UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO DE JOINVILLE
CURSO DE ENGENHARIA MECATRÔNICA

RENAN SEVILHA SAKAMOTO

COMPARAÇÃO DE MÉTODOS DE DETECÇÃO FACIAL SOB INFLUÊNCIA DA VARIAÇÃO DA ILUMINAÇÃO

Joinville
2024

RENAN SEVILHA SAKAMOTO

COMPARAÇÃO DE MÉTODOS DE DETECÇÃO FACIAL SOB INFLUÊNCIA DA VARIAÇÃO DA ILUMINAÇÃO

Trabalho apresentado como requisito parcial para obtenção do título de bacharel em Engenharia Mecatrônica, no Curso de Engenharia Mecatrônica, do Centro Tecnológico de Joinville, da Universidade Federal de Santa Catarina.

Orientador: Dr. Pablo Andretta Jaskowiak

Coorientador: Dr. Benjamin Grando Moreira

Joinville
2024

Dedico este trabalho a meus colegas queridos e meus queridos pais e meu irmão.

AGRADECIMENTOS

Agradeço aos meus pais, Maria de Fátima Sevilha Sakamoto e Jairo Yukio Sakamoto, e meu irmão Fernando, por todo o apoio e carinho durante todo esse período na faculdade.

Agradeço aos meus amigos de antes de faculdade e aos grandes amigos que fiz nessa jornada, que me acompanharam nesse desafio, certamente levarei comigo lembranças boas para a vida.

Ao professor e orientador Pablo, e coorientador Benjamin meus sinceros agradecimentos, por aceitarem me orientar e auxiliar neste trabalho.

"Para que a luz brilhe tão intensamente, a escuridão tem de estar presente.” - Francis Bacon.

RESUMO

A detecção facial, parte crucial do reconhecimento facial, tem sido um avanço tecnológico de crescente relevância em áreas como segurança, autenticação, análise forense e publicidade, sendo significativa a análise da influência da variação da iluminação. Nesse cenário, a exploração da interação entre a luminosidade e os algoritmos de identificação busca discernir como as flutuações na iluminação repercutem na precisão e confiabilidade do reconhecimento facial. Neste trabalho, a investigação dos impactos dessas variações nas características faciais por meio da análise de um banco de dados, para diferentes condições de iluminação, avalia a eficácia de métodos distintos para detecção facial, como o pré-processamento de imagens e algoritmos especializados de aprendizado de máquina. Os métodos comparados neste estudo foram modelos diferentes do Haar-Cascaded, HOG, Deep Neural Networks (DNN), Multi-Task Cascaded Neural Networks (MTCNN) e You Only Look Once (YOLO). Dentre esses, os melhores resultados foram obtidos com DNN e MTCNN. Com a interpretação dos resultados, pretende-se contribuir para a compreensão dos desafios impostos pela variação da iluminação em sistemas de detecção de faces, consolidando um embasamento crítico para futuros avanços.

Palavras-chave: luminosidade; características faciais; métodos.

ABSTRACT

Face detection, a crucial part of facial recognition, has been a technological advance of growing relevance in areas such as security, authentication, forensic analysis and advertising, and the analysis of the influence of lighting variation is significant. In this scenario, exploring the interaction between lighting and identification algorithms seeks to discern how fluctuations in lighting affect the accuracy and reliability of facial recognition. In this work, the investigation of the impact of these variations on facial features through the analysis of a database, for different lighting conditions, evaluates the effectiveness of different methods for facial detection, such as image pre-processing and specialized machine learning algorithms. The methods compared in this study were models other than Haar-Cascaded, HOG, Deep Neural Networks (DNN), Multi-Task Cascaded Neural Networks (MTCNN) and You Only Look Once (YOLO). Among these, the best results were obtained with DNN and MTCNN. By interpreting the results, the aim is to contribute to understanding the challenges posed by lighting variation in face detection systems, consolidating a critical foundation for future advances.

Keywords: brightness; facial features; methods.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CNN	Convolutional Neural Network
DNN	Deep Neural Network
FN	False Negative
FP	False Positive
HC-D	Haar-Cascaded Default
HC-FFAT	Haar-Cascaded FrontalFace Alt Tree
HC-FFA2	Haar-Cascaded FrontalFace Alt 2
HOG	Histogram of Oriented Gradients
IOU	Intersection Over Union
ML	Machine Learning
MLP	Multilayer Perceptron
MTCNN	Multi-task Cascaded Convolutional Networks
OpenCV	Open Source Computer Vision Library
RNA	Redes Neurais Artificiais
SIFT	Scale Invariant Feature
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
UFSC	Universidade Federal de Santa Catarina
YOLO	You Only Look Once

SUMÁRIO

1 INTRODUÇÃO

A identificação por meio da biometria é uma ferramenta de reconhecimento de indivíduos com base em características físicas ou comportamentais únicas e distintivas (Jain; Ross; Prabhakar, 2004). Essas características podem incluir impressões digitais, padrões faciais ou de retina, geometria da íris, geometria da mão, voz, termograma facial e assinatura (Jain; Hong; Pankanti, 2000). A tecnologia biométrica verifica ou identifica pessoas por meio dessas características, oferecendo elevada precisão e segurança (Pankanti; Bolle; Jain, 2000).

O reconhecimento facial é um recurso que utiliza a visão computacional a fim de identificar e validar a identidade de pessoas com base nos traços e características faciais, sendo que uma etapa essencial nesse processo é a detecção de faces. O desempenho dessa tecnologia pode ser afetado por diversos fatores: mudança na pose do indivíduo, alteração do ângulo em que é visto pela câmera; variação da idade da pessoa; oclusão, principalmente da porção superior da face; e a iluminação, para a qual a reflexão pode atrapalhar o processamento (Abate *et al.*, 2007).

O escopo do reconhecimento de faces abrange desde aplicações de segurança e personalização de experiências de usuário até aplicações comerciais. No âmbito da segurança, pode ser utilizado na autenticação do acesso virtual, no caso de sites e aplicativos, e físico, para entrada de portas, por exemplo, mas também, na identificação e rastreamento de indivíduos que cometeram crimes. Na área comercial, o reconhecimento facial pode ser aplicado para atendimento personalizado, em que as preferências são obtidas de maneira eficiente ao realizar a identificação automática de um cliente. Pode ser empregado, ainda, no treinamento de Inteligência Artificial, videogames, registro para documentos e realidade virtual (Zhao *et al.*, 2003).

Segundo Adini, Moses e Ullman (1997), um sistema de reconhecimento facial deve ser capaz de lidar com a variação na direção da iluminação. Um modelo facial eficiente para a computação e comparação do processo de reconhecimento deve conter um conjunto de imagens da mesma face. Além disso, de acordo com Adini, Moses e Ullman (1997) as funções que calculam distância entre uma face com pouca iluminação deve ser menor comparando a mesma pessoa em relação a outro indivíduo com condições padrões de luz. A título de exemplo, Adini Moses e Ullman (1997) utilizaram cinco imagens de 26 faces diferentes e perceberam que a mudança na direção da fonte de luz da esquerda para a direita resultou em falhas em mais de 50% dos casos, mesmo em circunstâncias controladas. Com isso, concluíram que é importante analisar parâmetros de imagem adicionais, como o estilo do cabelo e o posicionamento da face na imagem.

A influência da variação da iluminação nas imagens faciais, portanto, tem impacto direto na precisão e robustez dos sistemas de detecção de faces, e por consequência, no reconhecimento facial, já que a iluminação é uma variável complexa e multifacetada, suscetível a fatores como fontes de luz naturais e artificiais, condições climáticas e ambientes físicos (Gamage; Seneviratne, 2014). Tais variações podem distorcer a aparência dos rostos, prejudicando a capacidade dos algoritmos de reconhecimento em identificar indivíduos com precisão. Portanto, compreender e mitigar os efeitos da variação na iluminação é essencial para o desenvolvimento de sistemas de reconhecimento facial confiáveis e eficientes (Gamage; Seneviratne, 2014).

De acordo com Zhao *et al.* (2003), a detecção e o reconhecimento facial são duas etapas cruciais na visão computacional. A deteção de rostos é o processo de localização de rostos em imagens ou vídeos, determinando a presença e a posição de quaisquer regiões semelhantes a rostos. É o passo preliminar que identifica se um rosto está presente e onde está localizado. Por outro lado, o reconhecimento de faces vai um passo mais além, identificando ou verificando a identidade da face detectada (Zhao *et al.*, 2003). Enquanto a deteção de rostos se concentra em encontrar rostos, o reconhecimento de rostos envolve a comparação do rosto detectado com identidades conhecidas numa base de dados para determinar a identidade da pessoa (Zhao *et al.*, 2003).

Considerando isso, este trabalho explora a interação entre a variação da iluminação e o desempenho de diferentes métodos de detecção de face. A análise dessa influência pode fornecer melhor compreensão para qual método empregar em condições de iluminação diversa, ao comparar a eficácia da detecção facial. Para o estudo aqui realizado, os seguintes métodos são comparados: o algoritmo Haar-Cascaded, utilizando diferentes versões, por meio da biblioteca Open Source Computer Vision Library (OpenCV, 2023), Histogram of Oriented Gradients, por meio da biblioteca Dlib (dlib, 2023), You Only Look Once (Redmon *et. al*, 2016), Multi-task Cascaded Convolutional Networks (Zhang *et al.*, 2016) e a rede neural profunda, Deep Neural Network (LeCun *et al.*, 2015), aplicada a detecção facial. Referente a variação da iluminação, a base de dados escolhida possui 5 grupos diferentes, com relação ao ângulo da luz, e neste estudo foram separadas em Muito Claro, com ângulos de 0° a 12°, Claro, de 12° a 25°, Médio, com 26° a 50°, Escuro, com ângulos entre 51° e 70° e Muito Escuro com 71° a 130°.

1.1 OBJETIVOS

Com o intuito de aprofundar a compreensão dos impactos da luminosidade no contexto da detecção de faces, este trabalho busca investigar como as variações na iluminação afetam a precisão e a confiabilidade dos algoritmos de identificação.

Além disso, avalia os diferentes métodos escolhidos para compreender o desafio que a luz apresenta no campo da detecção facial. Assim, propõem-se neste trabalho os seguintes objetivos.

1.1.1 Objetivo Geral

Analisar a influência da variação da iluminação na identificação da biometria utilizando a detecção facial com um método controlado e sistemático.

1.1.2 Objetivos Específicos

A fim de alcançar o objetivo geral, serão necessários os seguintes objetivos específicos:

- Compreender conceitos de Redes Neurais Artificiais e aprendizado de máquina;
- Descrever como a Inteligência Artificial processa a informação para a detecção de face no reconhecimento facial;
- Compreender o funcionamento dos métodos escolhidos e suas especificidades;
- Comparar as detecções das faces do banco de dados nos diferentes métodos;
- Avaliar os resultados obtidos.

1.2 ORGANIZAÇÃO DO TEXTO

O trabalho inicia com a fundamentação teórica, apresentada no Capítulo 2, que ajuda a compreender conceitos e pesquisas relacionadas. Em seguida, no Capítulo 3, aborda-se a escolha da base de dados usada no estudo, a divisão do conjunto de dados conforme a iluminação, e a preparação dos dados para as análises. Com esses fatores devidamente definidos, é realizada a análise das detecções obtidas com variação na iluminação, cujos resultados são avaliados com métricas usuais da literatura no Capítulo 4. Por fim, a discussão, conclusão e o apontamento de possíveis trabalhos futuros são apresentados no Capítulo 5.

2 FUNDAMENTAÇÃO TEÓRICA

As bases teóricas subjacentes ao reconhecimento facial são essenciais para compreender como a variação da iluminação afeta as imagens de rostos, e explorar a natureza dessa variação é crucial para avaliar os efeitos nas características faciais. Além disso, é importante considerar as abordagens existentes para minimizar os efeitos indesejados, isso inclui a aplicação de técnicas de pré-processamento de imagens, juntamente com o uso de algoritmos de aprendizado de máquina especializados.

Em vista de uma sociedade cada vez mais conectada e orientada por dados, compreender e superar a influência da variação da iluminação na detecção de faces é essencial para promover a adoção responsável e eficaz dessa tecnologia no reconhecimento facial. Este trabalho é um esforço direcionado para ampliar o conhecimento sobre tal problemática e contribuir para a evolução de sistemas de reconhecimento facial mais confiáveis.

Apresenta-se neste capítulo o conceito de imagens digitais e suas propriedades, visão computacional e sua aplicação a imagens digitais. Em seguida, a caracterização do reconhecimento facial e seus processos, com ênfase na detecção de faces. Define-se então aprendizado de máquina, em específico, a tarefa supervisionada, conceitos de Redes Neurais Artificiais, sua estrutura e como processam a informação. Para tanto, é necessário compreender as redes neurais convolucionais e redes neurais profundas. Esses conceitos, utilizados por alguns dos métodos escolhidos para a detecção facial e finalmente, analisar este trabalho em comparação com outras pesquisas existentes.

2.1 IMAGENS DIGITAIS

As imagens digitais são compostas por unidade discretas, os pixels, dispostos em 2 dimensões. Cada pixel codifica informações de cor e intensidade em um local da imagem, sendo que as imagens podem ter um ou mais canais de cores. Por exemplo, imagens em escalas de cinza tem apenas um canal, enquanto imagens RGB possuem 3 canais. A resolução de uma imagem corresponde à largura e altura em pixels, assim, quanto maior a resolução, mais pixels e mais detalhes serão representados, neste trabalho, as imagens de Yale (2001) são padronizadas em 640x480 pixels e em escalas de cinza.

A imagem é o processo inicial para o reconhecimento facial, composta por uma matriz M x N pixels, na qual os índices das linhas e colunas indicam a posição de um ponto na imagem e o valor constitui no nível de cinza do ponto. Essas imagens geralmente são em escalas de cinza devido a redução da dimensionalidade, padronização e redução do ruído, enquanto as imagens coloridas são mais custosas e

complexas em termos de processamento (Santana; Rocha; Santos, 2014).

Para que essas imagens possam ser eficazmente utilizadas em sistemas de reconhecimento facial é essencial submetê-las a diversas etapas de processamento. O processamento de imagens digitais envolve a aplicação de técnicas para melhorar a qualidade das imagens, extrair características relevantes e preparar os dados para a análise subsequente. Isso é particularmente importante para lidar com variações de iluminação, ruído e outras imperfeições que possam prejudicar essa atividade.

2.1.1 Processamento de Imagens Digitais

Segundo Goodfellow et al. (2016), o pré-processamento das imagens é uma etapa estritamente necessária para o desempenho correto da visão computacional aplicada a Redes Neurais Artificiais. Isso se deve ao fato de que as imagens podem ter origens e propriedades diferentes, o que poderia causar problemas para as camadas das redes profundas. Para evitar esses problemas, padroniza-se os pixels das imagens em intervalos como [0, 1] ou [-1, 1] por meio de redimensionamentos e normalizações. Além disso, no contexto do treinamento de redes neurais, o aumento do conjunto de dados de entrada, por meio de técnicas de aumento de dados, ajuda a diminuir o erro por generalização (Goodfellow et al., 2016).

De acordo com Russell e Norvig (2013), o processamento de imagens é inicialmente composto por operações como detecção de arestas, análise de texturas e cálculo do fluxo óptico. O ponto comum dessas operações é sua capacidade de execução com apenas alguns pixels localmente, sem a necessidade de conhecimento sobre os objetos ao redor.

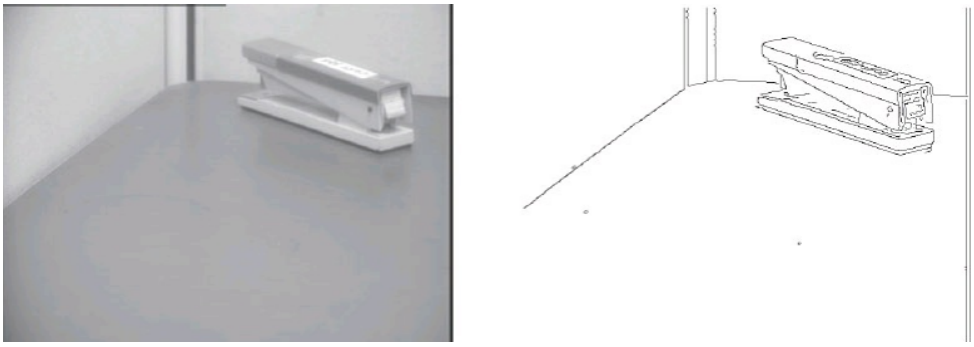
A detecção de bordas, por exemplo, abstrai a imagem desordenada em contornos compactos, que podem ser linhas retas ou curvas que diferenciam o brilho da imagem. Essa detecção não leva em conta possíveis descontinuidades, que podem ser classificadas em descontinuidades de profundidade, orientação da superfície, refletância e iluminação (Russell; Norvig, 2013).

A Figura 1 exibe uma imagem de um grampeador sobre uma superfície e à direita, a saída de um algoritmo que detecta bordas a partir da imagem original.

Nota-se que há diferenças entre a imagem original e as bordas produzidas que se devem ao cálculo dessas arestas, que nesse caso, envolvem o perfil do brilho em uma seção transversal perpendicular a uma das arestas. Por causa disso, a descontinuidade de iluminação não reproduziu corretamente uma parte do grampeador, mas esse erro é corrigido em etapas posteriores do processamento.

Segundo Russel e Norvig (2013), o próximo passo é a análise de texturas, que no contexto de visão computacional, é um padrão reproduzido no espaço da superfície, desse modo, ao contrário do processo anterior, é coerente apenas em

Figura 1 – Exemplo de imagem e suas bordas

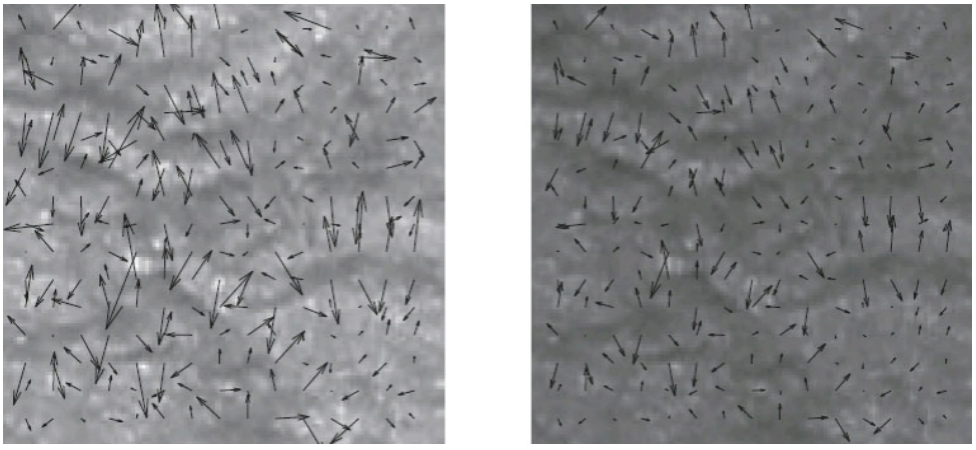


Fonte: Russel e Norvig, 2013

relação a múltiplos pixels, além disso, as texturas tem potencial de alterar os resultados com base na variação da iluminação, já que a orientação não muda, e outra vantagem se comparado às arestas é que algumas arestas importantes podem ser perdidas em imagens de objetos texturizados.

A Figura 2 apresenta o campo vetorial gradiente da mesma textura de uma imagem com dois níveis de iluminação diferentes. Esse processo já pode corrigir o erro gerado na detecção de arestas, pois as orientações desse campo vetorial permanecem as mesmas com alteração na iluminação.

Figura 2 – Exemplo de texturas com diferentes iluminações



Fonte: Russel e Norvig, 2013

Ao observar o impacto da variação na iluminação, nota-se que os vetores gradientes tendem a encurtar à medida que a luz diminui, embora sua orientação permaneça constante. Além disso, é importante destacar que a detecção de arestas apresenta menor eficiência em texturas rugosas em comparação com superfícies mais lisas.

O fluxo óptico, por sua vez, apresenta informações sobre a direção de velocidade de um objeto em uma cena ou sequência de imagens, sendo útil ao determinar a estrutura e as ações executadas, porém como o intuito deste trabalho

é analisar apenas imagens estáticas, esse procedimento não traz dados pertinentes (Russell; Norvig, 2013).

A Figura 3 mostra o exemplo do fluxo óptico em imagens extraídas de um vídeo separada em 2 quadros em sequência. A imagem a direita dos quadros ilustra o campo do fluxo óptico que corresponde à passagem do primeiro quadro, ou *frame* em inglês, para o segundo.

Figura 3 – Exemplo do fluxo óptico

Fonte: Russel e Norvig, 2013

É possível notar que as setas do campo do fluxo gerado indicam que o movimento capturado pelas imagens está acontecendo da direita para a esquerda, no caso da perna da pessoa e de cima para baixo, para a raquete e, um movimento diagonal com o braço. Pode-se supor, a partir dessas setas, que o movimento da perna possui uma velocidade maior comparado aos outros dois, devido a maior densidade de setas no campo do fluxo óptico.

2.2 VISÃO COMPUTACIONAL

A visão computacional consiste em descrever o mundo que é visto por meio de uma ou mais imagens e reconstruir suas propriedades, como forma, iluminação e distribuição de cores. Trata-se de uma área interdisciplinar que desenvolve técnicas para que computadores possam interpretar e compreender o mundo visual de forma similar aos humanos. As aplicações da visão computacional podem abranger segurança, como monitoramento e inspeção, modelagem em 3D e captura de movimento, e biometria, como autenticação de digitais e detecção de faces, entre outros (Szeliski, 2022).

A visão computacional, conforme Szeliski (2022), envolve a análise e interpretação de imagens digitais, permitindo que os computadores extraiam informações úteis do mundo visual. Imagens digitais, capturadas por câmeras e sensores, são processadas por algoritmos de visão computacional que identificam padrões, objetos e cenas, transformando dados visuais em conhecimento acionável.

A visão computacional é um campo de estudo que permite aos computadores interpretarem e compreenderem o conteúdo visual do mundo real, transformando imagens digitais em informações úteis e acionáveis. Segundo Forsyth e Ponce (2002),

essa disciplina interdisciplinar engloba técnicas avançadas de processamento de imagem, como reconhecimento de padrões, segmentação de objetos e reconstrução 3D. Essas técnicas são fundamentais para uma variedade de aplicações práticas, desde a automação industrial até a análise forense e a realidade aumentada.

Além de suas aplicações industriais e científicas, a visão computacional desempenha um papel crucial em aplicações voltadas para o consumidor, como reconhecimento de faces em redes sociais e assistentes pessoais virtuais. Conforme destacado por Forsyth e Ponce (2002), a capacidade dos sistemas de visão computacional de entender e interpretar imagens abre caminho para novas formas de interação homem-máquina e experiências digitais mais imersivas. Essa capacidade também impulsiona avanços na área de biometria, onde sistemas automatizados podem identificar indivíduos com base em características faciais únicas, reforçando a segurança e a autenticação em diversos contextos modernos.

Entre as diversas aplicações da visão computacional, o reconhecimento facial emerge como uma inovação e é amplamente utilizado. Esta tecnologia, que será detalhada em seguida, exemplifica como a visão computacional pode ser aplicada para identificar e verificar indivíduos com base em suas características faciais, oferecendo soluções eficazes em áreas como segurança, autenticação biométrica e interação humana-computador (Szeliski, 2022). Sendo assim, esta habilidade de compreender o ambiente visual é crucial para várias aplicações, incluindo o reconhecimento facial, que será detalhado a seguir.

2.2.1 Reconhecimento Facial

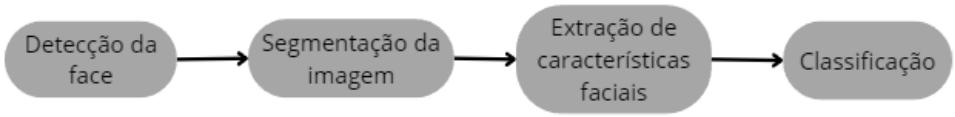
O reconhecimento facial é um dos ramos da visão computacional que utiliza redes neurais convolucionais (CNNs), perceptrons multicamadas (MLPs) e aprendizado supervisionado, todos componentes do aprendizado de máquina. Esse processo é utilizado para identificação de indivíduos através da biometria e, além de ser uma abordagem que está em constante evolução, é pouco invasivo.

Para o processo de reconhecimento facial é necessário seguir os passos: a aquisição de imagens, a detecção de faces, a segmentação, a extração de características faciais e por fim a de classificação de uma face (Santana; Rocha; Santos, 2014).

A Figura 4 apresenta um diagrama das etapas do reconhecimento facial, detalhando o fluxo do processo até a identificação final, considerando que a imagem já foi adquirida. Este diagrama é fundamental para compreender os passos sequenciais e os métodos aplicados em cada fase do reconhecimento facial.

O diagrama destaca as principais etapas envolvidas no processo de reconhecimento facial, desde a aquisição inicial das imagens até a classificação final.

Figura 4 – Diagrama de etapas do reconhecimento facial



Fonte: Autor, 2024

Cada etapa desempenha um papel crucial na identificação precisa e eficiente de indivíduos. A seguir, as etapas são explicadas separadamente, fornecendo uma visão detalhada dos métodos e técnicas utilizados em cada fase do processo.

Para verificar a presença de uma ou mais faces em uma imagem, diferentes tipos de algoritmos e softwares com Inteligência Artificial podem ser utilizados. É nessa etapa que surgem os primeiros desafios com relação aos algoritmos. As dificuldades englobam as variações excessivas na iluminação, como reflexos e sombras, ruído de fundo, elementos não relacionados à face no fundo da imagem, imagens de baixa qualidade e resolução, oclusão, quando a face está sendo obstruída por outros objetos, expressões faciais e movimento (Santana; Rocha; Santos, 2014). Este trabalho visa analisar os aspectos relacionados à influência da variação da iluminação na detecção de face para o reconhecimento facial.

Conforme Santana; Rocha e Santos (2014), após a detecção, efetua-se a segmentação da imagem, onde é delimitada a área de interesse, neste caso, o rosto e as partes relevantes, como olhos, boca e nariz, e descarta-se o resto da imagem. Essa parte ajuda o sistema a funcionar de maneira eficaz, melhorando a precisão do algoritmo ao concentrar as informações pertinentes. De acordo com Gonzalez (2009), a segmentação pode ser feita com métodos como limiarização, crescimento de regiões, divisão e fusão de regiões, além de técnicas baseadas em bordas, são essenciais para isolar objetos de fundo ou separar diferentes componentes de uma imagem.

Ainda em relação a segmentação, segundo Gonzalez (2009), essa pode ser entendida como o ato de particionar a imagem em pixels com propriedades similares, brilho, cor e textura por exemplo. Assim, esse particionamento preserva ou altera de maneira pouco significativa as propriedades de um mesmo objeto. A segmentação de imagem é um processo que envolve a divisão de uma imagem em suas partes constituintes ou objetos, facilitando a análise e interpretação das mesmas (Gonzalez, 2009).

Embora esses métodos sejam eficazes, o método You Only Look Once (YOLO) oferece uma abordagem alternativa, realizando a detecção de objetos em uma única etapa, incluindo a detecção de faces.

Em seguida, o algoritmo, que pode diferir daquele responsável pela detecção facial, segundo Santana *et al.* (2014), precisa extrair as características específicas

de cada indivíduo, no entanto, essa etapa apresenta um desafio adicional: o volume excessivo de dados a serem processados pelos algoritmos, portanto, é crucial reduzir a quantidade de características, ou seja, a dimensionalidade dos dados, porém essa redução deve ser cuidadosamente equilibrada, uma vez que qualquer perda significativa de informações pode comprometer a qualidade da análise. A extração de características é responsável por encontrar um conjunto mínimo de atributos que possa discernir uma face da outra, entre elas, os olhos, sobrancelhas, nariz e boca, com suas propriedades, orientação, distância e tamanho (Santana; Rocha; Santos, 2014).

Por fim, a classificação aponta para qual pessoa, de um determinado conjunto finito, pertence à face reconhecida. Esse processo também se dá por meio de algoritmos de Inteligência Artificial com diferentes métricas e métodos (Santana; Rocha; Santos, 2014). Este trabalho está focado na primeira etapa do reconhecimento facial, a detecção de face.

2.3 APRENDIZADO DE MÁQUINA

A aprendizagem refere-se a capacidade de melhorar o desempenho em tarefas futuras a partir de observações. No caso do aprendizado de máquina, ou machine learning (ML), o foco está no desenvolvimento de algoritmos e modelos que permitem computadores tomar decisões com base em dados, sem serem explicitamente programados. Esse aprendizado envolve os dados de entrada para o agente, o algoritmo de aprendizado, treinamento e avaliação, podendo ser dividido em aprendizagem por reforço, não supervisionado e supervisionado (Russell; Norvig, 2013). Neste trabalho, o aprendizado de máquina utilizado é o aprendizado supervisionado.

Segundo Russel e Norvig (2013), a aprendizagem supervisionada consiste em situar o agente para observar exemplos de pares de entrada e saída já rotulados e aprender uma função que mapeia da entrada para saída em rótulos ou classes. A função que faz esse mapeamento é escolhida entre um espaço de hipóteses possíveis na qual possui melhor desempenho, consistência e simplicidade, mesmo em novos exemplos, além do conjunto de treinamento.

O problema da aprendizagem supervisionada pode ser separado em regressão, quando a saída da função é um conjunto não enumerável de valores, e classificação, em que a saída é um conjunto finito de valores, como no reconhecimento facial. Utilizando o aprendizado supervisionado acompanhado com Redes Neurais Artificiais, pode-se aprender representações complexas dos dados (Russell; Norvig, 2013).

Uma vez estabelecida a importância do aprendizado de máquina, é essencial explorar as técnicas específicas que permitem sua aplicação eficaz em diversos domínios. Entre essas técnicas, as Redes Neurais Artificiais se destacam como uma das abordagens promissoras.

2.4 REDES NEURAIS ARTIFICIAIS

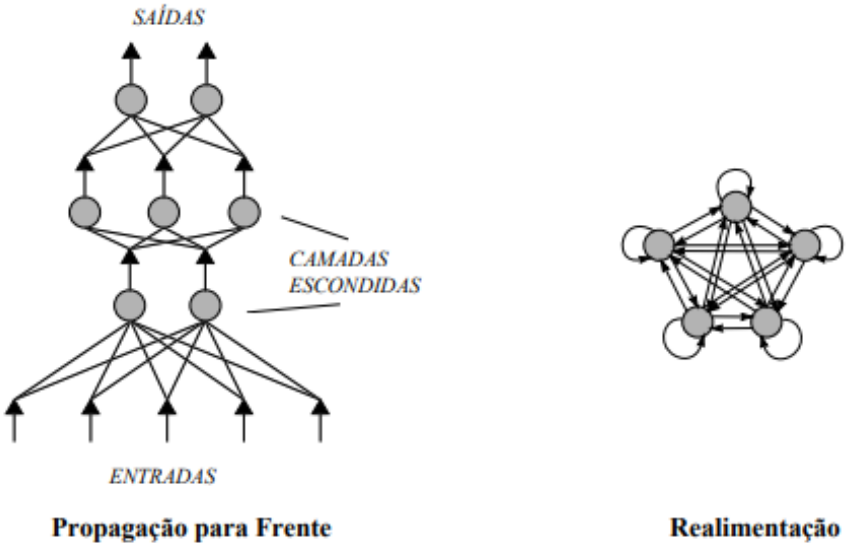
Redes Neurais Artificiais (RNAs) são modelos matemáticos inspirados pelo funcionamento do cérebro humano. Consistem em uma rede de unidades de processamento interconectadas, chamadas de neurônios artificiais ou nós, que colaboram para realizar tarefas de processamento de informações, como reconhecimento de padrões, classificação e tomada de decisões (Russell; Norvig, 2013).

De acordo com Russel e Norvig (2013), essas redes são capazes de fazer cálculos distribuídos, recebendo entradas com distorções e, com isso aprender. Os nós recebem informações propagam a ativação por meio de ligações direcionadas, as quais possuem pesos associados. Cada unidade calcula a soma ponderada de suas entradas com uma função de ativação para produzir uma saída.

O processamento dessas informações, juntamente com a realimentação das saídas dos nós às próprias entradas, leva ao aprendizado. Esse processo é realizado por múltiplas camadas, procurando sempre melhorar o resultado (Russell; Norvig, 2013). As RNAs, quando aplicadas ao processamento de imagens, operam especialmente bem em tarefas como reconhecimento facial e detecção de objetos, onde aprendem a partir de conjuntos de dados vastos e variados, ajustando seus parâmetros ao longo do treinamento para otimizar o desempenho nas tarefas visuais específicas.

A Figura 5 exemplifica a arquitetura de uma rede neural artificial, à esquerda uma rede com *feed forward*, e duas camadas escondidas, e à direita, uma rede realimentada.

Figura 5 – Dois exemplos de RNA



Fonte: Rauber, 2005

Segundo Rauber (2005), essas duas topologias de neurônios são fundamentais para formação das Redes Neurais Artificiais, onde no caso da propagação para frente o fluxo da informação é unidirecional, enquanto na realimentação não, e as ligações entre neurônios não tem restrições, resultando em um comportamento mais dinâmico. As camadas de neurônios que não estão ligadas diretamente à entradas e saídas são chamadas de camadas escondidas.

2.4.1 MultiLayer Perceptron (MLP)

A utilização de múltiplas camadas em Redes Neurais Artificiais, os Multilayer Perceptrons (MLPs), se configura como uma ferramenta para solucionar problemas complexos. Essa eficácia se deve à capacidade dos MLPs de decomporem a aprendizagem de uma função complexa com múltiplas saídas em uma série de operações de aprendizagem mais simples, cada uma realizada por uma camada intermediária. A camada oculta, entre a camada de entrada e camada de saída, cada uma com sua respectiva contribuição e peso. Essa decomposição permite a MLP aprender funções não lineares complexas, tornando-as capazes de lidar com problemas mais difíceis, como classificação de imagens, processamento de linguagem natural e reconhecimento de padrões (Russell; Norvig, 2013).

Segundo Russel e Norvig (2013), o ajuste dos pesos das camadas ocultas ocorre com a retropropagação do erro da camada de saída para as intermediárias, tendendo a minimizar o erro das previsões da rede em relação aos valores reais com derivadas do erro atual. No entanto, quando se trata de imagens, as MLPs possuem limitações no desempenho, em razão da capacidade de aprender representações complexas em dados tabulares ou sequenciais, então, se combinadas com as redes neurais convolucionais, que são projetadas para processar informações como as de imagens aperfeiçoando os resultados (Goodfellow; Bengio; Courville, 2016).

2.4.2 Redes Neurais Convolucionais

Redes Neurais Convolucionais, ou também Convolutional Neural Network (CNNs), são Redes Neurais Artificiais especializadas em processamento de dados com uma topologia conhecida, como as grades, em uma dimensão (1D), por exemplo as séries temporais, ou em duas dimensões (2D), como imagens de pixels. O nome convolucional é devido à operação matemática realizada, a convolução, que é empregada no lugar da multiplicação geral de matrizes em suas camadas. A convolução é um processo que combina duas funções para produzir uma terceira, destacando características específicas das entradas, como bordas, texturas e padrões, ao aplicar filtros sobre a imagem de entrada. (Goodfellow; Bengio; Courville, 2016).

Conforme descrito por Goodfellow *et al*. (2016), as redes neurais convolucionais

(CNNs) estabelecem interações esparsas entre cada unidade de entrada e saída, juntamente com pesos também esparsos, em contraste com as Redes Neurais Artificiais tradicionais, que dependem da multiplicação de matrizes com parâmetros para descrever a relação entre entradas e saídas. Esse design é particularmente eficaz no processamento de imagens, onde milhares de pixels estão presentes. As CNNs têm a capacidade de capturar pequenas características significativas usando kernels de tamanho reduzido, muitas vezes com apenas algumas dezenas ou centenas de pixels.

Além disso, o processo de convolução em contextos de redes neurais refere-se à extração de características por meio de kernels individuais. Essa convolução ocorre em várias localizações espaciais simultaneamente, com várias camadas convolucionais executando esse processo em paralelo. Essa abordagem permite que a rede detecte padrões complexos em diferentes níveis de abstração, aumentando assim sua capacidade de aprendizado e generalização.

As CNNs utilizam interações esparsas (ou também, pesos esparsos), juntamente com parâmetros compartilhados por suas camadas, o que resulta em menos parâmetros necessários para uma mesma atividade de uma rede neural comum, com isso, há melhorias na eficiência estatística e nos requisitos de memória (LeCun; Bengio; Hinton, 2015). Segundo Goodfellow *et al.* (2016), a rede neural convolucional é tipicamente constituída por 3 estágios: no início a camada executa convoluções em paralelo para formar ativações lineares; em seguida, cada ativação linear passa por uma função de ativação não-linear; por fim, a função de agrupamento junta para que as saídas sejam ainda mais modificadas.

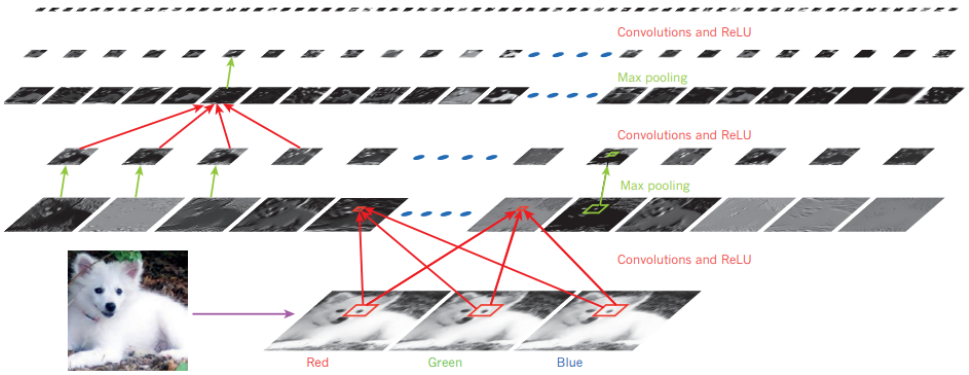
Tratando-se de uma imagem em tons de cinza passando por uma rede neural convolucional, as camadas de convolução desempenham o papel de realizar múltiplas operações de convolução diferentes em cada pixel. Nesse contexto, a entrada de uma camada pode ser a saída de outra, permitindo que a rede processe e extraia características complexas da imagem em tons de cinza. Combinando esse procedimento com a *backpropagation*, ou retropropagação, da saída para as entradas e para os pesos, após ter passado pelo menos uma vez pela rede com o feedforward, a propagação direta, é possível calcular o erro e os gradientes para então treinar a rede neural e otimizar seus resultados (Goodfellow; Bengio; Courville, 2016).

Outro estágio, conforme Goodfellow *et al.* (2016), é chamado de camada de *pooling* e auxilia no processo de tornar a representação de uma entrada invariante a uma translação da mesma em pequenas quantidades, assim a maior parte dos valores agrupados não é alterada. A invariância à translação local é especialmente útil na detecção de faces, para se determinar se algum recurso está ou não presente, ao contrário de saber precisamente onde está tal recurso. Um exemplo disso está no reconhecimento facial, no qual basta saber que há um olho no lado esquerdo e um olho no lado direito do rosto para definir a face e não necessariamente a posição exata

dos pixels (Goodfellow; Bengio; Courville, 2016).

A Figura 6 ilustra um exemplo de aplicação da CNN utilizando a imagem de um cachorro Samoieda, em que a rede foi empregada para classificação da raça do cão.

Figura 6 – Exemplo de aplicação da CNN



Fonte: LeCun; Bengio; Hinton (2015)

Como pode ser observado, a informação é processada de baixo para cima, passando inicialmente pela camada de *pooling*, após isso, introduzindo-se às camadas de convolução e a função de ativação unidade linear rectificada (Rectified Linear Unit - ReLU), e assim sucessivamente e, cada retângulo representa um mapa de características aprendidas e detectadas em determinadas posições da imagem.

2.4.3 Deep Learning

Deep Learning é um subcampo da Inteligência Artificial e do aprendizado de máquina que se concentra no treinamento de algoritmos de redes neurais profundas para aprender representações complexas dos dados a partir de conceitos mais simples por meio de hierarquias. Por exemplo, as redes feed forward (MLPs) combinam funções matemáticas de vários perceptrons, permitindo ao computador aprender sequências de instruções complexas. Isso resulta na capacidade de representar o mundo com uma hierarquia de conceitos aninhados e abstratos, cada um construído a partir de conceitos mais simples e concretos (Goodfellow; Bengio; Courville, 2016).

Conforme Goodfellow *et al.* (2016), a profundidade da *deep learning* proporciona ao computador a capacidade de aprender um conjunto de instruções em múltiplas etapas, onde cada camada de representação pode armazenar seu estado na memória após execução. Além disso, quanto maior a profundidade, mais instruções sequenciais as redes podem executar, o que resulta em uma capacidade aprimorada de processamento de dados.

Segundo LeCun *et al.* (2015), a aprendizagem profunda permite aos modelos computacionais aprender representações de dados com vários níveis de abstração, conseguindo descobrir estruturas complexas em grandes conjuntos de dados por meio

da retropropagação para guiar a Inteligência Artificial à maneira em que deve ser feita a alteração dos parâmetros internos no cálculo de todas as camadas, cada uma a partir da resposta da camada anterior.

Parte fundamental desse aprendizado se deve ao fato de que as camadas de características não são projetadas por humanos, mas aprendidas com os dados, por meio de um procedimento de aprendizagem generalizado. A base de sua estrutura são módulos simples, e todos ou a maior parte deles passam pelo processo de aprendizado, mapeando entradas e saídas de forma não linear. Assim, um dado que passa por várias dessas camadas não lineares, sendo uma profundidade maior, é sensível a detalhes pequenos em detrimento de variações maiores, como o plano de fundo e o ambiente. (LeCun *et al.*, 2015).

De acordo com Goodfellow *et al.* (2016), as redes neurais de aprendizagem profunda são compostas por redes neurais convolucionais e redes neurais recorrentes, possuindo centenas de hiperparâmetros e realizando algoritmo de *backpropagation* e a escolha da função de custo, com modelos lineares, além de normalização de lotes, processo que reparametriza a rede, desse modo, reduz o problema de atualizar múltiplas camadas ordenadamente, atuando na padronização da média e variância de unidades lineares, ao passo que permite a alteração das não lineares. Também possui camadas que são visíveis e camadas ocultas entre a entrada e saída características de Redes Neurais Artificiais.

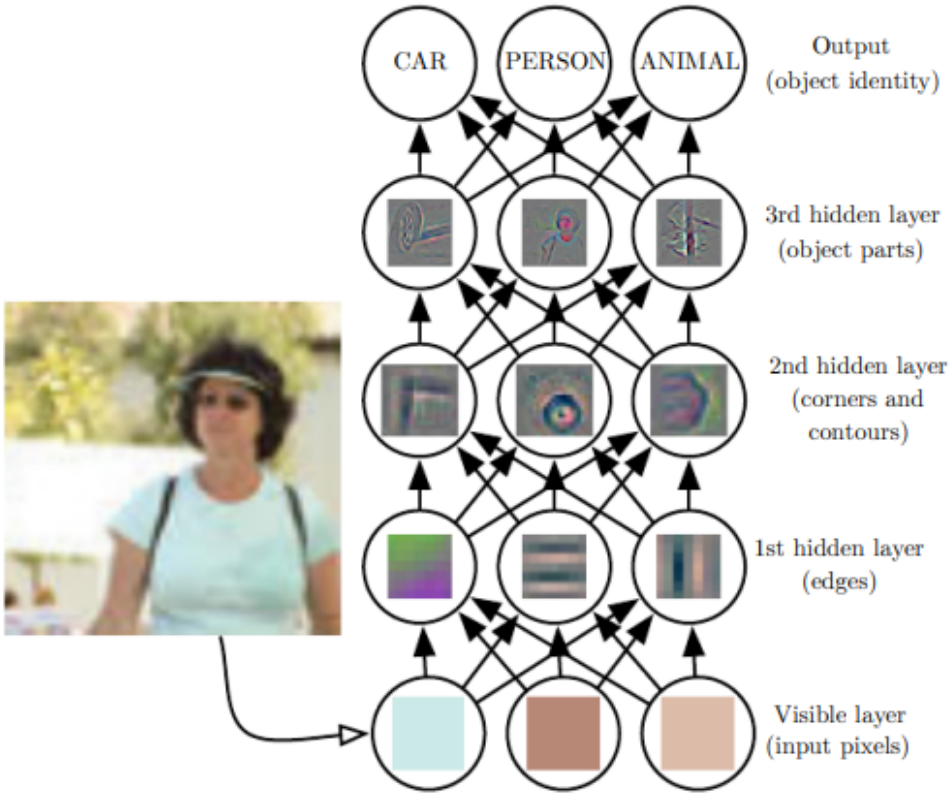
A Figura 7 ilustra um modelo que utiliza deep learning, onde pode ser visto ao lado esquerdo, a imagem de uma pessoa, que ao passar pela rede neural profunda, primeiramente divide os pixels na camada visível de entrada, para serem processados nas camadas ocultas, que tem o papel de identificar as arestas, contornos e partes, respectivamente, ao comparar o brilho dos pixels vizinhos.

Por fim, esses dados, após processados por todas as camadas, têm como resultado a classificação da entrada em carro, pessoa ou animal, nesse caso. As redes neurais profundas podem ter profundidade variável, que é baseada na quantidade de intruções sequenciais em sua estrutura, quanto maior o número de camadas ocultas, maior será o nível de abstração dos dados e maior a profundidade.

2.4.4 Multi-Task Cascaded Neural Network (MTCNN)

As redes neurais de convolução multitarefa em cascata combinam alinhamento facial e regressão de pontos faciais e sua estrutura em cascata inclui redes convolucionais profundas de três estágios, ou camadas, sendo eles: P-net, R-net e O-net. A primeira etapa consiste em passar a imagem em janelas candidatas produzidas por meio da P-Net, que utiliza um filtro 3x3 para convolução das imagens de entrada de 12x12 e obtém a classificação facial e respectiva *bounding box* usando um filtro 1x1

Figura 7 – Modelo de aprendizagem profunda



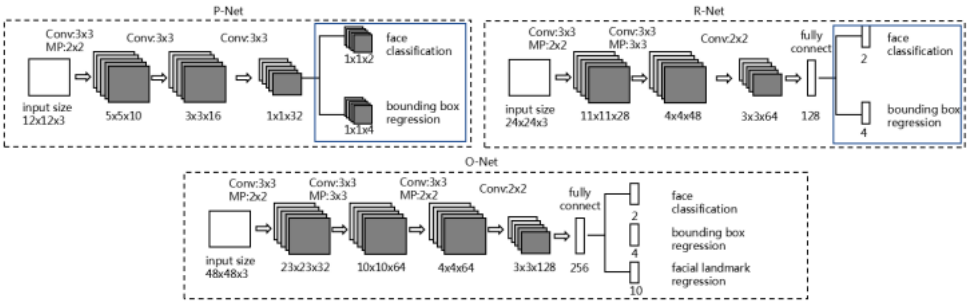
Fonte: Goodfellow *et al.* (2016)

na última camada com dois canais diferentes para convolução, não é estruturada com camadas totalmente conectadas. Após isso, os candidatos seguem para um processo de refinamento ainda mais preciso na R-Net. Essa rede neural convolucional profunda utiliza camadas de convolução com filtros de diferentes tamanhos (3x3 e 2x2) para extrair informações mais complexas e discriminativas dos dados de entrada. Ao término desta rede, uma camada totalmente conectada faz a classificação e caixa delimitadora. Por fim, na O-Net é feito a *bounding box* final e a posição dos pontos de referência faciais e comparado as redes anteriores, possui uma camada de convolução a mais e a camada de convolução total é de 256, realizando a regressão das caixas e dos pontos de referência do nariz, olhos e boca (Zhang; Wang; Chen, 2021).

A Figura 8 mostra a estrutura da MTCNN proposta por Zhang *et al.* (2005), que consiste na estrutura em cascata de 3 fases de redes neurais convolucionais profundas, como citado anteriormente, P-Net, R-Net e O-Net, que foram integradas a duas tarefas na P-Net e R-Net, para classificação das faces e regressão das *bounding boxes* e, a O-Net que além de realizar as mesmas etapas, também faz a regressão das referências faciais, olhos, boca e nariz.

Pode-se observar que nas 3 redes dessa estrutura a imagem de entrada

Figura 8 – Estrutura da MTCNN



Fonte: Zhang *et al.* (2001)

passa por diferentes camadas de convolução apesar de possuírem um tamanho de entrada diferente, como também, nota-se a saída da R-Net e O-Net, que são totalmente conectadas, isto é, todos os neurônios estão conectados aos neurônios das camadas anteriores, com isso, facilita os processos posteriores, a classificação entre face ou não e a previsão das coordenadas da caixa delimitadora.

2.5 ALGORITMO VIOLA-JONES

Segundo Viola e Jones (2001), o algoritmo de mesmo nome propõe três contribuições para a estrutura da detecção de objetos: a imagem integral, que é calculada a partir de uma imagem com operações por pixel e após esse cálculo, as características da função de Haar podem ser avaliadas em qualquer escala ou local com tempo constante; um classificador que elege poucos atributos importantes por meio do AdaBoost, já que mesmo pequenas janelas de uma imagem possuem vários recursos, essa é uma maneira de limitar e generalizar os recursos críticos; por fim, combinar classificadores continuamente mais complexos dentro de uma estrutura em cascata, fato que aumenta a velocidade da detecção como também concentra o foco da atenção ao analisar a taxa de falsos negativos.

2.5.1 Detecção em cascata

Os classificadores em cascata podem ser construídos para rejeitar subjanelas negativas, ao mesmo tempo em que detectam quase todas as instâncias positivas. Isso é feito atribuindo classificadores mais simples para rejeitar parcelas negativas e classificadores mais complexos para reduzir a taxa de falsos positivos. O processo em cascata ocorre passando uma subjanela por sucessivos classificadores, sendo que ao ser testada e obter um resultado positivo, acionará um segundo classificador, com taxa de detecção maior e assim por diante. Resultados negativos em qualquer ponto da árvore levam à rejeição imediata da subjanela (Viola; Jones. 2001).

De acordo com Viola e Jones (2001), os estágios da cascata são construídos por classificadores de treinamento usando o AdaBoost e realizando ajustes no limite a fim de minimizar falsos negativos, e geralmente limites mais baixos produzem taxas de detecção mais altas, porém taxas de falsos positivos maiores também. Como consequência da estrutura em cascata, grande parte das subjanelas de uma imagem é dada como negativa e são rejeitadas nos estágios iniciais, entretanto, devido ao fato de que os primeiros estágios são "mais difíceis", os estágios mais profundos detêm maior taxa de falsos positivos.

A estrutura, como descrita anteriormente, começa de modo simples e recursos são adicionados com a finalidade de atingir as taxas de detecção e falsos positivos, onde também são anexados mais estágios para cumprir as metas gerais. Assim, com o objetivo de realizar a ativação das camadas mais profundas por meio das iniciais, é necessário avaliar um detector de características em cada local agrupando e encontrando co-ocorrências de propriedades incomuns (Viola; Jones. 2001).

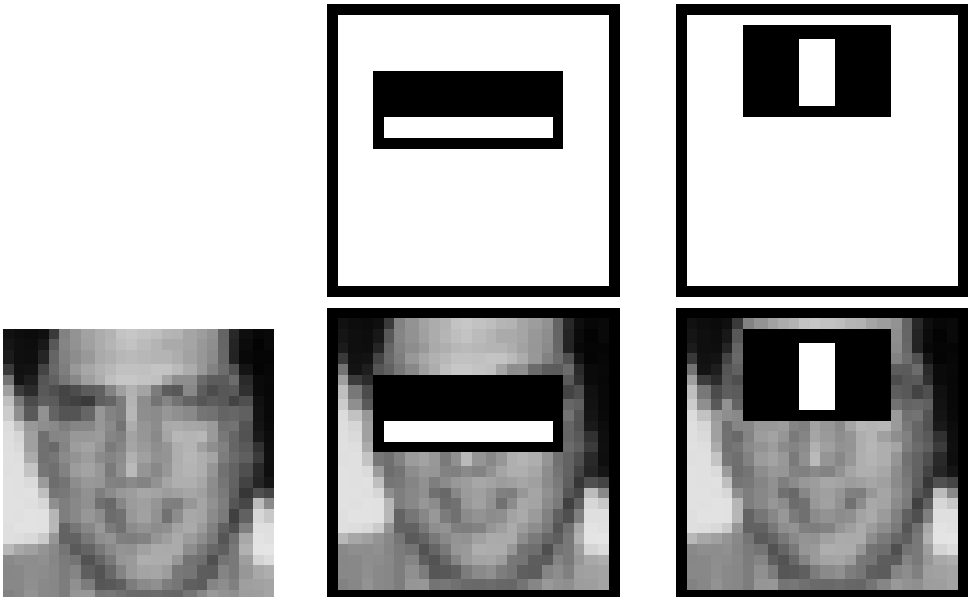
2.5.2 Modelo Haar-Cascade

O algoritmo de Haar-Cascade tem como base as três contribuições propostas por Viola e Jones (2001). Primeiramente, a imagem passa por um processo de cálculo da Imagem Integral, uma técnica que permite calcular a soma dos valores de pixel em qualquer retângulo da imagem. Em seguida, são formados retângulos (ou recursos Haar), previamente ordenados. A partir da análise das diferenças de iluminação em regiões específicas da imagem, determina-se a possibilidade de um fragmento conter um rosto. Alguns exemplos dessas características, também chamadas de *Haar-like features*, incluem diferenciações entre as sobrancelhas, o nariz e a boca em contraste com a pele. Essas características são representadas por retângulos divididos ao meio, com uma parte branca e outra preta. (Sousa *et al.*, 2002).

A Figura 9 ilustra um exemplo da aplicação proposta por Viola e Jones, que utiliza características discriminantes selecionadas por meio do AdaBoost, um algoritmo de aprendizado de máquina para classificação. A linha superior mostra duas dessas características, representadas por retângulos claros e escuros, e abaixo vemos a sobreposição dessas características na imagem de treinamento.

Os retângulos claros e escuros representam os filtros de Haar, que são usados para detectar características específicas na imagem. Esses filtros consistem em regiões claras e escuras que calculam a diferença de intensidade de pixels entre elas. A primeira característica (retângulo horizontal) compara a intensidade da luz na região dos olhos com as bochechas, sendo a região dos olhos frequentemente mais escura. A segunda característica (retângulo vertical) também se concentra nos olhos, mas compara a intensidade da luz com a região do nariz, que tende a ser mais clara que os

Figura 9 – Exemplo do algoritmo de Viola-Jones



Fonte: Viola; Jones (2001)

olhos. Essas comparações ajudam a identificar padrões comuns em rostos humanos, facilitando a detecção facial pelo algoritmo.

2.6 HISTOGRAMA DE GRADIENTES ORIENTADOS

Desenvolvido por Dalal e Triggs (2005), Histograma de Gradientes Orientados (HOG, sigla do inglês Histogram of Oriented Gradients) consiste em analisar histogramas locais normalizados de orientações de gradiente de imagem em uma grade densa por meio de sua distribuição, sem necessitar das bordas, ao subdividir a janela da imagem em pequenas regiões espaciais, as células, e atribuindo um histograma de 1 dimensão local para gradientes e orientações sobre os pixels, dessa forma, a combinação das entradas do histograma formam uma representação. Os blocos descritores podem ser normalizados a fim de melhorar a invariância à iluminação e sombreamento, capturando a estrutura da borda ou gradiente característicos e representando localmente e em vista disso facilita as transformações geométricas e fotométricas, como translações e rotações.

Segundo Dalal e Triggs (2005), o HOG é capaz de extrair bordas de um ou mais objetos ao analisar os gradientes dos histogramas e verificar se os pixels sofrem uma mudança abrupta de cor com a derivada de uma função multivariável, significando uma borda. Ao contrário de outros algoritmos como o Scale Invariant Feature Transform (SIFT), que também utilizam orientação de gradientes, os Histogramas de Gradientes Orientados (HOG) são especializados na detecção de seres humanos, utilizando orientação de gradientes em cada célula para melhorar a precisão dessa tarefa.

A sequência utilizada pelo algoritmo de HOG é dada pela sequência de passos em que a imagem de entrada é normalizada em cor e gama, então seus gradientes são computados, após isso os pesos são atribuídos com base no espaço e orientação das células, em seguida ocorre a normalização de contraste em blocos sobrepostos para assim os dados do HOG serem coletados na janela de detecção e por fim ser enviado a máquina de vetores de suporte linear para obter a classificação entre pessoa detectada ou não. A taxa de falsos positivos é reduzida em mais de uma ordem de grandeza se comparado as transformadas de Haar (Dalal; Triggs, 2005).

2.7 MÉTODOS E MÉTRICAS DE AVALIAÇÃO

A avaliação de algoritmos de aprendizado de máquina desempenha um papel crucial no desenvolvimento e na otimização de modelos. Dentre os diversos métodos de avaliação disponíveis, o método de avaliação proposto nesse trabalho é a matriz de confusão, que se destaca como uma ferramenta fundamental, fornecendo uma visão abrangente do desempenho do modelo ao comparar suas previsões com os resultados reais por meio de análises estatísticas e cálculos.

A matriz de confusão é uma ferramenta utilizada para avaliar o desempenho de algoritmos de classificação, é uma tabela que permite a visualização das previsões de um modelo em comparação com os resultados esperados, ajudando a quantificar a qualidade de classificação de um modelo. Essa matriz é particularmente relevante em problemas de classificação binária, mas pode ser estendida para problemas de classificação multiclasse (Muller; Guido, 2016).

Para verificar se os modelos de Inteligência Artificial utilizados detectaram as faces corretamente e as posicionaram de maneira precisa, empregou-se uma matriz que relacionou as *bounding boxes* (caixas delimitadoras) geradas pelos métodos com as ideais. Essas caixas são retângulos definidos por quatro coordenadas que definem as regiões de interesse, neste caso, as faces. O cálculo referente às *bounding boxes* é realizado pela *Intersection Over Union* (IOU), que leva em conta as áreas e um *threshold*, de 50% por exemplo, da interseção dividida pela união das áreas das duas caixas. As equações para o cálculo das áreas da *bounding box*, intersecção, união e a IOU, respectivamente, são demonstradas nas Equações (1), (2), (3) e (4), de acordo com Szeliski (2022).

$$Area = (x_2 - x_1) * (y_2 - y_1) \tag{1}$$

$$Intersection = (x_{2i} - x_{1i}) * (y_{2i} - y_{1i}) \tag{2}$$

$$Union = A_1 + A_2 - Intersection \tag{3}$$

$$IOU = \frac{Intersection}{Union} \tag{4}$$

Nas Equações 1 e 2, x_2, y_2, x_1 e y_1 representam as coordenadas do canto inferior direito e o canto superior esquerdo, que descrevem a *bounding box*, enquanto que x_{2i}, y_{2i}, x_{1i} e y_{1i} referem-se às mesmas coordenadas, mas da intersecção das duas caixas, já na Equação (3), A_1 e A_2, são as áreas das caixas e com as Equações (2) e (3) pode-se calcula a IOU, na Equação (4). Um exemplo numérico pode ser visto posteriormente, na Seção 3.3.

A partir do cálculo da IOU e a comparação com o *threshold*, é possível verificar em quais dos quadrantes da matriz de confusão as previsões se encaixam (Muller; Guido, 2016):

- Verdadeiro Positivo (*True Positive* - TP): Representa os casos em que o modelo classificou corretamente as instâncias da classe positiva. Caso onde há uma face e o algoritmo detectou uma face.
- Verdadeiro Negativo (*True Negative* - TN): Representa os casos em que o modelo classificou corretamente as instâncias da classe negativa. Caso onde não há uma face e o algoritmo não detectou uma face.
- Falso Positivo (*False Positive* - FP): Representa os casos em que o modelo classificou incorretamente instâncias da classe negativa como positiva (erro tipo I). Caso onde não há uma face e o algoritmo detectou uma face.
- Falso Negativo (*False Negative* - FN): Representa os casos em que o modelo classificou incorretamente instâncias da classe positiva como negativa (erro tipo II). Caso onde há uma face e o algoritmo não detectou uma face.

Segundo Muller e Guido (2016), ao obter os valores da matriz de confusão, é possível determinar os seguintes conceitos: Acurácia (Accuracy), a proporção de previsões corretas em relação ao total de previsões, calculada conforme a Equação 5:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{5}$$

Precisão (Precision), a proporção de verdadeiros positivos em relação ao total de instâncias classificadas como positivas, calculada pela Equação (6):

$$Precision = \frac{TP}{(TP + FP)} \tag{6}$$

Revocação (Recall), a proporção de verdadeiros positivos em relação ao total de

instâncias reais positivas, calculada por meio da Equação (7):

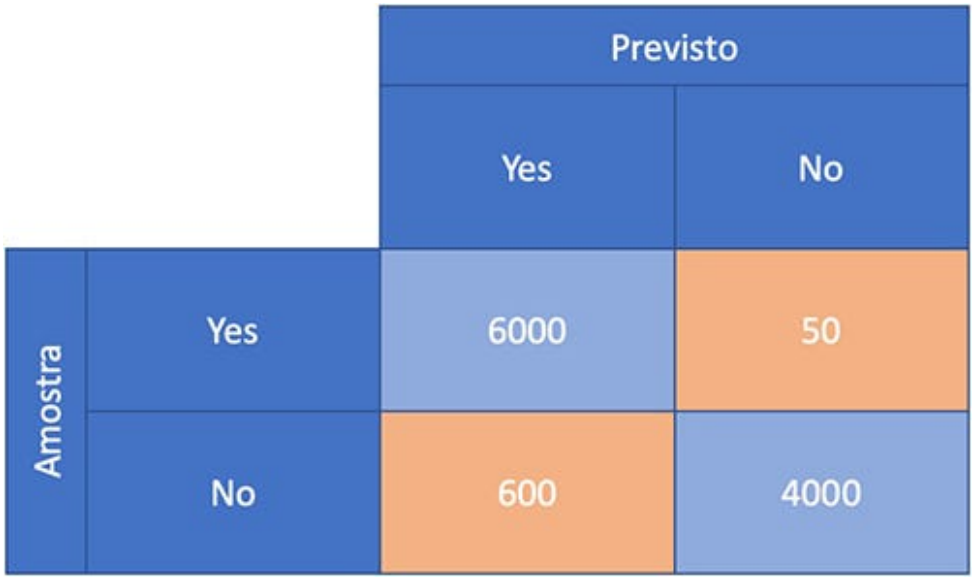
$$Recall = \frac{TP}{(TP + FN)} \tag{7}$$

F1-Score, uma métrica que combina precisão e recall em um único valor, útil quando o equilíbrio entre essas duas métricas é importante, calculada com a Equação (8).

$$F1_{score} = \frac{2(Precision * Recall)}{(Precision + Recall)} \tag{8}$$

A Figura 10 mostra uma das matriz de confusão genérica, com dados binários. A partir dessa matriz, é possível extrair os dados das equações anteriores e comparar com os demais algoritmos. Tendo os valores numéricos de TP, TN, FP e FN, pode-se calcular as métricas de precisão, revocação, acurácia e F1-Score.

Figura 10 – Exemplo de Matriz de Confusão Binária



Fonte: Lago, 2021

Pode-se perceber que conforme há maior quantidade de certo quadrante, a cor se torna mais intensa, representação comumente utilizada ao exibir matrizes de confusão. As matrizes de confusão não se limitam a casos binários, podem ser utilizadas para classificação multi-classe.

Sendo assim, a matriz de confusão fornece uma visão detalhada do desempenho do modelo, permitindo identificar se está cometendo mais erros tipo I (falsos positivos) ou tipo II (falsos negativos). Portanto, com os valores obtidos, pode-se ter uma visão sobre a eficácia dos diferentes modelos e softwares.

2.8 TRABALHOS RELACIONADOS

Investigações no campo do reconhecimento facial têm buscado compreender os desafios impostos pela variação da luminosidade e seus efeitos no desempenho dos sistemas de reconhecimento facial, e esses não apenas aprofundaram a compreensão das nuances da interação entre iluminação e características faciais, mas também propuseram abordagens inovadoras para mitigar os efeitos negativos dessa variação.

Em Behling (2019), o autor procurou identificar diferentes emoções em faces a partir de vídeos. A abordagem de reconhecimento facial teve melhor desempenho utilizando uma camada com múltiplos perceptrons, em detrimento do tempo para seu treinamento, enquanto que redes com máquinas de vetores de suporte, *Support Vector Machines* (SVM), são mais eficientes em relação ao treinamento, porém, geram mais erros conforme o conjunto de dados aumenta. Ainda, o autor apresenta que, as redes neurais profundas conseguem encontrar um rosto em uma imagem com uma taxa maior de acertos, mas a imagem deve ser processada anteriormente para amenizar os efeitos da mudança na iluminação.

Jagadiswary, Appasami e Rajesh (2011), por outro lado, realizaram a normalização das características dos olhos para a identificação humana, capaz de detectar, em imagens degradadas. Isso inclui o reconhecimento da íris, uma parte essencial na observação das características do rosto. Essa abordagem resultou em melhorias na precisão do reconhecimento com um método determinístico linear, que também pode ser utilizado em aplicações de tempo real.

Böhm (2021) concentrou-se em explorar a aplicação de redes neurais convolucionais (CNNs) para aprender representações robustas em sistemas embarcados. Abordagens como a de Böhm não apenas oferecem pontos de vista sobre como as redes neurais podem ser treinadas para discernir características consistentes apesar das mudanças de iluminação, mas também sugerem soluções promissoras para o desafio contínuo de melhorar a confiabilidade dos sistemas de reconhecimento facial.

De acordo com Kaur e Sharma (2023), um estudo comparativo entre abordagens de detecção de rosto utilizando Haar-Cascaded, Histogram of Oriented Gradients (HOG) com SVM e Multi-Task Cascaded Neural Network, constatou-se que o método Haar, apesar de simples, é robusto e eficiente para detecção de rostos posicionados de maneira frontal com 95% de precisão, mas obteve piores resultados em rostos de perfil, com óculos e oclusos, enquanto HOG conseguiu 98.6% de precisão para rostos frontais, pode detectar também rostos com óculos, mas teve desempenho pior quanto ao rosto de perfil e ocluso. Por fim, a MTCNN obteve o melhor resultado nesse estudo, com 99.7% de precisão e foi capaz de identificar rostos em todos os casos, o que sugere que MTCNN, por se tratar de um método mais recente, é menos propenso a erros mesmo com condições e restrições nas imagens (Kaur; Sharma,

2023). Porém, esse estudo comparou duas técnicas que se baseiam em aprendizado de máquina com redes neurais artificias, no caso do Haar Cascaded e HOG com SVM, enquanto a MTCNN utiliza 3 estágios e *deep learning*.

Ao examinar os trabalhos relacionados sobre a influência da variação da iluminação no reconhecimento facial, fica claro que essa é uma área de pesquisa em evolução constante. As abordagens inovadoras e as soluções propostas por esses estudos formam a base sobre a qual este trabalho se apoia, à medida que se busca uma compreensão profunda e eficaz da relação entre iluminação e precisão do reconhecimento facial.

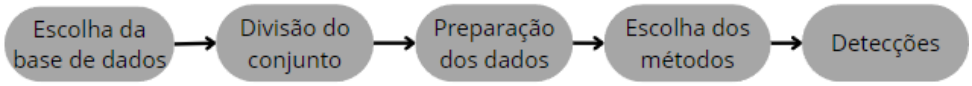
Tendo em vista esse objetivo, o procedimento empregado neste trabalho fundamenta-se nas Redes Neurais Artificiais, as quais foram treinadas para classificar e identificar as imagens, também em métodos baseados no algoritmo de Viola e Jones (2001) e de processamento de imagens, como HOG.

3 MATERIAIS E MÉTODOS

A metodologia começa com a definição, separação e preparação da base de dados, seguida pela escolha dos métodos a serem empregados. Após a definição dos objetivos a serem alcançados é realizada a seleção da base de dados que serve de fundamento para as análises. A divisão adequada do conjunto de dados é efetuada para garantir a robustez das análises. A escolha dos algoritmos de detecção a serem utilizados é feita com base na versatilidade, flexibilidade, desempenho, precisão e facilidade do uso.

A Figura 11 ilustra a sequência da metodologia seguida neste estudo, abrangendo desde a escolha da base de dados até as detecções dos resultados obtidos através dos diferentes métodos de detecção facial.

Figura 11 – Diagrama da metodologia



Fonte: Autor (2024)

A análise da variação da iluminação é conduzida avaliando os diferentes modelos e métodos com o conjunto de dados separados, permitindo maior compreensão de seus efeitos. Posteriormente, os resultados obtidos são avaliados. Finalmente, com base nas conclusões, possíveis direções para trabalhos futuros são destacadas, ampliando o escopo e a relevância das investigações no campo do reconhecimento facial.

3.1 ESCOLHA DA BASE DE DADOS

Em relação a base de dados para este trabalho, foi considerado a utilização de *Labeled Faces in the Wild*[1], IMDB-Wiki[2], *MUCT face database*[3], *CMU Multi-PIE Database*[4] e por fim, a base de dados de Yale. As duas primeiras opções foram descartadas devido a variação no posicionamento da face na imagem, o que dificultaria a análise, além de possuir pouca diferença na iluminação, já a *MUCT face database*, apesar de possuir variação na iluminação e ângulo, possuia um número menor de

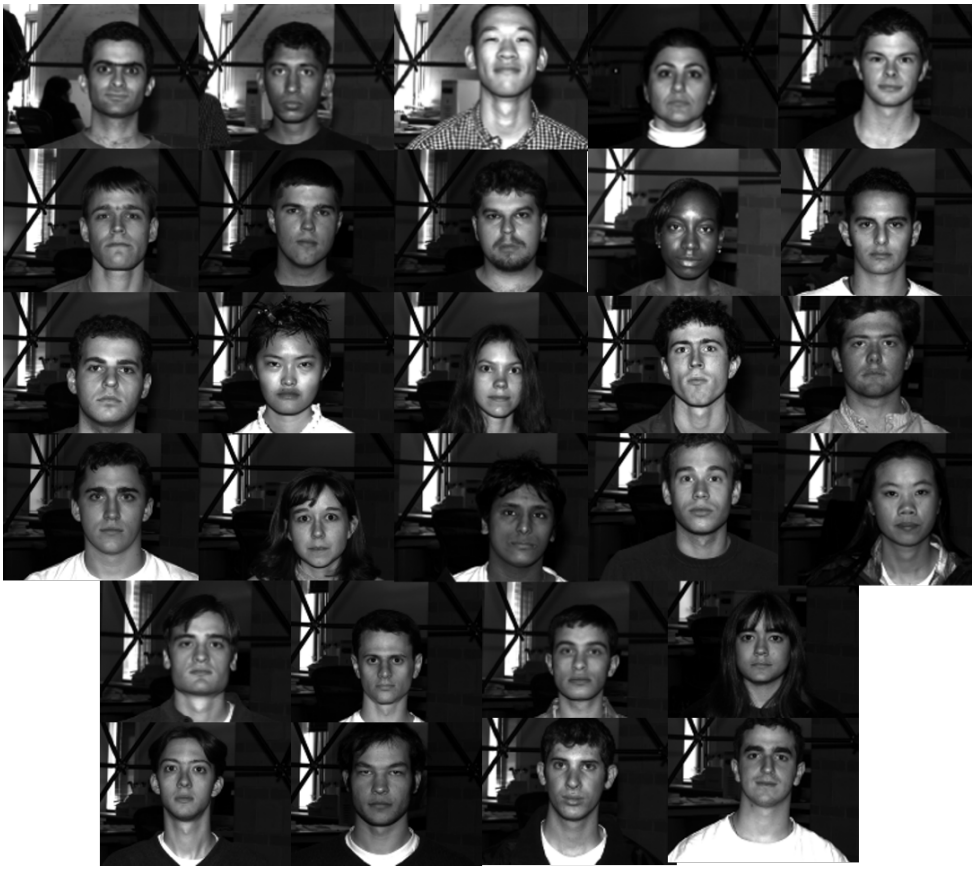
[1] Disponível em: http://vis-www.cs.umass.edu/lfw/
[2] Disponível em: https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/
[3] Disponível em: http://www.milbo.org/muct/The-MUCT-Landmarked-Face-Database.pdf
[4] Disponível em: http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html

imagens comparada a Yale. Por outro lado, a *CMU Multi-PIE Database* possui uma base de dados extensa, mas com menos condições de variação de iluminação se comparado a base de Yale.

A base de dados estabelecida foi a base de dados expandida de faces de Yale (Yale, 2001), essa escolha foi embasada em sua adequação aos objetivos deste estudo e por possuir mais imagens se comparado a base de Yale padrão. A base de dados de faces de Yale oferece uma ampla gama de imagens faciais capturadas sob várias condições de iluminação e ângulos, além de seu tamanho ser padronizado e possuir apenas tons de cinza. Tal diversidade de dados permite uma análise mais abrangente e precisa da influência da variação da iluminação no reconhecimento facial.

A Figura 12 mostra todos os 28 indivíduos da base de dados de Yale utilizados neste trabalho. Para exemplificar, as imagens mostradas fazem parte do primeiro grupo de iluminação, com o ângulo entre o eixo da câmera e a direção da fonte da luz entre 0° e 12° graus, onde pode-se ter uma noção de como a base foi feita, utilizando uma maneira padronizada para as fotos, apenas em tons de cinza, com a mesmo tamanho da imagem e rostos frontais.

Figura 12 – Indivíduos da base de dados com eixo da câmera em 0°



Fonte: Adaptado pelo autor, 2024

Também é importante notar que nessa base de dados não há objetos obstruindo a visão da face, como gorros, bonés, máscaras ou óculos, então a única oclusão das

faces é a sombra, causada pela iluminação, o que se alinha com os objetivos deste trabalho.

Vale destacar que a base de dados foi selecionada com a intenção de proporcionar um cenário sólido para as investigações deste estudo, enquanto novas possibilidades de bases de dados poderão ser exploradas em trabalhos futuros, enriquecendo ainda mais as análises e conclusões a serem obtidas.

3.2 DIVISÃO DO CONJUNTO DE DADOS

A base de dados expandida B de Yale contém 16380 imagens de 28 indivíduos em 9 poses diferentes e sob 64 condições de iluminação distintas. Essas condições de iluminação estão divididas em cinco subgrupos, conforme o ângulo entre o eixo da câmera e a direção da fonte de luz (Yale, 2001). Para facilitar a compreensão, os subgrupos foram nomeados conforme os ângulos: Muito Claro para ângulos menores que 12°, Claro para ângulos entre 13° e 25°, Médio para ângulos entre 26° e 50°, Escuro para ângulos de 51° a 70°, e Muito Escuro para ângulos maiores que 71°. Cada indivíduo tem suas imagens distribuídas da seguinte forma:

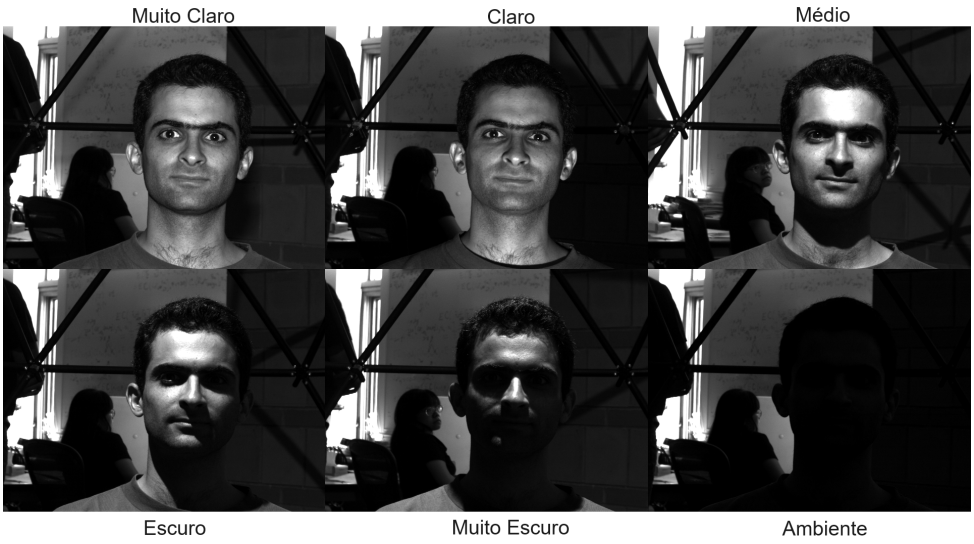
- 7 imagens com ângulos de iluminação menores que 12° (Muito Claro);
- 12 imagens com ângulos entre 13° e 25° (Claro);
- 12 imagens com ângulos entre 26° e 50° (Médio);
- 14 imagens com ângulos variando de 51° a 70° (Escuro);
- 19 imagens com ângulos maiores que 71° (Muito Escuro);
- 9 imagens do ambiente, com a face totalmente obscurecida.

Como o objetivo é avaliar apenas a questão da iluminação, em vez de separar as 585 imagens de cada indivíduo, a divisão do conjunto de dados foi feita com base nos ângulos de iluminação, resultando em 6 grupos de imagens conforme descrito acima. O último grupo, correspondente ao ambiente, é utilizado exclusivamente para a classificação de verdadeiros negativos, onde os algoritmos não devem detectar faces.

A Figura 13 mostra a separação dos 6 grupos diferentes de iluminação para este trabalho, onde pode-se ter uma noção de como os grupos estão divididos.

Para simplificar, o primeiro grupo é constituído com imagens que apresentam iluminação considerada padrão, o segundo com uma iluminação levemente menor, o terceiro onde já fica difícil ver alguns pontos da face, o quarto onde quase não é possível ver os olhos e um dos lados da face, o quinto em que a luz é escassa e sobra pouco visível do rosto e por fim, a imagem do ambiente, completamente sombreada.

Figura 13 – Exemplos separação dos dados



Fonte: Adaptado pelo autor, 2024

3.2.1 Preparação dos dados

A preparação dos dados foi realizada com um *script* em Python para criação das *bounding boxes* para marcação dos rostos, sendo essa a localização real da face na imagem. O *script* permitiu gerar as coordenadas da caixa com dois cliques do mouse, o primeiro para as coordenadas do canto superior esquerdo e o segundo para o canto inferior direito. Ao realizar esse procedimento com todas as imagens, o código salva a nome da imagem juntamente com as duas coordenadas em um documento de texto, para ser comparado com as detecções dos algoritmos. O código pode ser visto no Apêndice A.

Além disso, foi feito outro *script* em Python para mostrar as caixas delimitadoras criadas e verificar se estão de acordo, a Figura 14 ilustra o rosto de um indivíduo com a *bounding box* feita pelo código anterior.

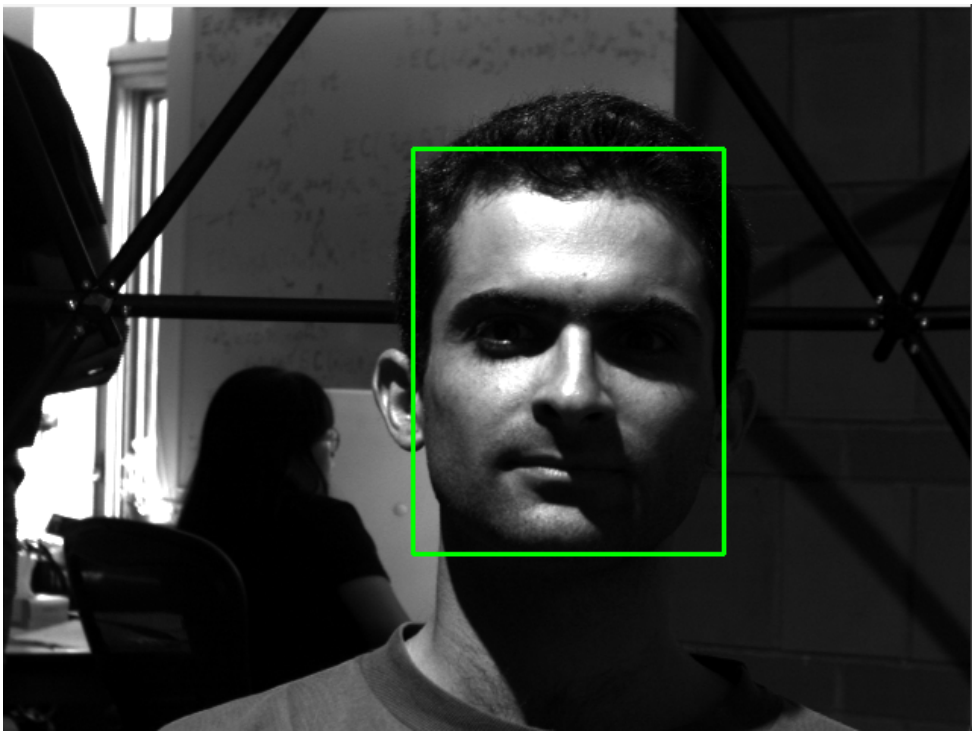
Esse processo ajudou na verificação e validação dos resultados obtidos pelos algoritmos, ao se ter uma base de comparação e facilitar nos cálculos de precisão, acurácia, verdadeiros e falsos tanto positivos quanto negativos.

3.3 ESCOLHA DOS ALGORITMOS

Em relação aos métodos, foi optado pelos seguintes:

OpenCV (Open Source Computer Vision Library) é uma biblioteca amplamente utilizada em visão computacional, devido à sua versatilidade como biblioteca de código aberto, oferecendo uma ampla gama de funções para processamento de imagem, detecção de objetos e análise de vídeo. Além disso, o OpenCV é conhecido por sua

Figura 14 – Demonstração da caixa delimitadora criada



Fonte: Adaptado pelo autor (2024)

facilidade de uso e é suportado por várias linguagens de programação, incluindo Python, C++ e Java.

No contexto da detecção de faces, o OpenCV oferece módulos específicos, como o cv2.CascadeClassifier, utilizado neste trabalho. Este módulo usa o algoritmo de Viola-Jones, que aplica o modelo Haar-Cascaded, para detectar faces em imagens estáticas. O OpenCV oferece diferentes versões desses modelos para detecção facial, incluindo a padrão (default) e versões específicas para faces frontais, como frontal face alt tree e frontal face alt 2, todas disponíveis na própria biblioteca do OpenCV (OpenCV, 2023).;

- Padrão (default): A versão padrão do modelo Haar-Cascades para detecção facial no OpenCV é adequada para detecção geral de faces em diferentes orientações e condições de iluminação, oferecendo um bom equilíbrio entre precisão e desempenho computacional.
- Frontal Face Alt Tree: Esta variante otimizada do modelo Haar-Cascades no OpenCV é especialmente projetada para detecção frontal de faces, concentrando-se principalmente na detecção de faces em perfil frontal e podendo ser mais rápida que a versão padrão em condições ideais de iluminação.
- Frontal Face Alt 2: Outra variação do modelo Haar-Cascades no OpenCV, esta é direcionada para melhorar a precisão em diferentes ângulos e rotações leves das faces. É especialmente útil em cenários onde as faces não estão perfeitamente

alinhadas, proporcionando resultados mais confiáveis em detecções de faces.

- YOLO (You Only Look Once): O YOLO é capaz de realizar detecção de objetos em tempo real de maneira robusta e veloz, tornando atraente para aplicativos que exigem baixa latência, além de conseguir detectar múltiplos objetos em uma única passagem pela imagem, sendo eficiente em termos de recursos, e possui equilíbrio entre velocidade e precisão na detecção de objetos, incluindo faces (Redmon *et. al*, 2016). O YOLO é baseado em redes neurais convolucionais para detecção não só de faces como também objetos, apesar de ser mais indicado para operações em tempo real, diferente deste trabalho. A versão do YOLO escolhida foi a mais recente, a versão 8 (YOLO v8) do ano de 2023, seu modelo foi pré-treinado com a base de dados COCO;
- DNN (Deep Neural Network): As redes neurais profundas, DNNs, são usadas em tarefas de detecção de objetos, incluindo faces, e são capazes de aprender representações complexas de dados. Alcançam desempenho superior em comparação com métodos tradicionais, quando treinadas em conjuntos de dados substanciais, além da compatibilidade com o OpenCV, que oferece suporte à integração de modelos de DNN treinados para detecção de objetos, o que permite a utilização de redes pré-treinadas para a detecção de faces (LeCun *et al.*, 2015). Assim como o nome supõe, DNN utiliza da deep learning para seu aprendizado de máquina, e consequentemente, na detecção facial, além disso, seu modelo foi pré-treinado com a base de dados LMDB;
- HOG: O HOG é especialmente útil em cenários onde a precisão é fundamental, na detecção de objetos, como faces. A biblioteca dlib utiliza o HOG para oferecer funcionalidades adicionais, como rastreamento de faces e identificação facial. Essas características fazem do dlib uma escolha apropriada em aplicações de análise de vídeo e rastreamento de rostos. A dlib é uma biblioteca de código aberto com uma licença amigável que permite seu uso em uma variedade de projetos, devido a sua simplicidade, suporte a linguagem python e C++, além de sua compatibilidade com aprendizado de máquina e visão computacional (dlib, 2023);
- MTCNN (Multi-task Cascaded Convolutional Networks): O MTCNN é um modelo projetado especificamente para a detecção de faces, é preciso na localização de faces em imagens, pode detectar várias faces em uma única imagem e utiliza uma hierarquia de redes para a detecção de faces, incluindo a detecção de pontos-chave do rosto (Zhang *et al.*, 2016). Nesse algoritmo, como o próprio nome já implica, usa redes neurais convolucionais multitarefa em cascata e seu modelo foi pré-treinado com a base de dados CelebA.

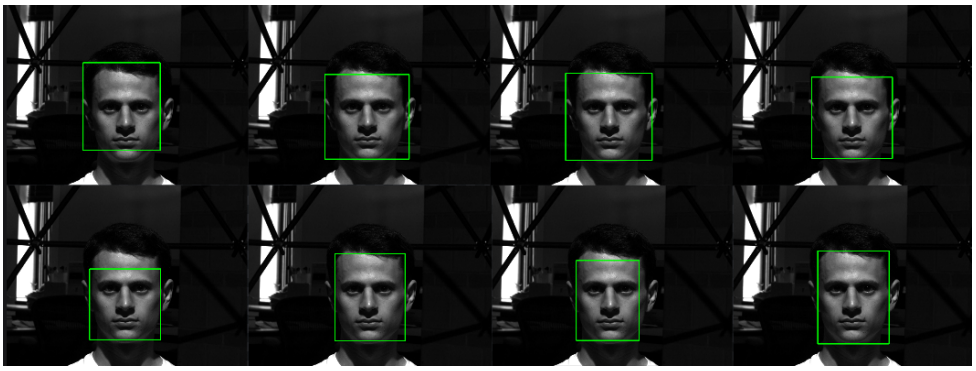
Todos os modelos mencionados acima são compatíveis com a linguagem de programação Python, para ter uma generalização com os códigos e devido à suas

bibliotecas prontas para análise e processamento de dados. Além disso, todos os modelos dos algoritmos citados já foram pré-treinados para realização deste trabalho.

A Figura 15 mostra, no campo superior esquerdo, a imagem de um indivíduo juntamente com a *bounding box* realizada manualmente por meio do script em Python, enquanto as demais se referem às detecções realizadas pelos diferentes métodos.

A título de exemplo, as coordenadas da caixa da anotação são: 203, 156, 405, 385, e as coordenadas obtidas pelo dlib com HOG foram: 221, 222, 407, 407, referente a imagem no canto inferior esquerdo. Assim, o processo para determinação do IOU começa identificando a área de interseção, que é delimitada pelos pontos de maior valor de cada par de coordenadas iniciais e os pontos de menor valor dos pares de coordenadas finais. Calcula-se a área dessa interseção multiplicando a diferença das coordenadas x e y obtidas. A união das duas áreas é então calculada somando as áreas das caixas e subtraindo a área de interseção. O IOU é finalmente obtido dividindo a área de interseção pela área de união, resultando em um valor que indica a precisão da sobreposição entre as duas caixas, sendo 0.5915 o valor resultante para as coordenadas fornecidas. Esse valor é maior que o limite adotado, 0.50, logo a detecção foi classificada como Verdadeiro Positivo.

Figura 15 – Demonstração das detecções feitas



Fonte: Autor (2024)

O processo de obtenção das coordenadas das detecções e cálculo do IOU foram feitos para todos os métodos e seus resultados foram salvos em arquivos de texto e também em matrizes de confusão.

4 RESULTADOS

Os resultados da análise da variação da iluminação utilizando os sete métodos diferentes são apresentados neste capítulo. Inicialmente, os resultados das detecções de cada método foram coletados separadamente e registrados em arquivos de texto correspondentes. Em seguida, foi realizada uma comparação com as coordenadas das *bounding boxes* das faces, obtidas manualmente, para calcular a *Intersection over Union* (IOU) para todos os métodos analisados.

Posteriormente, utilizando um *threshold* de 50% para determinar a validade das detecções, os resultados foram classificados em Verdadeiros Positivos (VP), Falsos Positivos (FP), Falsos Negativos (FN) e Verdadeiros Negativos (VN), o que permitiu a construção das matrizes de confusão. Para cada método, foram geradas seis matrizes de confusão, uma para cada grupo de grau de iluminação avaliado e uma geral, que considera todas as detecções do método em questão. Com base nessas matrizes, foi realizada uma avaliação quantitativa do desempenho de cada método em termos de precisão, *recall*, acurácia e F1-Score.

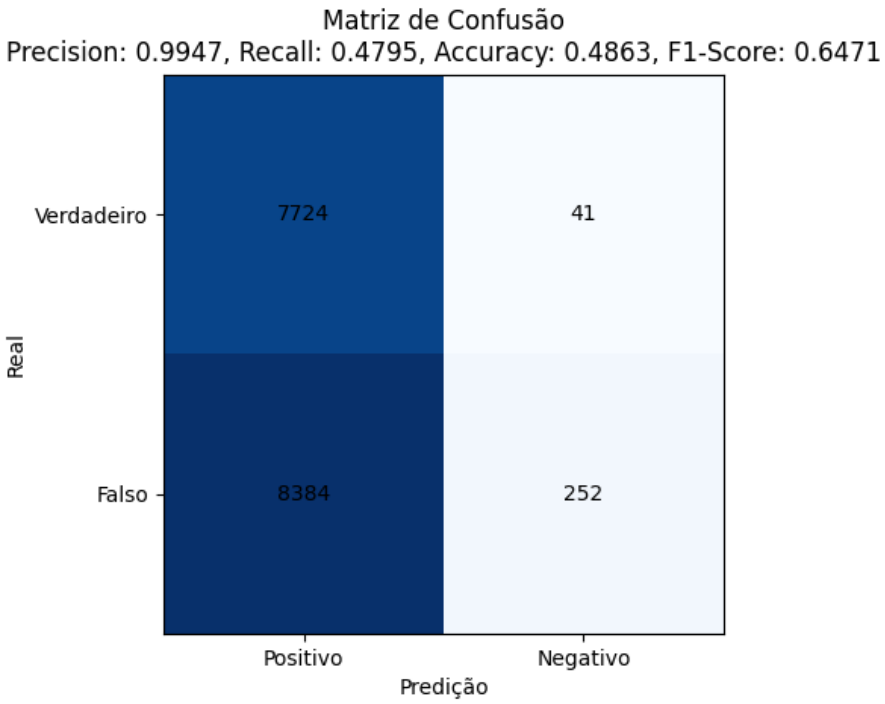
A Figura 16 mostra uma das matrizes de confusão obtidas no estudo, com a aplicação do método YOLO v8, de todas as imagens da base de dados analisadas. A partir dessa matriz, é possível extrair os dados das equações anteriores e comparar com os demais algoritmos. Tendo os valores numéricos de TP, TN, FP e FN, pode-se calcular as métricas de precisão, revocação, acurácia e F1-Score.

Pode-se perceber que conforme há maior quantidade de certo quadrante, a cor se torna mais intensa, representação comumente utilizada ao exibir matrizes de confusão, logo nota-se que há uma grande discrepância entre TP e FN, com 7724 e 8384, respectivamente, em relação a FP e TN, com 41 e 252, refletindo-se na alta precisão e valores de revocação e acurácia próximos.

A Figura 17 apresenta a comparação dessas quatro métricas para os sete métodos analisados. Observa-se que os valores de precisão são geralmente mais altos devido à baixa incidência de falsos positivos, em contraste com o *recall*, que considera os falsos negativos. Estes últimos tendem a ser mais numerosos devido à variação na iluminação, o que dificulta a detecção das faces. Para melhor visualização dos gráficos, os métodos estão abreviados, sendo Haar-Cascaded Default (HC-D), Haar-Cascaded FrontalFace Alt Tree (HC-FFAT), Haar-Cascaded FrontalFaceAlt 2 (HC-FFA2), dlib HOG (HOG) e demais seguem com a mesma sigla.

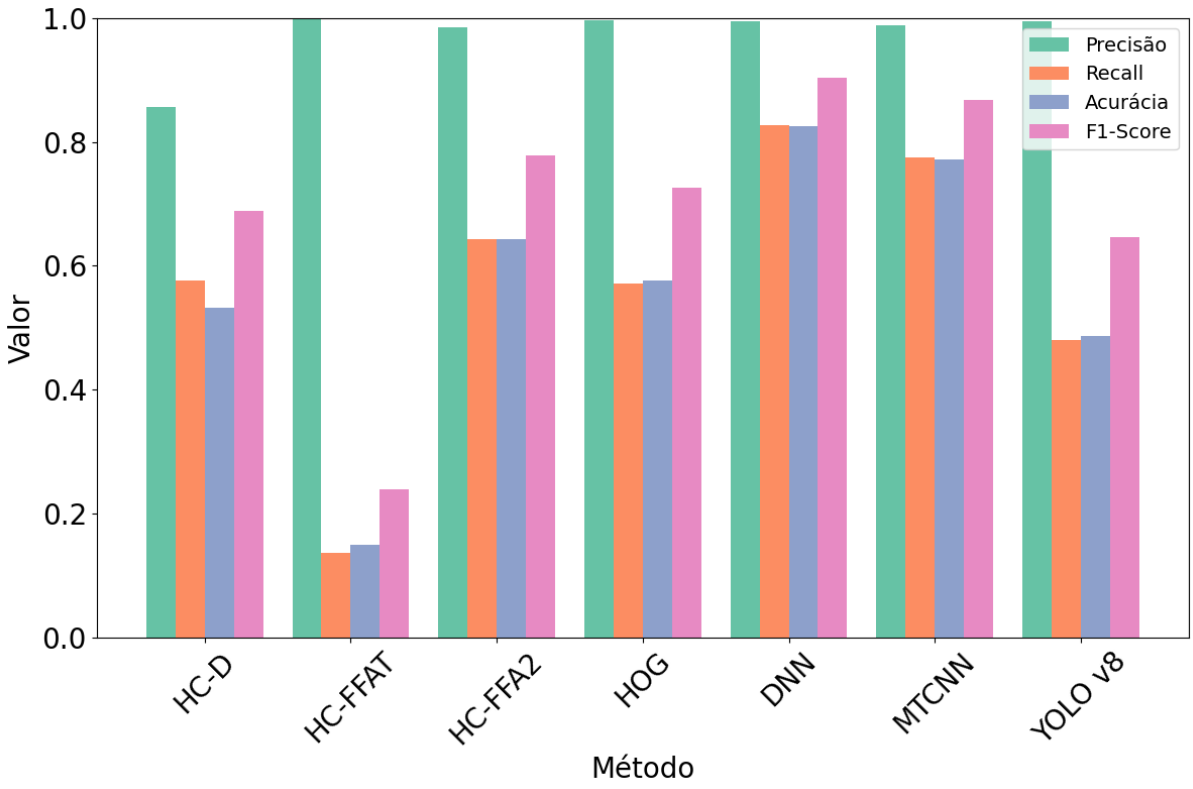
Já a acurácia teve seu valor próximo ao *recall* em decorrência do pequeno número de falsos positivos e verdadeiros negativos em relação aos verdadeiros positivos

Figura 16 – Matriz de Confusão YOLO v8



Fonte: Autor, 2024

Figura 17 – Comparação das métricas dos 7 métodos



Fonte: Autor (2024)

e falsos negativos. O primeiro, FP, é menor, pois, em geral, detectava as faces no local correto quando ocorria uma detecção. O segundo, VN, é pequeno, pois a base de dados possuía apenas 9 imagens do ambiente para cada indivíduo, que foram usadas para verdadeiros negativos. Isso resultou em 252, um número que não influencia tanto se comparado ao total de 16380 imagens. Vale ressaltar que todos os métodos não realizaram a detecção nesses casos, portanto o valor de VN foi igual a 252 para todos. No caso do F1-Score, o valor condiz com o apresentado na Figura 17 pelo fato do cálculo levar em consideração tanto a precisão, valor em geral maior, quanto *recall*, valor menor, como consequência, o F1-Score permaneceu como um valor intermediário entre os dois.

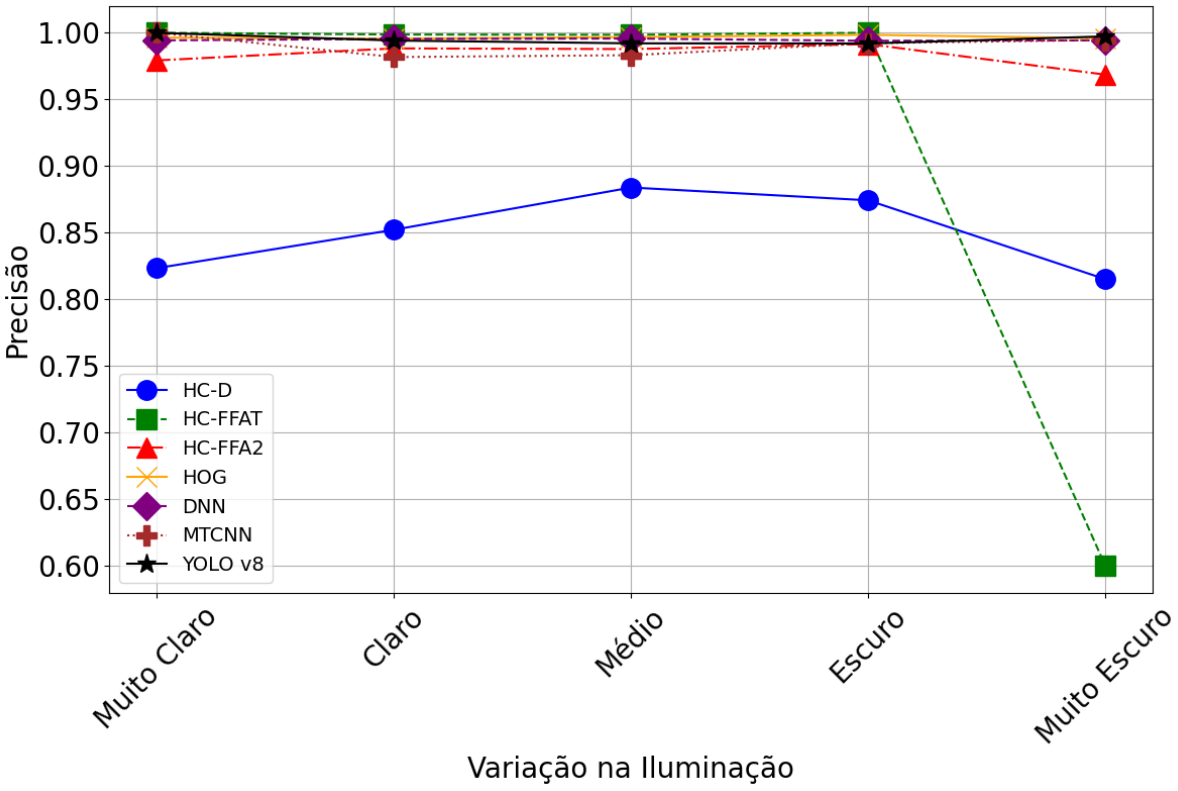
Em geral, os métodos MTCNN e DNN demonstraram maior robustez à variação da iluminação, mantendo um desempenho consistente em todas as condições testadas. Eles alcançaram os maiores valores de acurácia e F1-Score, embora a precisão não seja a mais alta em comparação ao dlib com HOG e HC-FFAT. Excluindo o método Haar-Cascaded com o modelo frontalface alt tree, que resultou no pior desempenho, apesar da sua maior precisão, e os dois melhores, os três métodos restantes, HC-D, HC-FFA2 e o YOLO v8, apresentaram resultados intermediários e próximos entre si. Os valores dos dados nesse gráfico podem ser visualizados no Apêndice B, na tabela 1, com quatro algarismos significativos. Os resultados gerais para cada método separado também podem ser vistos no Apêndice B, nas tabelas 2 a 8 para HC-D, HC-FFAT, HC-FFA2, HOG, DNN, MTCNN e YOLO, respectivamente

Como sugere a análise de Kaur e Sharma (2023), o MTCNN possuiu um desempenho superior se comparado ao método Haar e HOG considerando a revocação, acurácia e F1-Score para a detecção de faces com maiores restrições nas imagens, nesse caso, a variação na iluminação, apesar de, contrário aos resultados obtidos por Kaur e Sharma, o MTCNN obteve precisão menor que dois dos três métodos do Haar-Cascaded e também comparado ao HOG utilizando dlib, mesmo que no caso desse trabalho os rostos sejam apenas frontais, com oclusão devido a sombra.

A Figura 18 ilustra a comparação entre os sete diferentes métodos em relação a precisão, e como varia o comportamento nos cinco grupos de iluminação. É possível perceber que a precisão se mantém alta independente da variação na iluminação para a maior parte dos métodos, com exceção do método de HC-D, que não obteve um bom desempenho comparado aos demais em todos os grupos, possivelmente por ser um método mais simples em relação aos outros, e o HC-FFAT, que teve uma queda drástica na precisão no grupo muito escuro, passando de aproximadamente 99% nos demais para 60% no último grupo.

Os valores de precisão se mantiveram próximos de 100% em quase todos os métodos devido ao número significativamente maior de Verdadeiros Positivos em

Figura 18 – Comparação da evolução da Precisão



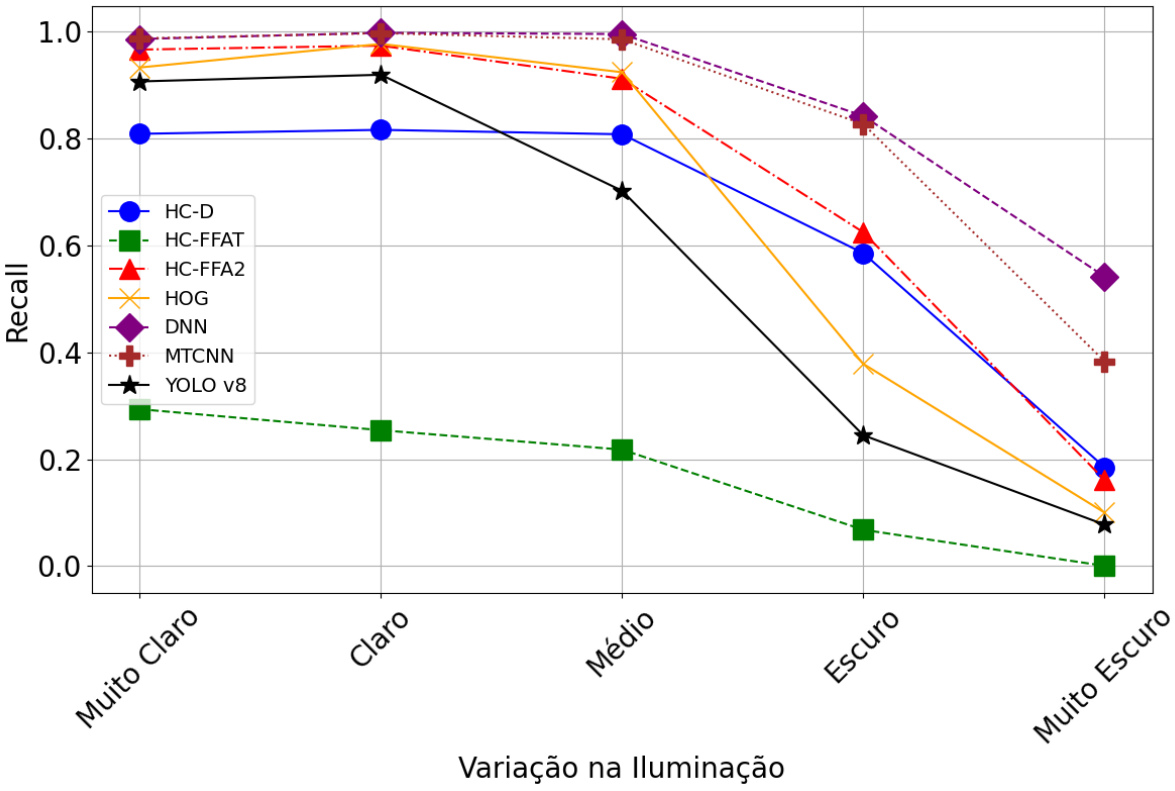
Fonte: Autor (2024)

comparação com os Falsos Positivos. Isso se deve ao fato de que a maioria dos algoritmos, ao detectar uma face, realizava a detecção corretamente. O cálculo da Interseção sobre União (IOU) da caixa delimitadora da detecção era, no mínimo, de 50% em relação à caixa delimitadora manualmente criada, que corresponde à localização real da face.

A Figura 19, de maneira similar, mostra a comparação do *recall* dos métodos e como ocorre sua evolução de acordo com a diminuição da luz. Nesse caso, é nítido como o valor do recall cai conforme a iluminação passa pelos diferentes grupos como era esperado, em especial, as transições entre Médio e Escuro e entre Escuro e Muito Escuro, onde há maior diferença. Vale destacar o melhor método na questão do recall, o DNN, se manteve superior aos demais em todos os casos, e o pior, o HC-FFAT, que teve um desempenho abaixo de 30% mesmo com maior iluminação.

A visível queda nos valores de recall é provocada pelo grande número de Falsos Negativos, o que também era algo esperado, já que suponha-se que os algoritmos detectem menos faces com o obscurecimento aumentando. Com relação aos métodos intermediários, obtiveram um desempenho semelhante na evolução do recall com a variação da luz, apesar de que o YOLO v8 teve pior desempenho que HC-D nos grupos Médio em diante, a precisão foi melhor em todos os grupos, como visto anteriormente

Figura 19 – Comparação da evolução do Recall



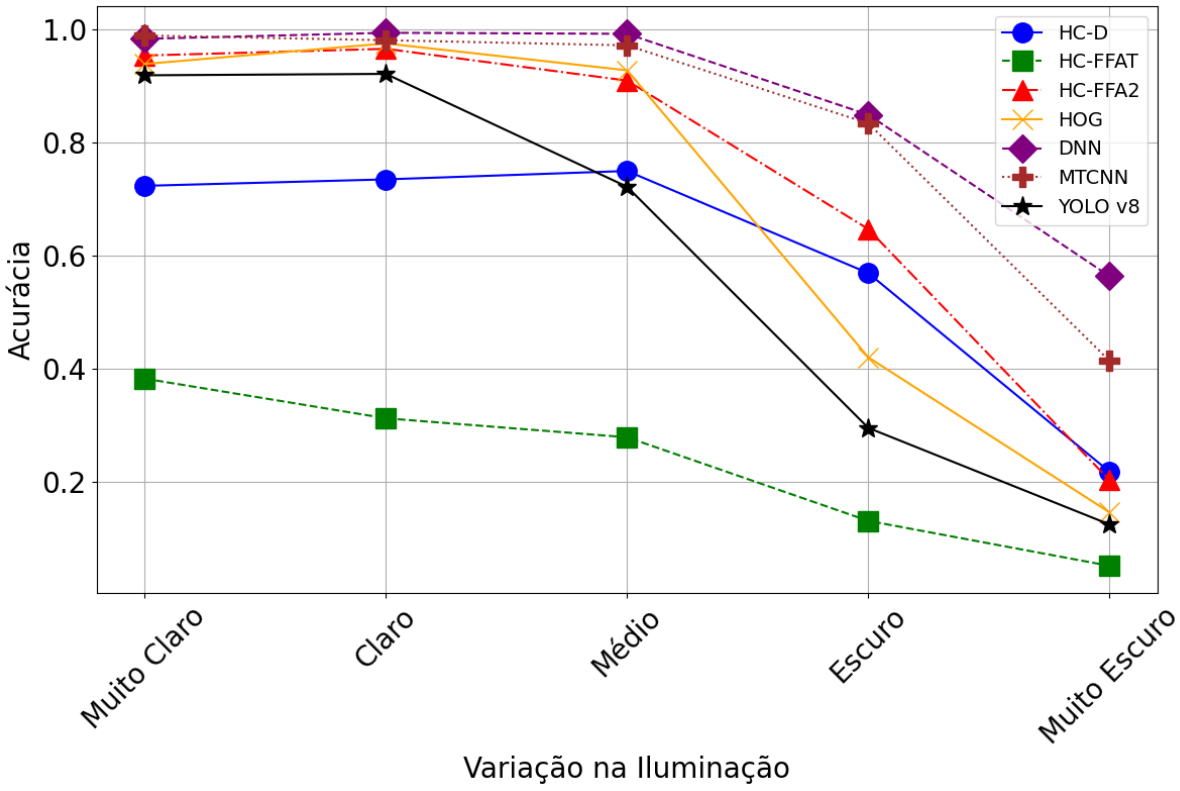
Fonte: Autor (2024)

na Figura 18.

As Figuras 20 e 21 expõem respectivamente, a acurácia e F1-Score e a evolução de seus números com os diferentes ângulos de iluminação comparando os sete métodos escolhidos. Para a acurácia, que leva em conta a proporção de todas as detecções corretas, utilizando tanto Verdadeiros Positivos e Verdadeiros Negativos, em relação ao total de previsões, somando Falsos Positivos e Falsos Negativos ao valor anterior. Em razão do pequeno número de Falsos Positivos, como já citado, e Verdadeiros Negativos, por haver menos imagens do grupo do ambiente, o cálculo da acurácia levou a um comportamento similar ao do recall na evolução dos grupos de iluminação. Entretanto, há um deslocamento no eixo y para baixo no caso do HC-D, devido à precisão menor, e para cima no caso de HC-FFAT, que possui valores maiores de acurácia por causa da precisão próxima de 1.

Já referente a F1-Score, como se trata de um valor intermediário relacionando precisão e recall, os números resultaram próximos ao recall, com deslocamento no eixo y para cima também devido a precisão que foi ao recall em todos os casos, "deslocou"os gráficos para cima. Pode-se notar que tanto para o F1-Score quanto a acurácia dos métodos YOLO v8 e Haar-Cascaded Default estão próximos no grupo Médio, em contraste com gráfico do recall, além disso o YOLO v8 obteve a precisão em

Figura 20 – Comparação da evolução da Acurácia



Fonte: Autor (2024)

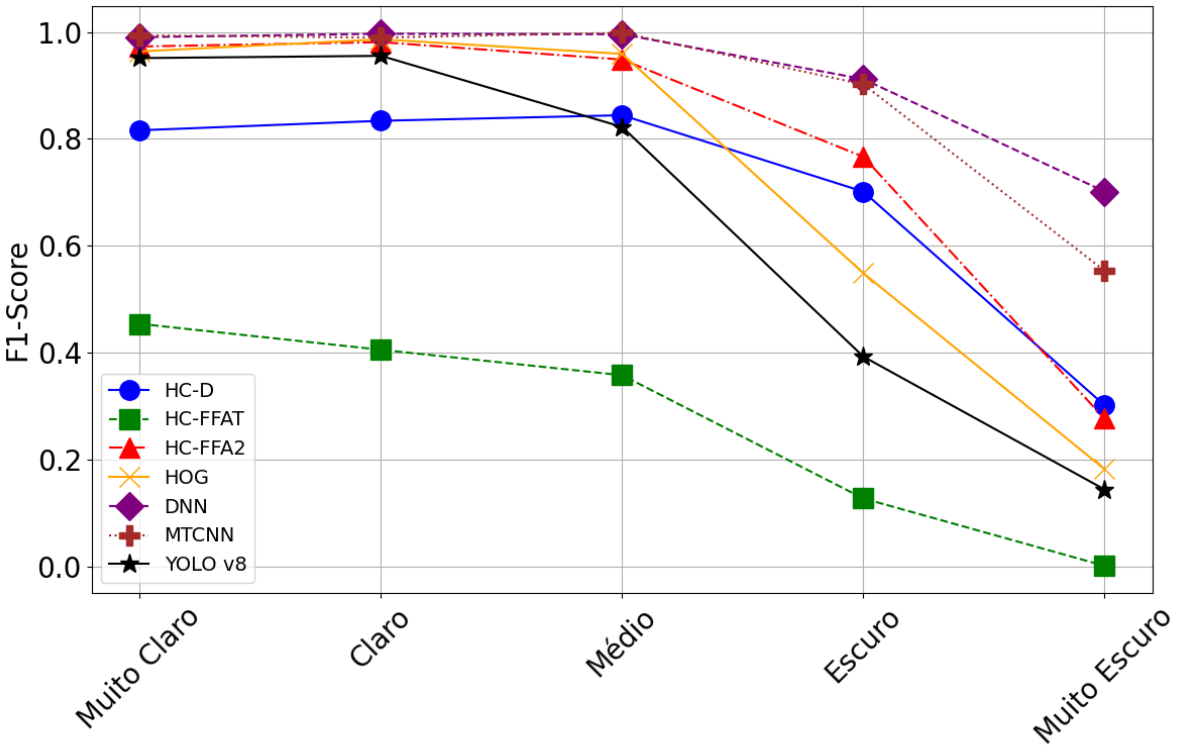
torno de 99,50% em todos os casos enquanto HC-D com 85% de precisão em média.

A Figura 22 ilustra a comparação dos boxplots para precisão, foi considerado os dados de precisão dos diferentes métodos com os 28 indivíduos e também de maneira correlata, a variação na iluminação. Pode-se perceber que os valores médios ficaram próximos de 1 para a maior parte, porém o Haar-Cascaded Default apresentou grande variação dentre os métodos analisados em todas os grupos de iluminação. Outro ponto de destaque é HC-FFAT, que obteve uma média de precisão nula para o grupo Muito Escuro, com apenas três outliers próximos de 1, fato que indica que a luminosidade foi determinante para detecção para este método.

Com relação aos outliers, é evidente que HC-D e HC-FFAT tiveram resultados similares independente do grupo com um indivíduo, resultando em uma média 0, embora HC-FFA2 tenha identificado a face para o mesmo indivíduo. Além disso, o YOLO v8 possui pior desempenho para precisão, tanto nos quartis quanto nos outliers, para os grupos Muito Claro e Claro, porém, contraintuitivamente, melhorou nos grupos com menor luminosidade.

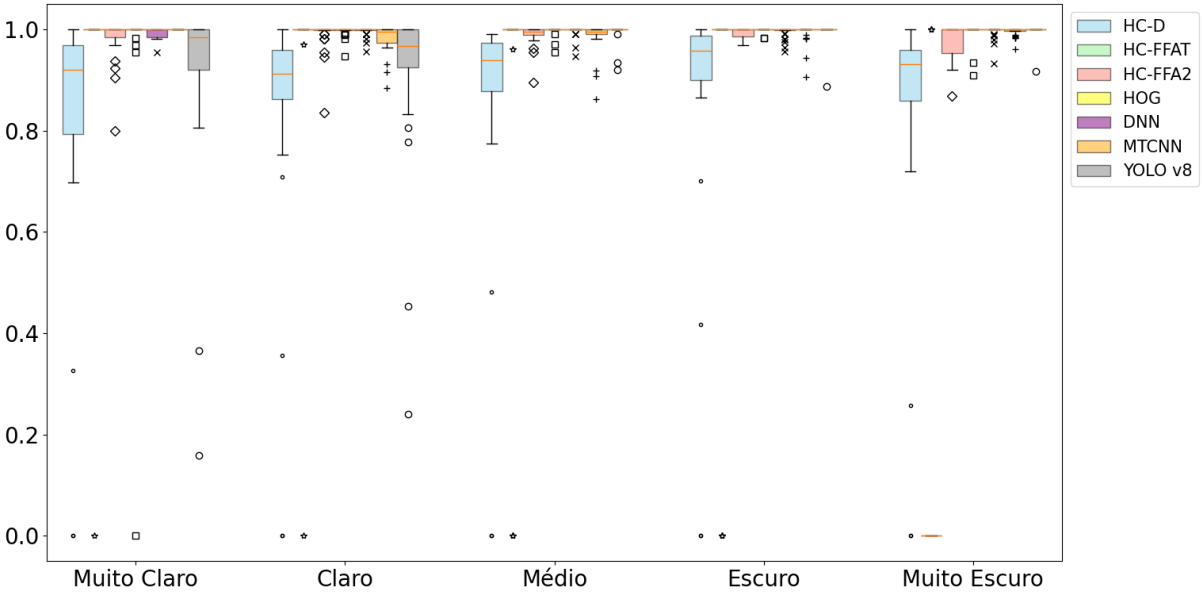
Já a Figura 23 mostra a comparação dos boxplots para recall, de maneira análoga à anterior, pode-se reparar a grande variação de desempenho em todos os métodos. Em especial para o YOLO v8, para os dois primeiros grupos, com mais

Figura 21 – Comparação da evolução da F1-Score



Fonte: Autor (2024)

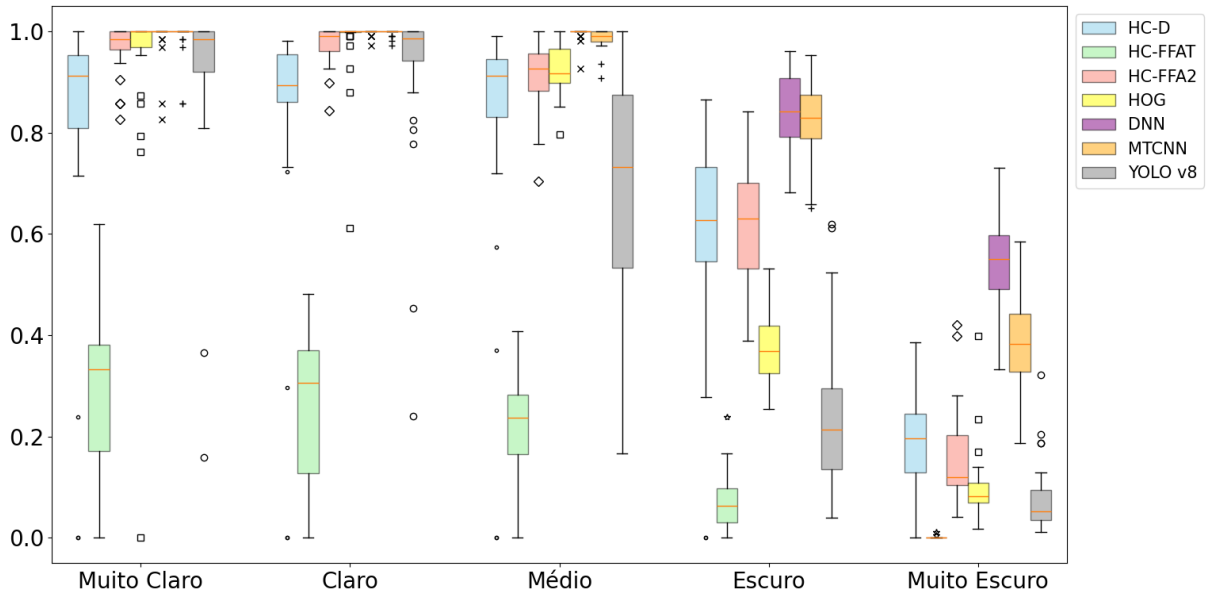
Figura 22 – Comparação de Boxplots de Precisão



Fonte: Autor (2024)

iluminação, possuiu valores maiores de recall, apesar dos outliers, no entanto, houve uma enorme distribuição de valores no grupo médio em relação aos demais métodos.

Figura 23 – Comparação de Boxplots de Recall



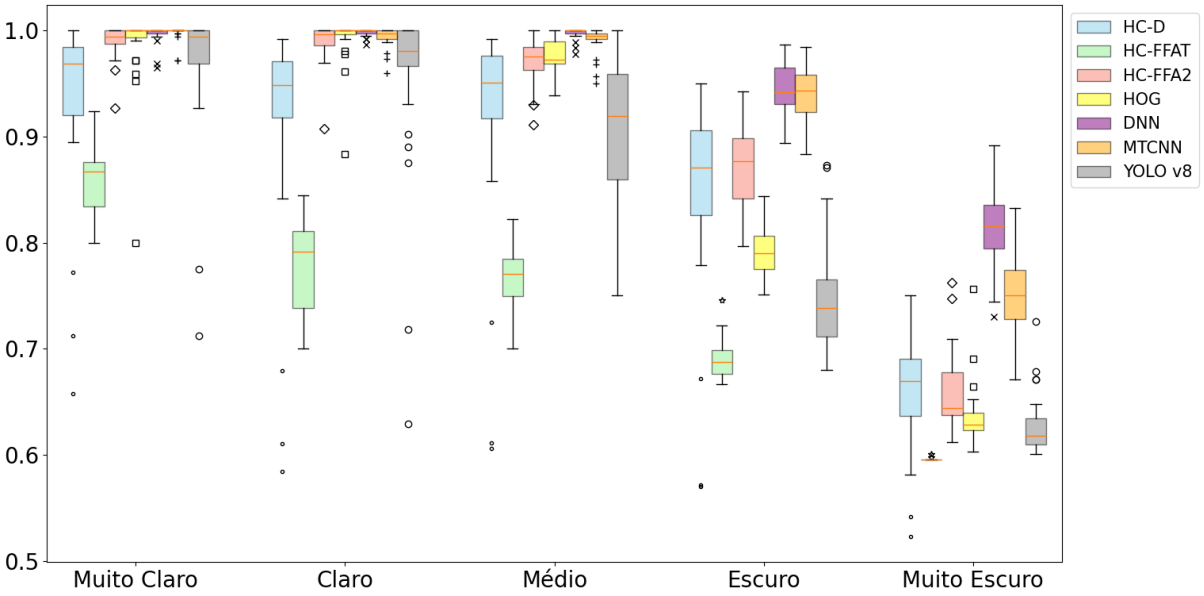
Fonte: Autor (2024)

Constata-se que para os grupos Muito Claro e Claro, excluindo HC-D e HC-FFAT, os desempenhos foram similares e, conforme a iluminação decai, a diferenciação entre eles aumenta. Também é possível notar que os métodos DNN e MTCNN possuíram maior desempenho em geral, assim como nos boxplots de precisão, enquanto HC-FFAT teve o pior desempenho para o recall.

As Figuras 24 e 25 seguem de modo semelhante aos boxplots de recall, mas considerando a acurácia e F1-Score, respectivamente. No caso da acurácia é visível que de igual modo, os valores médios para os quartis e outliers tiveram um acréscimo em todos os grupos e métodos. Adicionalmente, os boxplots tornaram-se mais achatados, com menor discrepância em relação à mediana.

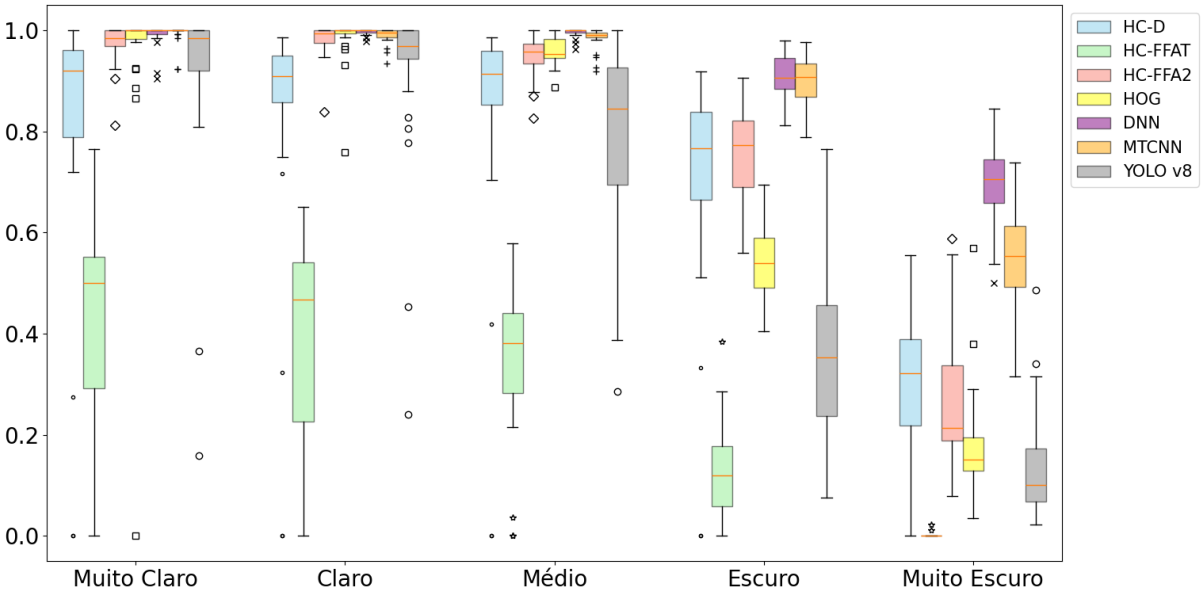
No que diz respeito a F1-Score, o que se observa é justamente o contrário do ocorrido com a acurácia, os boxplots tornaram-se mais largos e seus valores médios diminuíram, indicando que além de haver maior variabilidade média para todos os casos, a F1-Score registrou, em geral, resultados piores que a recall para o conjunto de grupos, comparativamente.

Figura 24 – Comparação de Boxplots de Acurácia



Fonte: Autor (2024)

Figura 25 – Comparação de Boxplots de F1-Score



Fonte: Autor (2024)

5 CONCLUSÕES

Esse trabalho investigou o desempenho de diferentes métodos de detecção de face sob variações de condições de iluminação. Através de uma base de dados com 5 grupos de iluminação diferentes e 28 indivíduos foram obtidas conclusões sobre a eficácia e robustez desses métodos. Os resultados demonstraram que os métodos baseados em deep learning, com CNNs, apresentaram um desempenho superior tanto em cenários de baixa iluminação quanto em geral, quando comparados aos métodos mais simples, como o Haar-Cascaded. Essa superioridade pode ser atribuída à capacidade das CNNs de aprender características mais discriminativas e adaptáveis às variações de iluminação.

Para realizar esse trabalho, utilizou-se metodologias e métricas comuns na literatura de Inteligência Artificial e dados previamente coletados para a base de dados. Aproveitou-se também as implementações de algoritmos já desenvolvidas, como as arquiteturas de CNNs com modelos pré-treinados e as bibliotecas para detecção de faces. Foi realizado a delimitação das *bounding boxes* para as faces, para então calcular a IOU com as detecções dos algoritmos para assim, gerar a matriz de confusão. Essa abordagem possibilitou focar na análise comparativa dos resultados sob diferentes condições de iluminação.

Como conclusão da comparação entre os métodos escolhidos, é possível notar que os métodos utilizando DNN e MTCNN tiveram um desempenho melhor, com 56,32% e 41,34% de acurácia, respectivamente, para a detecção de faces independente da iluminação, inclusive com menor variação nas médias, para as métricas analisadas. Porém, vale ressaltar a biblioteca dlib utilizando HOG que, apesar de ter um desempenho que deteriorou nos grupos Escuro e Muito Escuro, obteve resultados próximos aos melhores métodos, com métricas acima de 92%, mesmo não sendo baseado em Inteligência Artificial.

Finalmente, os resultados desse estudo indicam que as mudanças nas condições de iluminação são fatores que influenciam a eficiência de algoritmos de detecção de face, porém a base de dados utilizada nesse trabalho envolve apenas imagens em tons de cinza, com tamanhos e localização padronizados. Portanto, para pesquisas futuras, poderia ser explorado imagens com todos os tons de cores, com tamanhos diferentes, possuindo mais de uma face por imagem em locais diferentes. Além disso, os métodos para a detecção das faces podem ser diferentes, por exemplo, designando apenas métodos utilizando CNN ou, comparando mais métodos.

REFERÊNCIAS

ABATE, A. *et al*. 2d and 3d face recognition: A survey. **Pattern Recognition Letters**, v. 28, n. 14, p. 1885–1906, oct. 2007.

ADINI, Y.; MOSES, Y.; ULLMAN, S. Face recognition: The problem of compensating for changes in illumination direction. **IEEE Transactions on pattern analysis and machine intelligence**, v. 19, n. 7, p. 721–732, 1997.

BEHLING, A. **Reconhecimento de emoções em vídeo utilizando redes neurais artificiais**. Trabalho de Conclusão de Curso (Curso de Ciências da Computação) — Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis, 2019.

BöHM, S. M. **Análise de Performance de um Algoritmo de Reconhecimento Facial por Visão Computacional Aplicado a Sistemas Embarcados**. Trabalho de Conclusão de Curso (Curso de Engenharia da Computação) — Campus Araranguá, Universidade Federal de Santa Catarina, Araranguá, 2021.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: **2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)**. [S.l.: s.n.], 2005. v. 1, p. 886–893 vol. 1.

DLIB. **Biblioteca dlib**. Disponível em: http://dlib.net. Acesso em: 26 out. 2023.

FORSYTH, D. A.; PONCE, J. **Computer vision: a modern approach**. [S.l.]: prentice hall professional technical reference, 2002.

GAMAGE, C.; SENEVIRATNE, L. Development of a learning algorithm for facial recognition under varying illumination. *In*: Proceedings of the 7th INTERNACIONAL CONFERENCE ON INFORMATION AND AUTOMATION FOR SUSTAINABILITY. v. 1, n. 7, p. 1–6, 2014. Disponível em: https://ieeexplore.ieee.org/document/7069626. Acesso em: 05 set. 2023.

GONZALEZ, R. C. **Digital image processing**. [S.l.]: Pearson education india, 2009.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. [S.l.]: MIT press, 2016.

JAGADISWARY, D.; APPASAMI, G.; RAJESH, S. Eye features normalization and face emotion detection for human face recognition. *In*: Proceedings of the 2011 INTERNACIONAL CONFERENCE ON ELETRONICS, COMMUNICATION AND COMPUTING TECHNOLOGIES. v. 1, n. 3, p. 64–68, 2011. Disponível em: https://ieeexplore.ieee.org/document/6077071. Acesso em: 05 set. 2023.

JAIN, A.; HONG, L.; PANKANTI, S. Biometric identification. **Communications of the ACM**, New York, v. 43, n. 2, p. 90–98, 2000.

JAIN, A.; ROSS, A.; PRABHAKAR, S. An introduction to biometric recognition. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 14, n. 1, p. 4–20, 2004.

KAUR, S.; SHARMA, D. Comparative study of face detection using cascaded haar, hog and mtcnn algorithms. In: **2023 3rd International Conference on Advancement in Electronics Communication Engineering (AECE)**. [S.l.: s.n.], 2023. p. 536–541.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LORENA, A. C.; DE CARVALHO, A. C. Uma introdução às support vector machines. **Revista de Informática Teórica e Aplicada**, v. 14, n. 2, p. 43–67, 2007.

MEDIUM. **Matriz de Confusão**. 2021. Disponível em: https://medium.com/@bernardolago/matriz-de-confusÃčo-7c0e36468323. Acesso em: 4 jul. 2024.

MULLER, A. C.; GUIDO, S. **Introduction to machine learning with Python**. Gravenstein Highway North, Sebastopol: O'Reilly, 2016.

OPENCV. **Open Source Computer Vision Library**. Disponível em: https://opencv.org. Acesso em: 26 out. 2023.

PANKANTI, S.; BOLLE, R. M.; JAIN, A. Biometrics: The future of identification [guest eeditors' introduction]. **Computer**, v. 33, n. 2, p. 46–49, 2000.

RAUBER, T. W. Redes neurais artificiais. **Universidade Federal do Espírito Santo**, v. 29, 2005.

REDMON, J. et al. You only look once: Unified, real-time object detection. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 779–788.

RUSSELL, S.; NORVIG, P. **Inteligência artificial**. 3. ed. Rio de Janeiro: Elsevier Editora Ltda, 2013.

SANTANA, L. M. Queiroz de; ROCHA, F.; SANTOS, T. Uma análise do processo reconhecimento facial. **Cadernos de Graduação: Ciências Exatas e Tecnológicas**, v. 2, p. 49–58, 10 2014.

SOUSA, M. D. d. A. et al. Análise comparativa entre os principais algoritmos de detecção facial: Haar cascade, hog, cnn, yolo e deepface. **OPEN SCIENCE RESEARCH V**, Editora Científica Digital, v. 5, n. 1, p. 439–454, 2022.

SZELISKI, R. **Computer vision: algorithms and applications**. [S.l.]: Springer Nature, 2022.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: **Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001**. [S.l.: s.n.], 2001. v. 1, p. I–I.

YALE. The extended Yale face database B. v. 1, n. 1, 2001. Disponível em: http://vision.ucsd.edu/ iskwak/ExtYaleDatabase/ExtYaleB.html. Acesso em: 25 out. 2023.

ZHANG, K. et al. Joint face detection and alignment using multitask cascaded convolutional networks. **IEEE signal processing letters**, v. 23, n. 10, p. 1499–1503, 2016.

ZHANG, L.; WANG, H.; CHEN, Z. A multi-task cascaded algorithm with optimized convolution neural network for face detection. In: **2021 Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS)**. [S.l.: s.n.], 2021. p. 242–245.

ZHAO, W. *et al*. Face recognition: A literature survey. **ACM computing surveys**, New York, v. 35, n. 4, p. 399–458, 2003.

APÊNDICE A - CÓDIGO

```
import cv2
import os

# Lista para armazenar todas as coordenadas da bounding box
all_bounding_boxes = []

# Variaveis para armazenar informacoes da bounding box
start_point = None
bounding_box = []

# Funcao para desenhar a bounding box
def draw_bbox(event, x, y, flags, param):
    global start_point, bounding_box

    if event == cv2.EVENT_LBUTTONDOWN:
        if start_point is None:
            start_point = (x, y)
        else:
            end_point = (x, y)
            bounding_box = [start_point[0], start_point[1], end_point
                [0], end_point[1]]
            all_bounding_boxes.append({'image': current_image, 'bbox':
                bounding_box})
            cv2.rectangle(img, start_point, end_point, (0, 255, 0), 2)
            cv2.imshow('image', img)
            start_point = None

# Diretorio onde estao as imagens de cada grupo
directory = "C:\\Users\\hitsu\\OneDrive\\Imagens\\angulo_0_12"

# Loop pelas imagens no diret rio , de B11 a B39
for filename in os.listdir(directory):
    if filename.startswith("yaleB11") and filename.endswith(('.jpg', '.
        png', '.jpeg')):
        current_image = filename
        image_path = os.path.join(directory, filename)
        img = cv2.imread(image_path)
        cv2.namedWindow('image')
        cv2.setMouseCallback('image', draw_bbox)

        while True:
            cv2.imshow('image', img)
```

```
            k = cv2.waitKey(1) & 0xFF
            if k == 27:  # Tecla ESC para sair
                break

    cv2.destroyAllWindows()

# Salvando todas as coordenadas em um unico arquivo de texto
output_file = "C:\\Users\\hitsu\\OneDrive\\Imagens\\bounding_0_12.txt"
with open(output_file, 'w') as file:
    for entry in all_bounding_boxes:
        file.write(f"{entry['image']},{','.join(map(str, entry['bbox']))
            }\n")
```

APÊNDICE B - TABELAS

Tabela 1 – Resultados

Método	Precisão	Recall	Acurácia	F1-Score
Haar-Cascaded default	85,68%	57,62%	53,22%	68,90%
Haar-Cascaded frontalface_alt_tree	99,82%	13,58%	14,90%	23,90%
Haar-Cascaded frontalface_alt2	98,59%	64,39%	64,35%	77,90%
dlib HOG	99,65%	57,11%	57,66%	72,61%
DNN	99,48%	82,69%	82,60%	90,31%
MTCNN	98,89%	77,46%	77,15%	86,87%
YOLO v8	99,47%	47,95%	48,63%	64,71%

Tabela 2 – Resultados Haar-Cascaded default

Haar-Cascaded default	Precisão	Recall	Acurácia	F1-Score
Muito Claro	82,34%	80,90%	72,31%	81,61%
Claro	85,22%	81,64%	73,45%	83,39%
Médio	88,39%	80,81%	74,94%	84,43%
Escuro	87,43%	58,56%	56,86%	70,14%
Muito Escuro	81,51%	18,51%	21,72%	30,17%

Tabela 3 – Resultados Haar-Cascaded frontal face alt tree

Haar-Cascaded frontalface_alt_tree	Precisão	Recall	Acurácia	F1-Score
Muito Claro	100,00%	29,37%	38,19%	45,40%
Claro	99,87%	25,44%	31,17%	40,55%
Médio	99,85%	21,80%	27,81%	35,79%
Escuro	100,00%	6,80%	13,02%	12,74%
Muito Escuro	60,00%	0,06%	5,06%	0,13%

Tabela 4 – Resultados Haar-Cascaded frontal face alt 2

Haar-Cascaded frontalface_alt2	Precisão	Recall	Acurácia	F1-Score
Muito Claro	97,93%	96,66%	95,37%	97,29%
Claro	98,83%	97,39%	96,56%	98,10%
Médio	98,78%	91,20%	90,93%	94,84%
Escuro	99,15%	62,47%	64,65%	76,65%
Muito Escuro	96,86%	16,13%	20,23%	27,66%

Tabela 5 – Resultados dlib HOG

dlib HOG	Precisão	Recall	Acurácia	F1-Score
Muito Claro	99,64%	93,31%	93,87%	96,37%
Claro	99,56%	97,72%	97,51%	98,63%
Médio	99,68%	92,42%	92,75%	95,91%
Escuro	99,85%	37,84%	41,96%	54,88%
Muito Escuro	99,59%	10,03%	14,52%	18,22%

Tabela 6 – Resultados DNN

DNN	Precisão	Recall	Acurácia	F1-Score
Muito Claro	99,43%	98,64%	98,32%	99,03%
Claro	99,54%	99,80%	99,39%	99,67%
Médio	99,57%	99,57%	99,21%	99,57%
Escuro	99,40%	84,24%	84,89%	91,19%
Muito Escuro	99,43%	54,20%	56,32%	70,16%

Tabela 7 – Resultados MTCNN

MTCNN	Precisão	Recall	Acurácia	F1-Score
Muito Claro	100,00%	98,75%	98,91%	99,37%
Claro	98,18%	99,74%	98,08%	98,95%
Médio	98,32%	98,58%	97,17%	99,85%
Escuro	99,22%	82,71%	83,36%	90,21%
Muito Escuro	99,46%	38,34%	41,34%	55,35%

Tabela 8 – Resultados YOLO v8

YOLO v8	Precisão	Recall	Acurácia	F1-Score
Muito Claro	100,00%	90,70%	91,87%	95,12%
Claro	99,42%	91,95%	92,12%	95,54%
Médio	99,21%	70,23%	72,14%	82,24%
Escuro	99,20%	24,49%	29,47%	39,28%
Muito Escuro	99,73%	7,77%	12,38%	14,42%