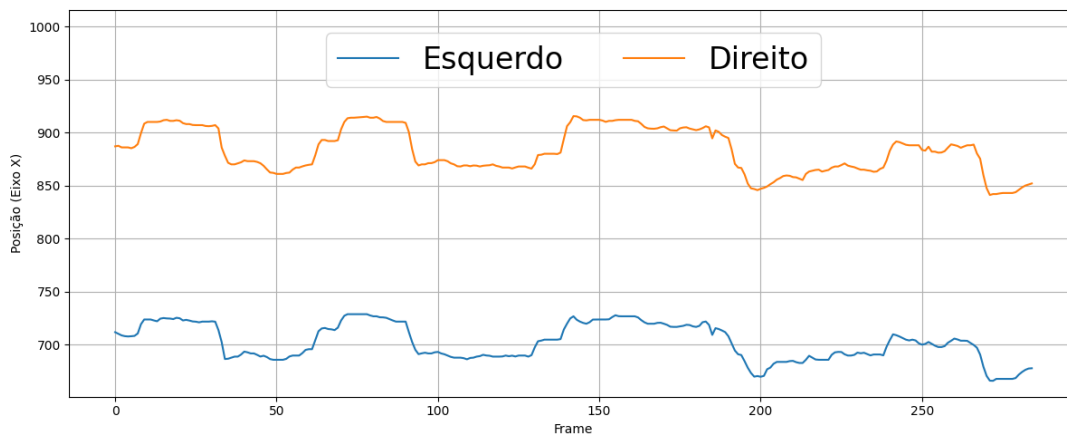
Na Figura 26 observa-se a série temporal da trajetória no eixo X dos olhos do paciente 1 ao longo dos frames do vídeo, conforme o processo de aquisição descrito na Seção 4.2.4. Além disso, o deslocamento ocular foi obtido por meio do modelo YOLOv8, tal qual descrito na Seção 5.4.

Figura 26 – Série temporal da trajetória dos olhos do paciente 1.



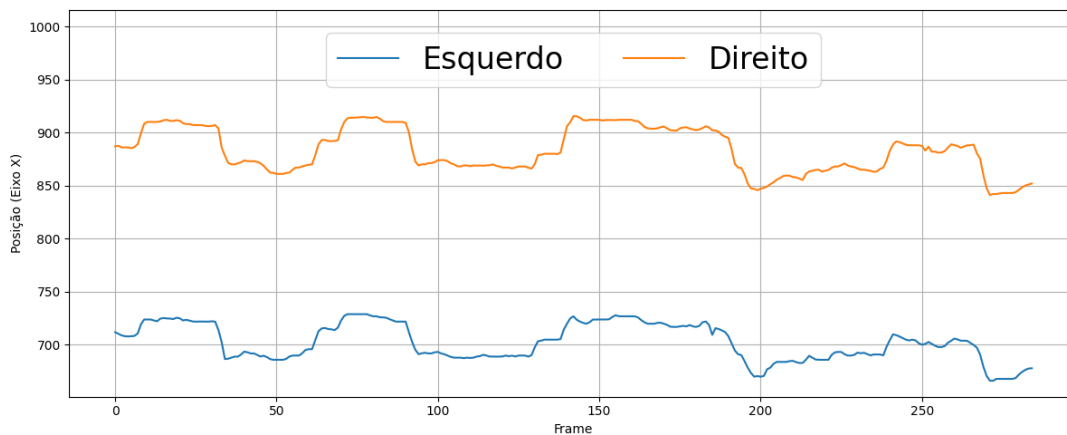
Além disso, na Figura 27, é perceptível que a série temporal após a interpolação não foi afetada por suas características originais, mesmo com a nova proporção de 285 frames, conforme descrito na Seção 4.2.5.1.

Figura 27 – Interpolação da série temporal do paciente 1.



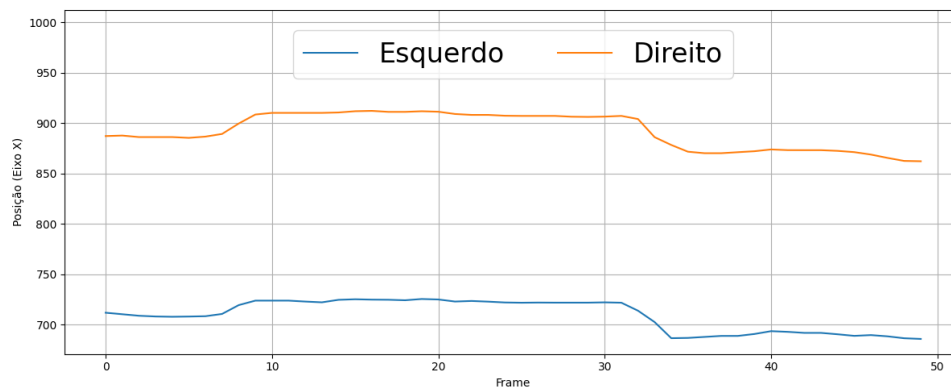
Em seguida, a Figura 28 apresenta a série temporal após a aplicação do filtro de Hampel, o qual remove ruído e valores discrepantes, como explicado na Seção 4.2.5.2.

Figura 28 – Filtro de Hampel aplicado.

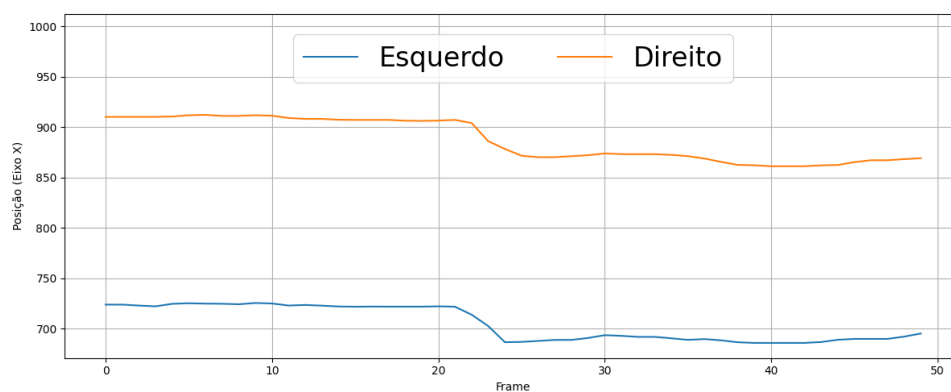


Já na Figura 29, observam-se as três primeiras janelas obtidas do paciente 1, seguindo o procedimento de janelamento por sobreposição da Seção 4.2.6, com o objetivo de gerar mais instâncias para o treinamento do modelo.

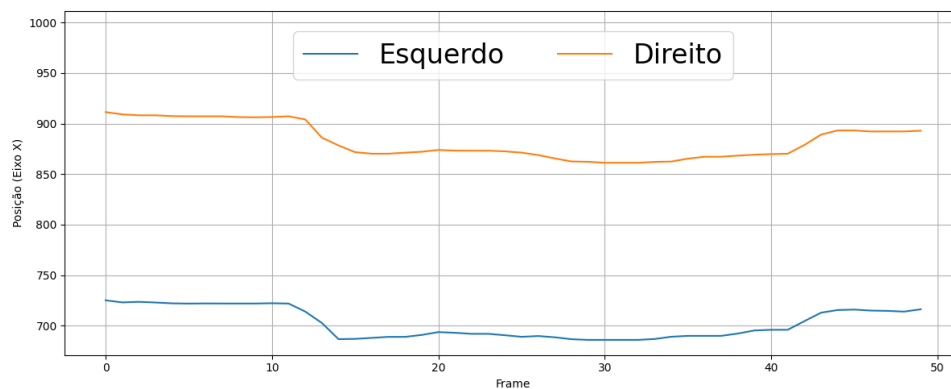
Figura 29 – Janelamento por sobreposição.



(a) Janela 1



(b) Janela 2



(c) Janela 3

Com base nas amostras apresentadas, constata-se a sutileza das diferenças comportamentais entre um olho com paralisia e um olho saudável, desde o processamento por interpolação até o filtro de Hampel (Figuras 27 e 28), até as janelas obtidas (Figura 29). A semelhança no deslocamento um do outro é perceptível. Dessa forma, o modelo deve ser capaz de identificar variações tênues na movimentação de cada olho, a fim de distinguir casos de paralisia de casos saudáveis. Isso é reforçado pelo aumento de dados

empregado apenas nos casos de paralisia, aplicado com o intuito de expandir o número de séries temporais no treinamento do modelo e realçar as diferenças em relação a séries de olhos saudáveis.

Entretanto, essa dificuldade não se restringe à similaridade entre as séries. Tal qual esclarecido na Seção 3.2, a paralisia do sexto nervo craniano compromete tão somente o movimento de abdução do olho, ou seja, o deslocamento horizontal para a região da têmpora (oposta ao nariz). Desse modo, os instantes em que se observa o comportamento característico de paralisia são escassos ao longo do vídeo, em que o paciente efetua movimentos semelhantes aos persecutórios (Seção 3.1.3) de forma consecutiva, conforme definido pelo protocolo de (COSTA et al., 2022). Logo, essa limitação se traduz, consequentemente, nos dados obtidos nas séries temporais, dificultando ainda mais a identificação consistente das nuances do olho paralítico.

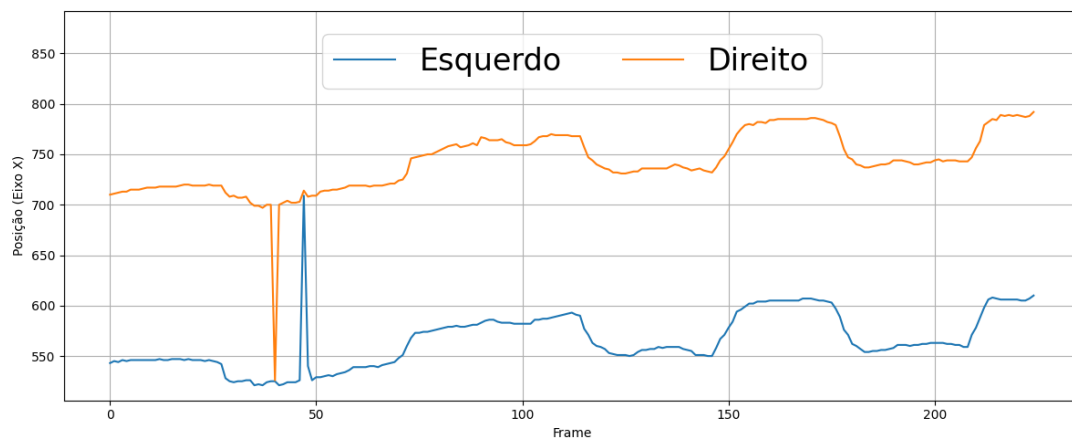
Adicionalmente, há uma limitação inerente ao processo de anotação dos dados, uma vez que cada paciente possui apenas um rótulo global, atribuído por um especialista, indicando a presença de paralisia. Dessa forma, após a implementação do janelamento por sobreposição, são geradas tanto janelas que não possuem o comportamento referente à paralisia como também janelas que contêm apenas uma parte de um movimento paralítico, ou seja, apenas um corte do mesmo. Apesar disso, como visto nas Tabelas 10 e 11, nossa pesquisa ainda assim conseguiu obter modelos capazes de identificar casos de paralisia do sexto nervo óptico.

5.5.2 Caso de falha: Paciente 14

Este é um caso de falha em que o paciente foi classificado como saudável, quando, na verdade, é portador da paralisia do sexto nervo. Será, portanto, discorrido sobre as etapas pelas quais as séries temporais provenientes passaram e discutida a possível causa da classificação equivocada.

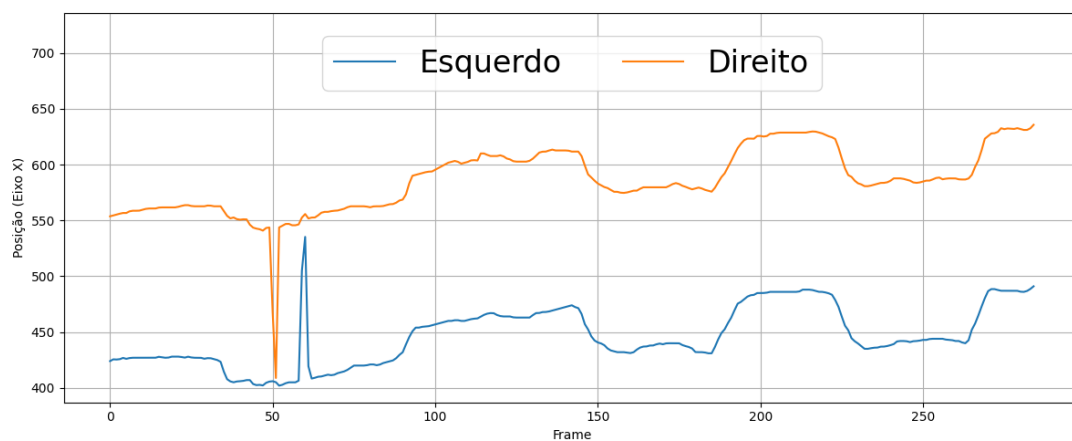
Na Figura 30 pode-se observar a trajetória dos olhos do paciente 14 no eixo X ao longo dos frames do vídeo. Na série temporal, é possível identificar um valor discrepante em torno do frame 50, que corresponde a um provável erro no rastreamento ocular.

Figura 30 – Série temporal da trajetória dos olhos do paciente 14.



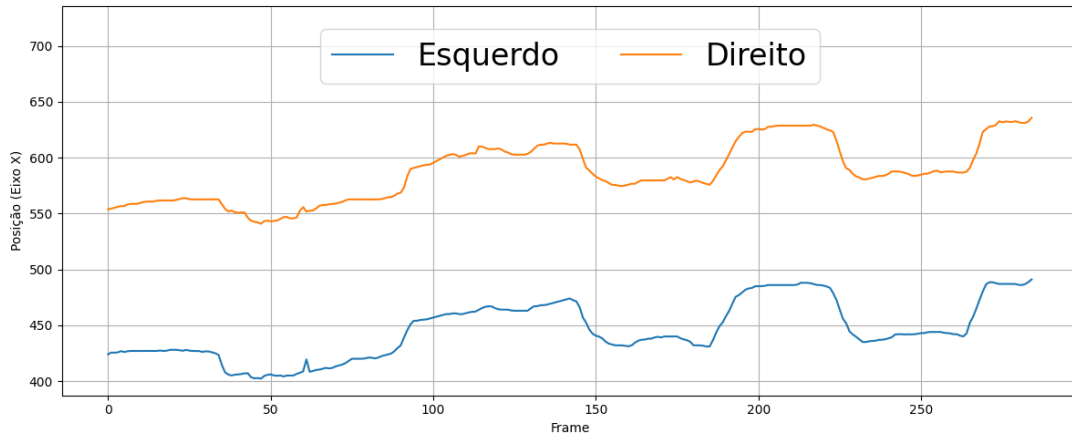
Em seguida, na Figura 31, apresenta-se a série temporal resultante da interpolação. Verifica-se que os valores discrepantes se mantêm, apesar de exibirem dimensões levemente menores.

Figura 31 – Interpolação da série temporal do paciente 14.



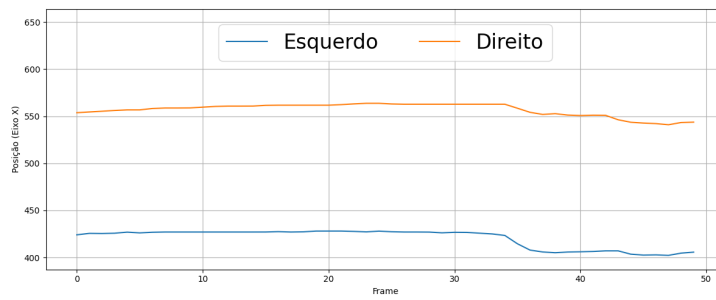
Após isso, a Figura 32 exibe a série temporal após a aplicação do filtro de Hampel. Comparando com a Figura 31, é perceptível a eficiência do filtro, que normalizou os valores discrepantes presentes na proximidade do frame 50.

Figura 32 – Filtro de Hampel.

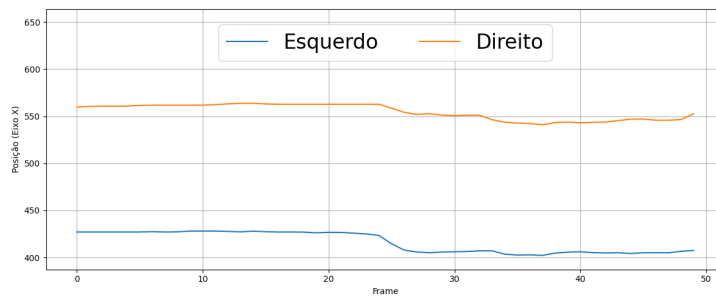


Já na Figura 33, observam-se as três primeiras janelas obtidas do paciente 1, seguindo o procedimento de janelamento por sobreposição, conforme a Seção 4.2.6.

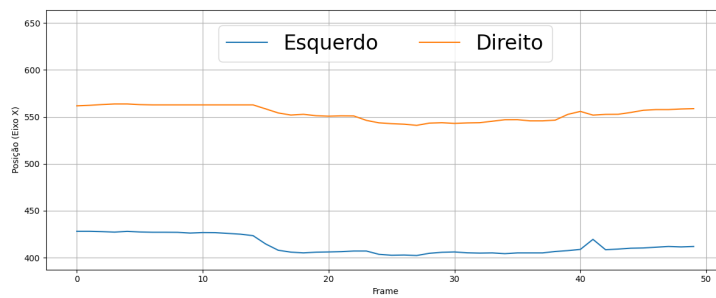
Figura 33 – Janelamento por sobreposição.



(a) Janela 1



(b) Janela 2

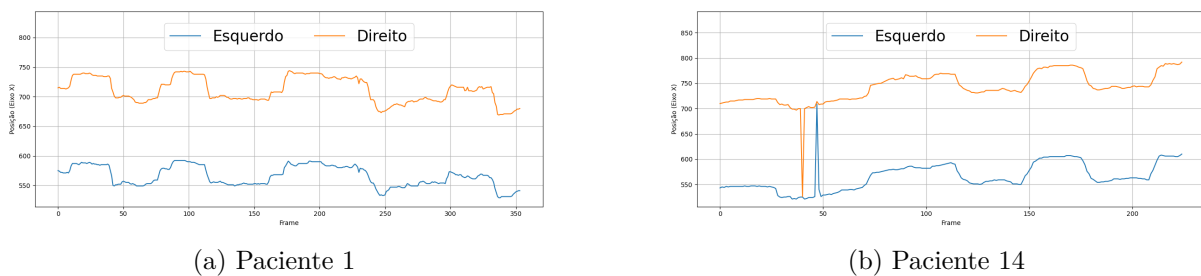


(c) Janela 3

Considerando as transformações observadas e comparando com o caso de acerto da seção anterior (5.5.1), é possível constatar que o paciente 14 apresenta uma trajetória mais suave do que o deslocamento dos olhos do paciente 1, como pode ser observado na Figura 34. Presumivelmente, isso pode levar o modelo a ter dificuldade em identificar o comportamento de paralisia, ainda mais considerando a problemática levantada no estudo de caso do paciente 1 (Seção 5.5.1).

Outra possível causa pode estar relacionada ao tamanho do vídeo e à etapa de interpolação, pois enquanto o vídeo do paciente 14 é menor, gerando uma série com 226 frames, em comparação, o paciente 1 forneceu uma série com 353 frames no total. Dessa forma, a interpolação, conforme descrito na Seção 4.2.5.1, realizará dois processos distintos para cada um desses pacientes. Na série temporal do paciente 1, a interpolação reduz a série para o tamanho padrão de 285 frames, já na série do paciente 14, ela é expandida para a nova quantidade. Portanto, é plausível a hipótese de que a interpolação linear aplicada não estimou adequadamente os valores da nova série do paciente 14, perdendo-se informação no processo e prejudicando a representação dos instantes que contêm a paralisia.

Figura 34 – Trajetórias oculares dos pacientes estudados.



6 CONCLUSÃO

A paralisia do sexto nervo óptico craniano é a patologia mais frequente dentre as paralisias motoras oculares, sendo responsável pelo movimento de abdução (Figura 1). Essa paralisia impede que o músculo reto lateral realize adequadamente esse movimento, causando dores de cabeça, enxaquecas, visão turva, tontura e visão dupla ao tentar mover o olho para o canto externo (ROSENBAUM; SANTIAGO, 1999; CURI; COSTA; BARROSO, 2013). Os métodos de diagnóstico dessa paralisia mais empregados são um exame invasivo, o teste de geração de força (TGF) e a eletro-oculografia (EOG), que é de alto custo.

Dessa forma, com o objetivo de proporcionar uma opção alternativa não invasiva, de baixo custo monetário e de fácil acesso e aplicação, este trabalho propôs um método de detecção de paralisia do sexto nervo óptico em vídeos de pacientes oftalmológicos, contribuindo para o diagnóstico dessa paralisia por especialistas.

Para isso, na primeira etapa da metodologia, é realizado o rastreamento do deslocamento ocular com a rede neural YOLOv8 (You Only Look Once) para obter a trajetória de cada olho do paciente, obtendo assim 0,91 de precisão e 0,94 de mAP50%. Seguido do rastreamento da face com o modelo de detecção facial do framework MediaPipe, que captura os movimentos da face nos vídeos, para que, então, seja realizada a compensação do deslocamento da face na trajetória dos olhos. Com essas trajetórias, são então geradas séries temporais multivariadas que, por sua vez, passam por duas etapas de pré-processamento: interpolação e filtro de Hampel. É então aplicada uma etapa de janelamento por sobreposição e, logo após, outra de aumento de dados e, finalmente, é empregado o modelo K Neighbors TsLearn para classificar a presença de paralisia com base nas séries temporais resultantes, alcançando média de 70% na sensibilidade, com desvio padrão de 24%, e 75% de sensibilidade no melhor fold.

O modelo obtido pelo método final do trabalho mostrou-se promissor, alcançando uma métrica de sensibilidade à altura do melhor resultado da literatura, além de seu melhor fold apresentar métricas em geral mais elevadas, ultrapassando a marca anterior de 70% em sensibilidade e atingindo 75%, respondendo assim à primeira questão de pesquisa, de que de fato é possível desenvolver uma metodologia que não depende da definição prévia de um limiar. Além disso, esses resultados revelam uma possibilidade de melhoria

na detecção de casos positivos de paralisia e realçam a necessidade de maior refinamento na classificação de pacientes saudáveis, evidenciada pela baixa especificidade. Esse fato é reforçado pelo resultado do melhor modelo apresentado em nosso artigo publicado (Rodrigues Fernandes et al., 2025), que apresentou desempenho promissor, comparável ao estado da arte, esclarecendo, assim, a segunda questão da pesquisa, que foi atingida parcialmente, pois o melhor resultado obtido não se mostra superior ao apresentado na literatura (COSTA et al., 2022), sendo apenas comparável a ele.

Portanto, os resultados deste trabalho mostram-se relevantes para essa linha de estudo, ainda escassa e pouco desenvolvida. Esta pesquisa se sustenta como referência e incentivo para trabalhos futuros da área, proporcionando experimentos e perspectivas no emprego de séries temporais na detecção de patologias oculares e no diagnóstico da paralisia do sexto nervo óptico craniano por meio de modelos de aprendizado profundo. Dentre as contribuições deste trabalho, incluem-se:

- Desenvolvimento de uma nova metodologia computacional de baixo custo, fácil acesso e aplicação para o auxílio de especialistas no diagnóstico da paralisia do sexto nervo óptico usando séries temporais por meio do uso de modelos de aprendizado profundo;
- Elaboração de um método de detecção da paralisia do sexto nervo óptico craniano que independe de um limiar ou outro parâmetro pré-definido;
- A expansão da base de vídeos de pacientes que apresentam a paralisia do sexto nervo óptico;

Em relação às limitações da pesquisa, destaca-se a necessidade de um refinamento da metodologia, com o objetivo de obter resultados melhores, especialmente nas métricas de sensibilidade e especificidade. Outra limitação é o baixo número de pacientes disponíveis, o que cria a necessidade de etapas de expansão de instâncias de dados para um treinamento mais adequado dos modelos utilizados. Há também poucos trabalhos na literatura que abordam a problemática do diagnóstico da paralisia do sexto nervo craniano, o que dificulta a comparação do trabalho desenvolvido com uma ampla variedade de pesquisas e resultados.

6.1 Trabalhos Futuros

Como trabalhos futuros, sugere-se investigar possíveis formas de anotar ou identificar os instantes particulares de presença da paralisia em séries temporais de olhos paralíticos, com o objetivo de isolar os trechos que realmente caracterizam a paralisia do sexto nervo e, então, disponibilizar aos modelos de aprendizado profundo séries temporais que melhor representem os casos positivos dessa patologia. Além disso, outra possibilidade é a expansão do método proposto para o diagnóstico da paralisia de outros nervos ópticos, como, por exemplo, o terceiro (SAMADANI et al., 2015).

6.2 Produções Científicas

Como consequência desta pesquisa, foi publicado um artigo científico com resultados preliminares, conforme mostra a Tabela 12.

Tabela 12 – Artigo publicado relacionado à metodologia proposta.

Anais	Título
CENTERIS 2024	Identifying Sixth Cranial Nerve Palsy using Time Series Classifiers - Artigo Completo

REFERÊNCIAS

- ABDI, H.; VALENTIN, D.; EDELMAN, B. **Neural networks**. [S.l.]: Sage, 1999.
- ADHIKARI, R.; AGRAWAL, R. K. An introductory study on time series modeling and forecasting. **arXiv preprint arXiv:1302.6613**, 2013.
- ALIF, M. A. R.; HUSSAIN, M. Yolov1 to yolov10: A comprehensive review of yolo variants and their application in the agricultural domain. **arXiv preprint arXiv:2406.10139**, 2024.
- ALMEIDA, J. D. Sousa de; 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, p. 462–473, 2015.
- ANGSANA, N. C.; INDRIASARI, A. D.; MAHAYANA, I. T. Real time video-based gaze tracking for detecting subtle deviation angle change in abducens nerve palsy. **International Journal of Ophthalmology**, v. 18, n. 9, p. 1713, 2025.
- 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, p. 12987–13004, 2019.
- BATTY, C.; PERKINS, C.; SITA, J. How we came to eye tracking animation: A cross-disciplinary approach to researching the moving image. **Refractory: a journal of entertainment media**, v. 25, p. 1 – 12, 2015. ISSN 1447-4905.
- BAZAREVSKY, V.; KARTYNNIK, Y.; VAKUNOV, A.; RAVEENDRAN, K.; GRUNDMANN, M. Blazeface: Sub-millisecond neural face detection on mobile gpus. **arXiv preprint arXiv:1907.05047**, 2019.
- BEVERIDGE, S. Least squares estimation of missing values in time series. **Communications in Statistics-Theory and Methods**, Taylor & Francis, v. 21, n. 12, p. 3479–3496, 1992.
- BEZDAN, T. Convolutional neural network layers and architectures. In: **Proceedings of the International Scientific Conference-Sinteza 2019**. [S.l.: s.n.], 2019.
- BICAS, H. E. Oculomotricidade e seus fundamentos. **Arquivos Brasileiros de Oftalmologia**, SciELO Brasil, v. 66, n. 5, p. 687–700, 2003.
- BOCHKOVSKIY, A.; WANG, C.-Y.; LIAO, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection. **arXiv preprint arXiv:2004.10934**, 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.
- CHEN, Y.; CHEN, Q.; HU, Q.; CHENG, J. Dual assignment of labels for end-to-end fully convolutional object detection. **Pattern Recognition**, Elsevier, v. 169, p. 111877, 2026.

CHEN, Z.-Y.; FAN, Z.-P.; SUN, M. A hierarchical multiple kernel support vector machine for customer churn prediction using longitudinal behavioral data. **European Journal of operational research**, Elsevier, v. 223, n. 2, p. 461–472, 2012.

CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 1251–1258.

CHOMBOON, K.; CHUJAI, P.; TEERARASSAMEE, P.; KERDPRASOP, K.; KERDPRASOP, N. An empirical study of distance metrics for k-nearest neighbor algorithm. In: **Proceedings of the 3rd international conference on industrial application engineering**. [S.l.: s.n.], 2015. v. 2, p. 4.

CHOWDHURY, G. G. **Introduction to modern information retrieval**. [S.l.]: Facet publishing, 2010.

CLARO, M.; VERAS, R.; SANTANA, A.; ARAUJO, F.; SILVA, R.; ALMEIDA, J.; LEITE, D. An hybrid feature space from texture information and transfer learning for glaucoma classification. **Journal of Visual Communication and Image Representation**, Elsevier, v. 64, p. 102597, 2019.

COSTA, P. B. da; ALMEIDA, J. D. S. de; TEIXEIRA, J. A. M.; JUNIOR, G. B.; PAIVA, A. C. de; SILVA, A. C. Computational method for aid in the diagnosis of sixth optic nerve palsy through digital videos. **Computers in Biology and Medicine**, Elsevier, v. 150, p. 106098, 2022.

COVER, T.; HART, P. Nearest neighbor pattern classification. **IEEE transactions on information theory**, IEEE, v. 13, n. 1, p. 21–27, 1967.

CUNNINGHAM, P.; DELANY, S. J. K-nearest neighbour classifiers: (with python examples). **arXiv preprint arXiv:2004.04523**, 2020.

CUNNINGHAM, P.; DELANY, S. J. K-nearest neighbour classifiers-a tutorial. **ACM computing surveys (CSUR)**, ACM New York, NY, USA, v. 54, n. 6, p. 1–25, 2021.

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.

CURI RENATO LUIZ NAHOUM, C. I. C. B. d. O. B. T. G. M. Vi nerve palsy (abducens palsy). 2013.

CUTURI, M. Fast global alignment kernels. In: **Proceedings of the 28th international conference on machine learning (ICML-11)**. [S.l.: s.n.], 2011. p. 929–936.

CUTURI, M.; BLONDEL, M. Soft-dtw: a differentiable loss function for time-series. In: PMLR. **International conference on machine learning**. [S.l.], 2017. p. 894–903.

DASARATHY, B. V. Nearest neighbor (nn) norms: Nn pattern classification techniques. **IEEE Computer Society Tutorial**, 1991.

DU, K.-L.; SWAMY, M. N. **Neural networks and statistical learning**. [S.l.]: Springer Science & Business Media, 2013.

DU, K.-L.; ZHANG, R.; JIANG, B.; ZENG, J.; LU, J. Understanding machine learning principles: Learning, inference, generalization, and computational learning theory. **Mathematics**, MDPI, v. 13, n. 3, p. 451, 2025.

DUCHOWSKI, A. **Eye Tracking Methodology: Theory and Practice**. Springer International Publishing, 2017. ISBN 9783319578811. Disponível em: <<https://books.google.com.br/books?id=0Ge2tAEACAAJ>>.

ELIWA, E.; EL-HAFEEZ, T. A. Advancing crop health with yolov11 classification of plant diseases. **Neural Computing and Applications**, v. 37, p. 15223–15253, 05 2025.

ESLING, P.; AGON, C. Time-series data mining. **ACM Computing Surveys (CSUR)**, ACM New York, NY, USA, v. 45, n. 1, p. 1–34, 2012.

EVERINGHAM, M.; ESLAMI, S. A.; GOOL, L. V.; WILLIAMS, C. K.; WINN, J.; ZISSERMAN, A. The pascal visual object classes challenge: A retrospective. **International journal of computer vision**, Springer, v. 111, n. 1, p. 98–136, 2015.

GLAVAN, A. F.; CROITORU, V. Cloud environment assessment using clustering techniques on microservices dataset. In: IEEE. **2022 14th International Conference on Communications (COMM)**. [S.l.], 2022. p. 1–6.

GNAUCK, A. Interpolation and approximation of water quality time series and process identification. **Analytical and bioanalytical chemistry**, Springer, v. 380, n. 3, p. 484–492, 2004.

GOOGLE. **MediaPipe - Guia de detecção facial**. 2023. <https://ai.google.dev/edge/mediapipe/solutions/vision/face_detector>. "Acesso em 13/03/2025".

GRABOCKA, J.; SCHILLING, N.; WISTUBA, M.; SCHMIDT-THIEME, L. Learning time-series shapelets. In: **Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining**. [S.l.: s.n.], 2014. p. 392–401.

HALDER, R.; CHATTERJEE, R.; SANYAL, D.; MALLICK, P. K. Deep learning based smart attendance monitoring system. 09 2019.

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.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 770–778.

HENDERSON, P.; FERRARI, V. End-to-end training of object class detectors for mean average precision. In: SPRINGER. **Asian conference on computer vision**. [S.l.], 2016. p. 198–213.

HIPEL, K. W.; MCLEOD, A. I. **Time series modelling of water resources and environmental systems**. [S.l.]: Elsevier, 1994. v. 45.

HOWARD, A. G.; ZHU, M.; CHEN, B.; KALENICHENKO, D.; WANG, W.; WEYAND, T.; ANDREETTO, M.; ADAM, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. **arXiv preprint arXiv:1704.04861**, 2017.

HUSSAIN, M. Yolov5, yolov8 and yolov10: The go-to detectors for real-time vision. **arXiv preprint arXiv:2407.02988**, 2024.

JAKHETE, S. A.; KULKARNI, N. A comprehensive survey and evaluation of mediapipe face mesh for human emotion recognition. In: **IEEE. 2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA)**. [S.l.], 2024. p. 1–8.

JYOTSNA, C.; AMUDHA, J.; RAM, A.; FRUET, D.; NOLLO, G. Predicteye: Personalized time series model for mental state prediction using eye tracking. **IEEE Access**, IEEE, v. 11, p. 128383–128409, 2023.

KARHUNEN, J.; RAIKO, T.; CHO, K. Unsupervised deep learning: A short review. **Advances in independent component analysis and learning machines**, Elsevier, p. 125–142, 2015.

KATARIA, A.; SINGH, M. A review of data classification using k-nearest neighbour algorithm. **International Journal of Emerging Technology and Advanced Engineering**, v. 3, n. 6, p. 354–360, 2013.

KEOGH, E.; RATANAMAHATANA, C. A. Exact indexing of dynamic time warping. **Knowledge and information systems**, Springer, v. 7, n. 3, p. 358–386, 2005.

KHANAM, R.; HUSSAIN, M. Yolov11: An overview of the key architectural enhancements. **arXiv preprint arXiv:2410.17725**, 2024.

KING, D. E. Dlib-ml: A machine learning toolkit. **The Journal of Machine Learning Research**, JMLR. org, v. 10, p. 1755–1758, 2009.

KOTTHAPALLI, M.; RAVIPATI, D.; BHATIA, R. Yolov1 to yolov11: A comprehensive survey of real-time object detection innovations and challenges. **arXiv preprint arXiv:2508.02067**, 2025.

KUROMIYA, H. Automatic classification of the learning pattern-time-series clustering of students' reading behaviors. 2021.

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 UK London, v. 521, n. 7553, p. 436–444, 2015.

LEPOT, M.; AUBIN, J.-B.; CLEMENS, F. H. Interpolation in time series: An introductory overview of existing methods, their performance criteria and uncertainty assessment. **Water**, MDPI, v. 9, n. 10, p. 796, 2017.

LI, J. Area under the roc curve has the most consistent evaluation for binary classification. **PLOS ONE**, v. 19, 12 2024.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft coco: Common objects in context. In: **SPRINGER. Computer Vision–ECCV 2014: 13th European Conference, Zurich**,

Switzerland, September 6-12, 2014, Proceedings, Part V 13. [S.l.], 2014. p. 740–755.

LIU, W.; ANGUELOV, D.; ERHAN, D.; SZEGEDY, C.; REED, S.; FU, C.-Y.; BERG, A. C. Ssd: Single shot multibox detector. In: SPRINGER. **European conference on computer vision**. [S.l.], 2016. p. 21–37.

LIU, W.; HASAN, I.; LIAO, S. Center and scale prediction: Anchor-free approach for pedestrian and face detection. **Pattern Recognition**, Elsevier, v. 135, p. 109071, 2023.

LIVERSEDGE, S.; GILCHRIST, I.; EVERLING, S. **The Oxford handbook of eye movements**. [S.l.]: OUP Oxford, 2011.

LÖNING, M.; BAGNALL, A.; GANESH, S.; KAZAKOV, V.; LINES, J.; KIRÁLY, F. J. sktime: A unified interface for machine learning with time series. **arXiv preprint arXiv:1909.07872**, 2019.

LUGARESI, C.; TANG, J.; NASH, H.; MCCLANAHAN, C.; UBOWEJA, E.; HAYS, M.; ZHANG, F.; CHANG, C.-L.; YONG, M.; LEE, J. et al. Mediapipe: A framework for perceiving and processing reality. In: **Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR)**. [S.l.: s.n.], 2019. v. 2019.

LUGARESI, C.; TANG, J.; NASH, H.; MCCLANAHAN, C.; UBOWEJA, E.; HAYS, M.; ZHANG, F.; CHANG, C.-L.; YONG, M.; LEE, J. et al. Mediapipe: A framework for perceiving and processing reality. In: LONG BEACH, CA. **Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR)**. [S.l.], 2019. v. 2019, p. 2.

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 às desordens temporomandibulares. **Revista Odontológica do Brasil Central**, v. 19, n. 51, 2010.

MARIANO, D. Métricas de avaliação em machine learning: acurácia, sensibilidade, precisão, especificidade e f-score. In: _____. [S.l.: s.n.], 2021. ISBN 9786599275326.

MARTINSEN, O. G.; HEISKANEN, A. Bioimpedance and bioelectricity basics. Elsevier, 2023.

MCCULLAGH, P. Face detection by using haar cascade classifier. **Wasit journal of computer and mathematics science**, v. 2, n. 1, p. 1–5, 2023.

METZ, H. S. Forced duction, active force generation, and saccadic velocity tests. **International ophthalmology clinics**, v. 16, n. 3, p. 47–73, 1976.

MORADI, M.; SAMWALD, M. Deep learning, natural language processing, and explainable artificial intelligence in the biomedical domain. 02 2022.

MUKHIDDINOV, M.; DJURAEV, O.; AKHMEDOV, F.; MUKHAMADIYEV, A.; CHO, J. Masked face emotion recognition based on facial landmarks and deep learning approaches for visually impaired people. **Sensors**, MDPI, v. 23, n. 3, p. 1080, 2023.

NEMER, M. A.; AZAR, J.; MAKHOUL, A.; BOURGEOIS, J. A review of early time series classification methods on machinery dataset. In: IEEE. **2024 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics**. [S.l.], 2024. p. 347–352.

OH, C.; HAN, S.; JEONG, J. Time-series data augmentation based on interpolation. **Procedia Computer Science**, Elsevier, v. 175, p. 64–71, 2020.

ORGANIZATION, W. H. **Blindness and vision impairment**. 2023. <<https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment>>. Accessed: 2025-04-22.

ORGE MD; K. DAVID EPLEY, M. F. H. **Complex Strabismus Simulator**. 2019. Disponível em: <<https://www.aao.org/education/interactive-tool/complex-strabismus-simulator>>.

O'SHEA, K.; NASH, R. An introduction to convolutional neural networks. **arXiv preprint arXiv:1511.08458**, 2015.

PEARSON, R. K.; NEUVO, Y.; ASTOLA, J.; GABBOUJ, M. Generalized hampel filters. **EURASIP Journal on Advances in Signal Processing**, Springer, v. 2016, p. 1–18, 2016.

PETITJEAN, F.; KETTERLIN, A.; GANÇARSKI, P. A global averaging method for dynamic time warping, with applications to clustering. **Pattern recognition**, Elsevier, v. 44, n. 3, p. 678–693, 2011.

PIRES, L. A. G. A constituição dirigente na era da quarta revolução industrial: Promessas e desafios. In: **Congresso Internacional de Direitos Humanos de Coimbra**. [S.l.: s.n.], 2025. v. 10, n. 1.

POPESCU, M.-C.; BALAS, V. E.; PERESCU-POPESCU, L.; MASTORAKIS, N. Multilayer perceptron and neural networks. **WSEAS transactions on circuits and systems**, v. 8, n. 7, p. 579–588, 2009.

POWERS, D. M. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. **arXiv preprint arXiv:2010.16061**, 2020.

PRASAD, S.; VOLPE, N. J. Paralytic strabismus: third, fourth, and sixth nerve palsy. **Neurologic clinics**, Elsevier, v. 28, n. 3, p. 803–833, 2010.

RAKHMATULIN, I.; DUCHOWSKI, A. T. Deep neural networks for low-cost eye tracking. **Procedia Computer Science**, Elsevier, v. 176, p. 685–694, 2020.

RANGEKING; JOCHER, G. **Brief summary of YOLOv8 model structure**. 2023. Disponível em: <<https://github.com/ultralytics/ultralytics/issues/189>>.

REIS, D.; KUPEC, J.; HONG, J.; DAOUDI, A. Real-time flying object detection with yolov8. **arXiv preprint arXiv:2305.09972**, 2023.

Rodrigues Fernandes, S. E.; da Costa, P. B.; de Almeida, J. D. S.; BRAZ, G.; PAIVA, A. Identifying sixth cranial nerve palsy using time series classifiers. **Procedia Computer Science**, v. 256, p. 1232–1240, 2025. ISSN 1877-0509. CENTERIS - International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1877050925005940>>.

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.

RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATY, A.; KHOSLA, A.; BERNSTEIN, M. et al. Imagenet large scale visual recognition challenge. **International journal of computer vision**, Springer, v. 115, n. 3, p. 211–252, 2015.

SADIKU, M.; AJAYI, A.; SADIKU, J. Machine learning: An overview. **International Journal of Advanced Scientific Research**, v. 9, p. 1121–1131, 10 2025.

SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. **IEEE transactions on acoustics, speech, and signal processing**, IEEE, v. 26, n. 1, p. 43–49, 2003.

SALEHINEJAD, H.; SANKAR, S.; BARFETT, J.; COLAK, E.; VALAEE, S. Recent advances in recurrent neural networks. **arXiv preprint arXiv:1801.01078**, 2017.

SAMADANI, U.; FAROOQ, S.; RITLOP, R.; WARREN, F.; REYES, M.; LAMM, E.; ALEX, A.; NEHRBASS, E.; KOLECKI, R.; JURELLER, M. et al. Detection of third and sixth cranial nerve palsies with a novel method for eye tracking while watching a short film clip. **Journal of neurosurgery**, American Association of Neurological Surgeons, v. 122, n. 3, p. 707–720, 2015.

SANDLER, M.; HOWARD, A.; ZHU, M.; ZHMOGINOV, A.; CHEN, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2018. p. 4510–4520.

SARKER, I. H. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. **SN computer science**, Springer, v. 2, n. 6, p. 1–20, 2021.

SCHLEGEL, S.; KORN, N.; SCHEUERMANN, G. On the interpolation of data with normally distributed uncertainty for visualization. **IEEE transactions on visualization and computer graphics**, IEEE, v. 18, n. 12, p. 2305–2314, 2012.

SCHMIDHUBER, J. Deep learning in neural networks: An overview. **Neural networks**, Elsevier, v. 61, p. 85–117, 2015.

SCHMIDT, R. M. Recurrent neural networks (rnns): A gentle introduction and overview. **arXiv preprint arXiv:1912.05911**, 2019.

SHINDE, P. P.; SHAH, S. A review of machine learning and deep learning applications. In: IEEE. **2018 Fourth international conference on computing communication control and automation (ICCCUBEA)**. [S.l.], 2018. p. 1–6.

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. **Image Analysis and Recognition: 15th International Conference, ICIAR 2018, Póvoa de Varzim, Portugal, June 27–29, 2018, Proceedings 15**. [S.l.], 2018. p. 679–686.

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

SOHAN, M.; RAM, T. S.; REDDY, C. V. R. A review on yolov8 and its advancements. In: SPRINGER. **International conference on data intelligence and cognitive informatics**. [S.l.], 2024. p. 529–545.

SOUSA, J. A.; PAIVA, A.; SILVA, A.; ALMEIDA, J. D.; JUNIOR, G. B.; DINIZ, J. O.; FIGUEREDO, W. K.; GATTASS, M. Automatic segmentation of retinal layers in oct images with intermediate age-related macular degeneration using u-net and dexined. **PLoS one**, Public Library of Science San Francisco, CA USA, v. 16, n. 5, p. e0251591, 2021.

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

TAO, Z.; XU, Q.; LIU, X.; LIU, J. An integrated approach implementing sliding window and dtw distance for time series forecasting tasks. **Applied Intelligence**, Springer, v. 53, n. 17, p. 20614–20625, 2023.

TAVENARD, R. **An introduction to Dynamic Time Warping**. 2021. <<https://rtavenar.github.io/blog/dtw.html>>.

TAVENARD, R.; FAOUZI, J.; VANDEWIELE, G.; DIVO, F.; ANDROZ, G.; HOLTZ, C.; PAYNE, M.; YURCHAK, R.; RUSSWURM, M.; KOLAR, K. et al. **Tslearn K neighbors**. 2018. Disponível em: <<https://github.com/tslearn-team/tslearn/blob/main/tslearn/neighbors/neighbors.py>>.

TAVENARD, R.; FAOUZI, J.; VANDEWIELE, G.; DIVO, F.; ANDROZ, G.; HOLTZ, C.; PAYNE, M.; YURCHAK, R.; RUSSWURM, M.; KOLAR, K. et al. Tslearn, a machine learning toolkit for time series data. **Journal of machine learning research**, v. 21, n. 118, p. 1–6, 2020.

THAMAN, B.; CAO, T.; CAPORUSSO, N. Face mask detection using mediapipe facemesh. In: IEEE. **2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)**. [S.l.], 2022. p. 378–382.

THAMCHAROEN, T.; SRISERTPOL, J.; CHOMMUANGPUCK, P.; DEEYING, J. Fault detection and classification for slider attachment process using convolution neural network. **International Journal of Neural Networks and Advanced Applications**, v. 7, p. 60–65, 12 2020.

ULTRALYTICS. **Ultralytics YOLO11**. 2024. Disponível em: <<https://docs.ultralytics.com/models/yolo11/>>.

- UMBELINO, C. C.; ÁVILA, M. P. d. **As condições de saúde ocular no Brasil 2023**. [S.l.]: São Paulo: Conselho Brasileiro de Oftalmologia, 2023.
- 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.
- WANG, A.; CHEN, H.; LIU, L.; CHEN, K.; LIN, Z.; HAN, J.; DING, G. Yolov10: Real-time end-to-end object detection. **arXiv preprint arXiv:2405.14458**, 2024.
- WANG, B. A parallel implementation of computing mean average precision. **arXiv preprint arXiv:2206.09504**, 2022.
- WEN, Q.; SUN, L.; YANG, F.; SONG, X.; GAO, J.; WANG, X.; XU, H. Time series data augmentation for deep learning: A survey. **arXiv preprint arXiv:2002.12478**, 2020.
- WU, Y.-c.; FENG, J.-w. Development and application of artificial neural network. **Wireless Personal Communications**, Springer, v. 102, n. 2, p. 1645–1656, 2018.
- XIN, Y.; KONG, L.; LIU, Z.; CHEN, Y.; LI, Y.; ZHU, H.; GAO, M.; HOU, H.; WANG, C. Machine learning and deep learning methods for cybersecurity. **Ieee access**, IEEE, v. 6, p. 35365–35381, 2018.
- YAMASHITA, R.; NISHIO, M.; DO, R. K. G.; TOGASHI, K. Convolutional neural networks: an overview and application in radiology. **Insights into imaging**, Springer, v. 9, n. 4, p. 611–629, 2018.
- YANG, Y.; DENG, Q.; SHEN, F.; ZHAO, J.; LUO, C. A shapelet learning method for time series classification. In: IEEE. **2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)**. [S.l.], 2016. p. 423–430.
- ZHOU, Z.; SHI, S.; TANG, X.; XU, Z.; YE, J.; HUANG, X.; LOU, L. A deep learning-based image analysis for assessing the extent of abduction in abducens nerve palsy patients before and after strabismus surgery. **Advances in Ophthalmology Practice and Research**, Elsevier, v. 4, n. 4, p. 202–208, 2024.
- 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.
- ZUO, Q.; CHEN, S.; WANG, Z. R2au-net: attention recurrent residual convolutional neural network for multimodal medical image segmentation. **Security and Communication Networks**, Wiley Online Library, v. 2021, n. 1, p. 6625688, 2021.