

UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA - CCET  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA - DEE

FELIPE KUTAIT

AVALIAÇÃO DOS MÉTODOS FACENET E LBPH DE RECONHECIMENTO FACIAL  
ATRAVÉS DE UMA DETECÇÃO HÍBRIDA DE YOLO COM MTCNN

São Carlos  
2025

FELIPE KUTAIT

AVALIAÇÃO DOS MÉTODOS FACENET E LBPH DE RECONHECIMENTO FACIAL  
ATRAVÉS DE UMA DETECÇÃO HÍBRIDA DE YOLO COM MTCNN

Trabalho de conclusão de curso apresentado ao  
Departamento de Engenharia Elétrica da  
Universidade Federal de São Carlos, para obtenção  
do título de bacharel em Engenharia Elétrica.

Orientador: Prof.Dr. Celso Ap.de França

São Carlos  
2025

## **DEDICATÓRIA**

Dedico este trabalho a toda minha família que sempre mantiveram sua confiança em mim e me deram forças para continuar

## **AGRADECIMENTOS**

Primeiramente ao meu professor orientador Celso Aparecido de França por todos os seus conselhos e auxílios durante este trabalho.

Aos meus pais, Edson e Claudia por sempre acreditarem em mim, me apoiando e incentivando ao longo de todo este tempo.

À minha avó Aparecida por todo carinho, amor e por sempre cuidar de mim.

À Rafaela por me trazer amor e alegria e me dar coragem para enfrentar meus desafios.



## RESUMO

Nos dias atuais há uma crescente busca por segurança e comodidade o que reflete tanto no nosso dia quanto em nosso ambiente. Seja desbloquear o celular ou entrar em uma área de acesso restrito, as leituras biométricas estão cada vez mais presentes no dia a dia e espera-se que a tecnologia evolua para tornar estes métodos confiáveis, práticos e não invasivos. Isso se adequa bem quando se entra na área de visão computacional e se depara com o reconhecimento facial, uma área que atua na identificação de pessoas sejam em fotos, vídeos ou até mesmo em tempo real e que tem um alto potencial em segurança, autenticação e monitoramento. Entretanto, esta área também contém uma série de desafios. Iluminação, posição, ângulo, acessórios que podem esconder partes do rosto e até mesmo envelhecimento podem apresentar dificuldades para um sistema de reconhecimento facial. Novos métodos modernos e sofisticados baseados em aprendizagem de máquinas vem se mostrando promissores para enfrentar estes obstáculos se provando ser sistemas robustos e eficientes. Dentre estes sistemas, o modelo *Local Binary Patterns Histograms* transforma imagens em histogramas e os compara para identificar pessoas. Outro método muito promissor utiliza redes neurais convolucionais para analisar características importantes e gerar representações altamente discriminativas, o FaceNet. Avaliando estes modelos, o estudo fará testes com ambos em diferentes situações e trazendo resultados e mostrando como a tecnologia avançou para os métodos de redes neurais convolucionais se tornando cada vez mais precisos e eficazes para lidarem com os diversos obstáculos do reconhecimento facial.

**Palavras-chave:** Redes neurais convolucionais; Reconhecimento facial; Detecção facial híbrida; FaceNet, LBPH

## **ABSTRACT**

There is currently a growing demand for safety and comfort, which is reflected in our daily lives and in our surroundings. Whether it is to unlock a cell phone or enter a restricted access area, biometric readings are increasingly present in our daily lives, and the expectation is that technology will evolve to make these methods reliable, practical and non-invasive. This fits well when we enter the area of computer vision and come across facial recognition, an area that works on identifying people, whether in photos, videos or even in real time, and which has great potential in security, authentication and monitoring. However, this area also contains a series of challenges. Lighting, position, angle, accessories that can hide parts of the face and even aging can present difficulties for a facial recognition system. To overcome these obstacles, a series of promising studies have brought modern and sophisticated methods based on machine learning that have proven to be robust and efficient systems. Among these systems, there is a more classical model known as Local Binary Pattern Histograms, which uses the comparison of pixels with their edges to transform images into histograms and compare them to identify people. Another very promising method uses convolutional neural networks to analyze the most important features of a face across all its neurons and generate highly discriminative representations, as is the case with FaceNet. To evaluate these models, this study will test both in different situations and provide results showing how technology has advanced and how these recent convolutional neural network methods are proving increasingly effective in dealing with the various obstacles of facial recognition.

**Keywords:** Convolutional neural network; Facial recognition; Hybrid facial detection; FaceNet; LBPH

## LISTA DE ILUSTRAÇÕES

|                                                                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1 — Esquemático de redes neurais artificiais .....                                                                                                | 14 |
| Figura 2 — Rede neural convolucional .....                                                                                                               | 15 |
| Figura 3 — Etapas para o reconhecimento de face .....                                                                                                    | 16 |
| Figura 4 — Arquitetura YOLO .....                                                                                                                        | 17 |
| Figura 5 — Funcionamento da camada Maxpool.....                                                                                                          | 18 |
| Figura 6 — Funcionamento YOLO .....                                                                                                                      | 19 |
| Figura 7 — Etapas MTCNN.....                                                                                                                             | 20 |
| Figura 8 — Pré-processamento pós detecção facial e antes da rede de reconhecimento.....                                                                  | 21 |
| Figura 9 — Comparação de pixels e nova matriz binária.....                                                                                               | 22 |
| Figura 10 — Extração de histogramas da imagem .....                                                                                                      | 23 |
| Figura 11 — Banco de faces Yalefaces .....                                                                                                               | 25 |
| Figura 12 — Banco de faces AT&T.....                                                                                                                     | 26 |
| Figura 13 — Banco de faces LFW .....                                                                                                                     | 26 |
| Quadro 1 — Bibliotecas utilizadas.....                                                                                                                   | 27 |
| Figura 14 — Diagrama das etapas do projeto.....                                                                                                          | 28 |
| Figura 15 — Modelo YOLO Simplificado.....                                                                                                                | 30 |
| Figura 16 — Teste da detecção de faces pelo YOLO .....                                                                                                   | 31 |
| Figura 17 — Teste Yolo em múltiplas faces.....                                                                                                           | 32 |
| Figura 18 — Estrutura das redes neurais MTCNN.....                                                                                                       | 33 |
| Figura 19 — Teste da detecção por MTCNN .....                                                                                                            | 34 |
| Figura 20 — Teste MTCNN em múltiplas faces .....                                                                                                         | 35 |
| Figura 21 — Processo de detecção de faces híbrida.....                                                                                                   | 35 |
| Figura 22 — Teste da detecção híbrida dos modelos YOLO e MTCNN .....                                                                                     | 36 |
| Figura 23 — Teste da detecção híbrida em múltiplas faces.....                                                                                            | 37 |
| Figura 24 — Processo de divisão de imagens no banco de dados entre treinamento e reconhecimento.....                                                     | 39 |
| Figura 25 — Processo de rotulação das imagens de treinamento .....                                                                                       | 40 |
| Figura 26 — Saída do sistema de reconhecimento LBPH .....                                                                                                | 40 |
| Figura 27 — Imagem reconhecida pelo sistema LBPH.....                                                                                                    | 41 |
| Figura 28 — Dados da imagem identificada pelo método FaceNet .....                                                                                       | 42 |
| Figura 29 — Imagem reconhecida pelo modelo FaceNet .....                                                                                                 | 43 |
| Figura 30 — Avaliação dos modelos LBPH e FaceNet para o banco AT&T .....                                                                                 | 46 |
| Figura 31 — Resultados dos modelos de reconhecimento facial para o banco de dados LFW.....                                                               | 47 |
| Tabela 1 — Resultados dos modelos de reconhecimento .....                                                                                                | 47 |
| Tabela 2 — Comparação entre o modelo desenvolvido por este estudo e o desenvolvido pelo artigo de Reddy, Ramesh, Praveen, Surekha e Sharma (2023). ..... | 48 |



## **LISTA DE ABREVIATURAS E SIGLAS**

|       |                                            |
|-------|--------------------------------------------|
| CNN   | Redes neurais convolucionais               |
| LBPH  | Local Binary Patterns Histograms           |
| LFW   | Labelled Faces in the Wild                 |
| MLP   | Multilayer perceptron                      |
| MTCNN | Multi-Task Cascaded Convolutional Networks |
| ORL   | Our Database of Faces                      |
| RNA   | Redes neurais artificiais                  |
| YOLO  | You Only Look Once                         |
| IDE   | Ambiente de desenvolvimento integrado      |

## LISTA DE SÍMBOLOS

|          |                               |
|----------|-------------------------------|
| C        | Grau de confiabilidade        |
| d        | Distância                     |
| $x_i^a$  | <i>Embeddings</i> da âncora   |
| $x_i^p$  | <i>Embeddings</i> positiva    |
| $x_i^n$  | <i>Embeddings</i> negativa    |
| $\alpha$ | Margem do <i>Triplet Loss</i> |
| VP       | Verdadeiro positivo           |
| VN       | Verdadeiro negativo           |
| FP       | Falso positivo                |
| FN       | Falso negativo                |

## SUMÁRIO

|       |                                                             |    |
|-------|-------------------------------------------------------------|----|
| 1     | <b>INTRODUÇÃO</b>                                           | 11 |
| 1.1   | CONTEXTUALIZAÇÃO                                            | 11 |
| 1.2   | OBJETIVOS                                                   | 11 |
| 1.3   | JUSTIFICATIVA                                               | 12 |
| 1.4   | ESTRUTURA DO TRABALHO                                       | 12 |
| 2     | <b>FUNDAMENTAÇÃO TEÓRICA E REVISÃO DE LITERATURA</b>        | 13 |
| 2.1   | <i>MACHINE LEARNING, DEEP LEARNING E REDES NEURAIS</i>      | 13 |
| 2.2   | RECONHECIMENTO FACIAL                                       | 15 |
| 2.3   | DETECÇÃO FACIAL                                             | 16 |
| 2.3.1 | YOLO ( <i>You Only Look Once</i> )                          | 17 |
| 2.3.2 | MTCNN ( <i>Multi-task Cascaded Convolutional Networks</i> ) | 19 |
| 2.4   | PRÉ-PROCESSAMENTO                                           | 21 |
| 2.5   | LBPH ( <i>LOCAL BINARY PATTERNS HISTOGRAMS</i> )            | 21 |
| 2.6   | FACENET                                                     | 23 |
| 2.7   | MÉTRICAS DE AVALIAÇÃO                                       | 24 |
| 2.8   | BANCOS DE DADOS DE FACES                                    | 25 |
| 3     | <b>DESENVOLVIMENTO DO TRABALHO</b>                          | 27 |
| 3.1   | DETECÇÃO                                                    | 29 |
| 3.1.1 | YOLO                                                        | 29 |
| 3.1.2 | MTCNN                                                       | 32 |
| 3.1.3 | DETECÇÃO HÍBRIDA                                            | 35 |
| 3.2   | PRÉ-PROCESSAMENTO                                           | 37 |
| 3.3   | RECONHECIMENTO FACIAL                                       | 38 |
| 3.3.1 | LBPH ( <i>Local Binary Patterns Histograms</i> )            | 39 |
| 3.3.2 | FACENET                                                     | 42 |
| 3.4   | AVALIAÇÃO                                                   | 44 |
| 4     | <b>RESULTADOS</b>                                           | 45 |
| 5     | <b>CONSIDERAÇÕES FINAIS</b>                                 | 50 |
|       | <b>REFERÊNCIAS</b>                                          | 52 |

# 1 INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

Dentro da área de visão computacional, o reconhecimento facial aborda tanto a identificação de rostos, seja em imagens ou vídeos, e em seguida os classifica em um determinado contexto. A análise feita através de pontos-chaves de uma face têm sido amplamente inserida no cotidiano como uma maneira moderna de autenticação biométrica (Goodfellow, Bengio & Courville, 2016).

Desde controle de acesso até os sistemas de segurança pública, o reconhecimento facial vem estando cada vez mais presente no dia a dia e junto a ele, diversos desafios vêm sendo enfrentados. Iluminação, expressões faciais, e até mesmo envelhecimento devem ser levados em consideração quando se está desenvolvendo um sistema de reconhecimento facial, fora outros fatores como o próprio posicionamento da face ou acessórios cobrindo partes do rosto que podem prejudicar a leitura das imagens (Piva Junior et al., 2023).

Enfrentando todas as possíveis adversidades encontradas, diversos métodos vêm sendo criados e evoluindo conforme o tempo, como é o caso de modelos clássicos como o *Fisherface* e o *Eigen Faces* que deram seus lugares para modelos mais modernos baseados em redes neurais convolucionais (Piva Junior et al., 2023).

Embora seja uma tecnologia promissora, ela vem levantando um grande debate em que há o contraste da segurança e privacidade, onde há um potencial de elevar cada vez mais a segurança seja pública, seja de sistemas de autenticação e em contrapartida surgem questões éticas relacionadas à vigilância e o uso de dados. Portanto, o levantamento de estudos sobre esta área impulsiona sistemas cada vez mais precisos e incentiva debates que são cruciais para garantir que esta tecnologia seja cada vez mais benéfica para a sociedade (Piva Junior et al., 2023).

## 1.2 OBJETIVOS

O objetivo desta pesquisa foi desenvolver e avaliar sistemas de reconhecimento facial que através de uma abordagem mais clássica, o *Local Binary Patterns Histograms*, ou LBPH, fazendo uso da comparação de pixels vizinhos e os

transformando em histogramas, e uma abordagem mais moderna que utiliza de redes neurais convolucionais, o FaceNet que gera vetores de características e as compara para identificação, analisando principalmente precisão, robustez e eficiência, além de observar a evolução dos modelos até o momento.

Para chegar neste ponto, serão implementados dois modelos de detecção facial com o intuito de avaliar o desempenho deles separadamente e em conjunto de forma híbrida. Desta maneira a detecção deverá atuar da melhor maneira para que o reconhecimento facial não seja afetado.

Outro ponto crucial para a pesquisa será a aprendizagem dos modelos que passarão por mais de um tipo de banco dados para avaliar seus desempenhos em problemas mais “fáceis” e mais complexos.

### 1.3 JUSTIFICATIVA

A autenticação biométrica através do reconhecimento facial tem se provado uma forma eficiente e não invasiva de gerir proteção e controle de acesso a diversos ambientes, entretanto, ainda é uma tecnologia que sofre com diversos fatores, o que abre margem para que seu desenvolvimento e evolução sejam cada vez mais cobiçados pela sociedade. Pesquisas neste tema remetem não só a enfrentar estes fatores, mas também sugerem explorar sua aplicação nas mais diversas e criativas situações.

### 1.4 ESTRUTURA DO TRABALHO

Este primeiro capítulo introduz o assunto estudado, abordando sua importância e descrevendo inicialmente sendo mais bem explorados ao decorrer do projeto. O capítulo 2 se aprofundará no embasamento teórico necessário para a pesquisa de acordo com as referências bibliográficas. No terceiro capítulo, será apresentado o desenvolvimento dos modelos implementados. O capítulo 4 mostrará os resultados obtidos das avaliações dos modelos. O quinto e último capítulo são as considerações finais junto a conclusão da pesquisa.

## 2 FUNDAMENTAÇÃO TEÓRICA E REVISÃO DE LITERATURA

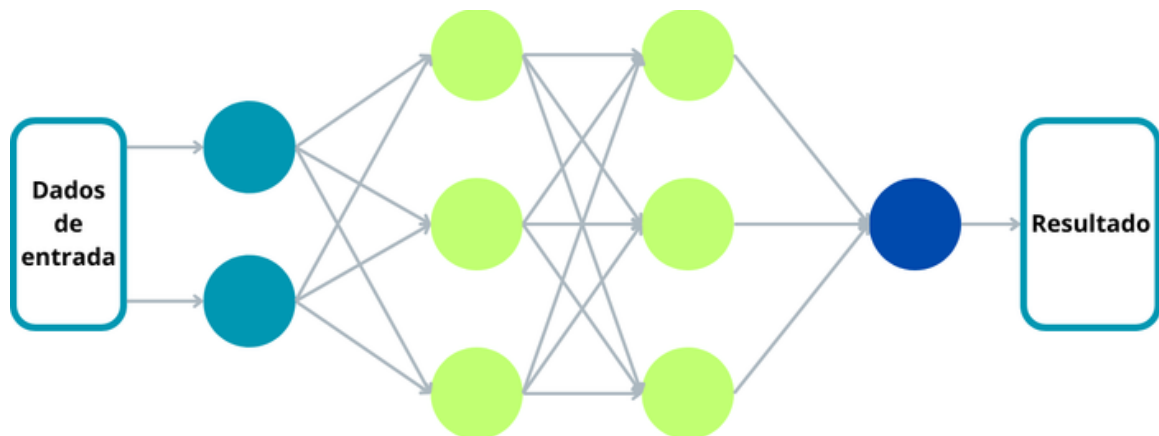
Este capítulo apresenta conceitos iniciais de *machine learning*, *deep learning* e redes neurais que são a base para os modelos de detecção *You Only Look Once* (YOLO) e *Multi-task Cascaded Convolutional Networks* (MTCNN) e para o modelo de reconhecimento facial FaceNet também apresentados ao decorrer do capítulo. Além dos modelos baseados em redes neurais, o capítulo também descreve os conceitos e etapas do processo de reconhecimento facial e apresenta um modelo clássico de reconhecimento que não utiliza de redes neurais, o LBPH.

### 2.1 MACHINE LEARNING, DEEP LEARNING E REDES NEURAIS

Atualmente há uma gigantesca quantidade de dados circulando por todo o mundo e detectar e analisar estes dados acabou se tornando uma tarefa quase impossível para a capacidade humana. Então para solucionar este problema, uma nova área surgiu, o machine learning, que se propõe a desenvolver modelos de identificação de padrões e predição de dados de maneira automática (Murphy, 2012).

Diversos métodos de aprendizado de máquina foram desenvolvidos, mas as redes neurais artificiais, ou RNAs, se sobressaíram por serem um sistema de neurônios inspirado no próprio cérebro humano. Seu funcionamento é composto por camadas de neurônios artificiais interconectados que processam dados para gerar a saída desejada. A Figura 1 mostra esta estrutura, onde inicialmente dados são inseridos no sistema, eles passam pelas camadas de entrada em que os neurônios de entrada representados pelos círculos azuis recebem os dados e os encaminham para as camadas ocultas onde neurônios (verdes) processam as informações para extrair padrões e as envia para a camada de saída (azul escuro) que gera o resultado final.

Figura 1 — Esquemático de redes neurais artificiais



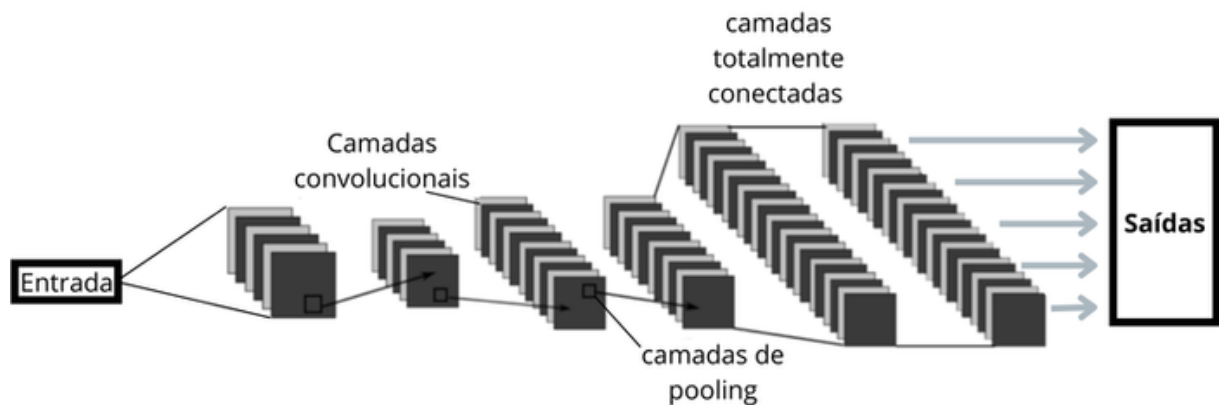
Fonte: Adaptado de Goodfellow, Bengio e Courville (2016).

Entretanto, com problemas cada vez mais complexos e a evolução da tecnologia fornecendo cada vez mais poder computacional a essas redes, elas passaram a ser cada vez mais complexas e profundas dando origem a um ramo do *machine learning*, o aprendizado profundo ou *deep learning*. (Goodfellow, Bengio & Courville, 2016).

Este campo consiste na utilização de redes neurais profundas com múltiplas camadas de neurônios reduzindo a necessidade de operações manuais e permitindo extrair representações hierárquicas complexas de reconhecimento de imagens (Murphy, 2012).

Uma das redes neurais mais reconhecidas do *deep learning* é a rede neural convolucional, ou CNN, formada por *Multilayer perceptron* (MLP) e se adequa muito bem para análise de imagens e outros sinais 2d. Além das camadas de neurônios, que neste caso são camadas totalmente conectadas, ou seja, um neurônio envia as suas descobertas relevantes para todos os neurônios das camadas seguintes, também existem outros dois tipos de camadas, as camadas convolucionais que aplicam filtros sobre os dados de entradas para descobrir padrões e características, e a camada de pooling que reduz a dimensionalidade de entrada, compactando informações extraídas pela convolução e tornando a rede mais eficiente (Goodfellow, Bengio & Courville, 2016). A Figura 2 mostra a estrutura de uma rede neural convolucional onde estão representadas camadas convolucionais e camadas de pooling intercaladas no início seguidas de camadas totalmente conectadas.

Figura 2 — Rede neural convolucional



Fonte: Adaptado de Rodrigues (2018).

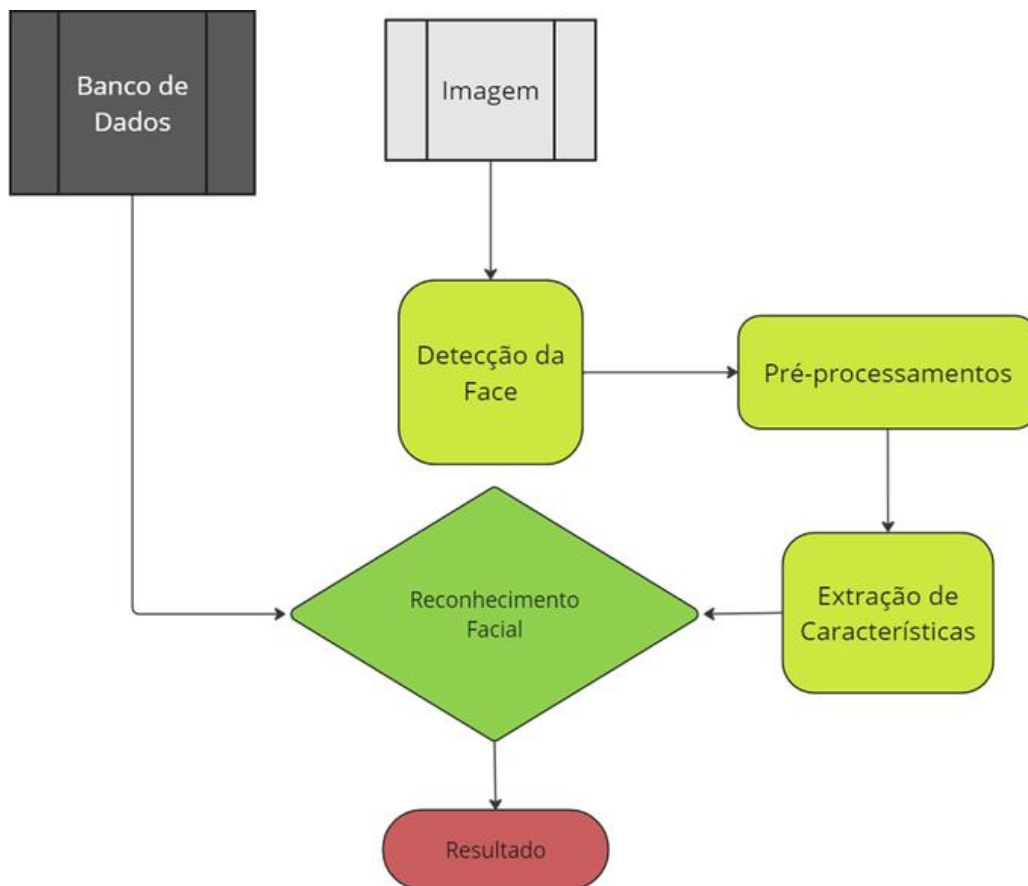
## 2.2 RECONHECIMENTO FACIAL

O reconhecimento facial pode ser entendido como uma das respostas mais naturais e instintivas do ser humano. Através das informações coletadas pelos olhos, ocorre o processamento de pontos característicos do rosto como formato, olhos, boca e outros dados que nos leva a reconhecer pessoas que já conhecemos e integrar rostos novos nesta forma de “banco de dados cerebral” para futuras interações sociais. Esta habilidade prática do cérebro humano despertou interesse em uma área pesquisas que tentam copiar e implementar este processo dentro de máquinas, o reconhecimento facial.

Para simular o reconhecimento facial, foram desenvolvidos séries de algoritmos que tendem a seguir determinados passos como detecção, tratamento e identificação (Piva Junior et al., 2023). Estes passos podem ser vistos na Figura 3 a seguir e cada uma das etapas é descrita ao decorrer do capítulo.



Figura 3 — Etapas para o reconhecimento de face



Fonte: adaptado de Piva Junior *et al.* (2023).

### 2.3 DETECÇÃO FACIAL

O processo inicial ao obter a imagem é a detecção facial. Nesta etapa os classificadores analisam e identificam a localização das faces presentes na imagem através de técnicas que enfrentam diversos problemas como a posição da câmera, a posição da pessoa, as expressões faciais ou presença de fatores como óculos, gorro, barba e outros objetos que podem obscurecer ou até mesmo ocultar parte da face. Além destes, a própria iluminação do ambiente tem bastante impacto na captação da face (Lopes, 2008).

Para contornar estes aspectos prejudiciais à detecção muitos estudos vêm utilizando de métodos híbridos. Estes métodos combinam diferentes técnicas de processamento de imagens e *machine learnings* com o fim de se complementarem aumentando assim a robustez do sistema sem que tenham grandes perdas de

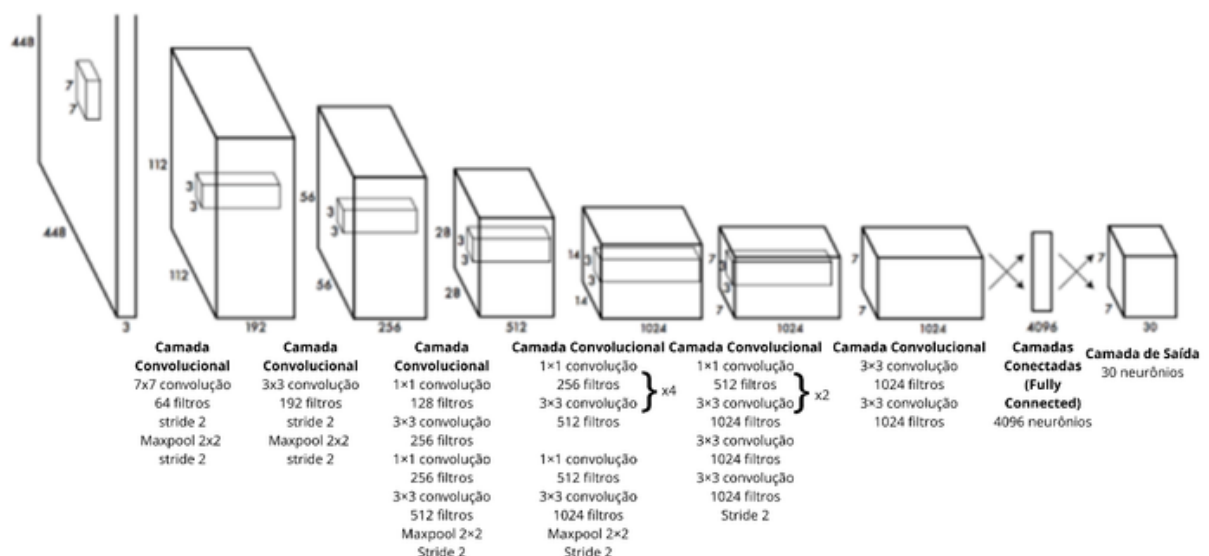
performance. Uma combinação que se aproveita de tanto de robustez quanto velocidade de processamento é o YOLO (*You Only Look Once*) e o MTCNN (*Multi-Task Cascaded Convolutional Networks*), dois métodos abordados a seguir.

### 2.3.1 YOLO (*You Only Look Once*)

Dentre os muitos algoritmos de detecção facial que existem nos dias de hoje, o *You Only Look Once*, ou YOLO, vem ganhando muito destaque por sua velocidade e precisão na detecção e é comumente utilizado em detecções por vídeos. (Redmon et al., 2016)

O conceito por trás do algoritmo YOLO está na divisão da imagem em uma matriz  $S \times S$  de regiões e cada uma dessas regiões será analisada em caixas delimitadas  $B$ , também conhecidas como *bounding boxes*. (Redmon et al., 2016) Estas *bounding boxes* recebem um grau de confiança  $C$ , onde este valor é a probabilidade de existir um objeto na grade, variando de 0 a 1, sendo 0 caso não haja objetos e 1 a maior chance de existir um objeto dentro da *bounding box* (Bonela, 2021). Todos estes elementos formam a arquitetura da rede neural convolucional (CNN) do modelo YOLO como pode ser visto na Figura 4:

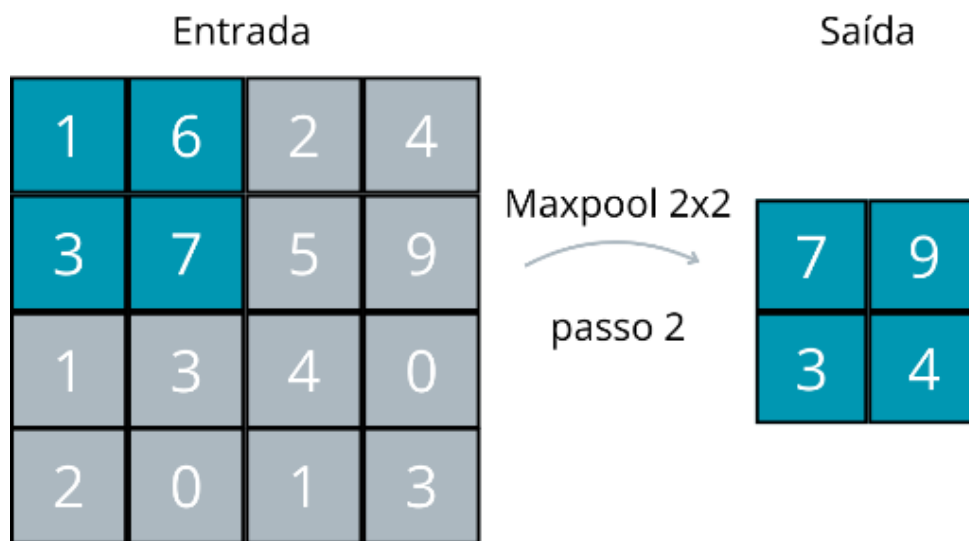
Figura 4 — Arquitetura YOLO



Fonte: Adaptada de Redmon et al. (2016).

A Figura 4 mostra o processo da coleta de uma imagem 448x448 com 3 canais que passa por camadas convolucionais, inicialmente com uma camada de 7x7 com 64 filtros e com uma camada de *Maxpool* 2x2 de passo de 2, que reduz as informações da imagem para 112x112. A camada de *Maxpool* utilizada seleciona uma região da imagem de 2x2 e guarda o maior valor desta região, reduzindo os 4 *pixels* em 1 e segue para a próxima região, este processo pode ser visto na Figura 5. O deslocamento desta matriz se dá pelo passo, no caso do modelo YOLO, ele é 2, ou seja, a matriz se desloca 2 pixels da região anterior. Este processo de camada convolucional e *Maxpool* se repete sendo que em algumas etapas só é aplicada a camada de convolução, isso se deve pela camada convolucional extrair as características da amostra enquanto a *Maxpool* reduz amostragem o que passa a não ser necessário nas etapas finais.

Figura 5 — Funcionamento da camada Maxpool



Fonte: Adaptado de Zhang *et al.* (2023).

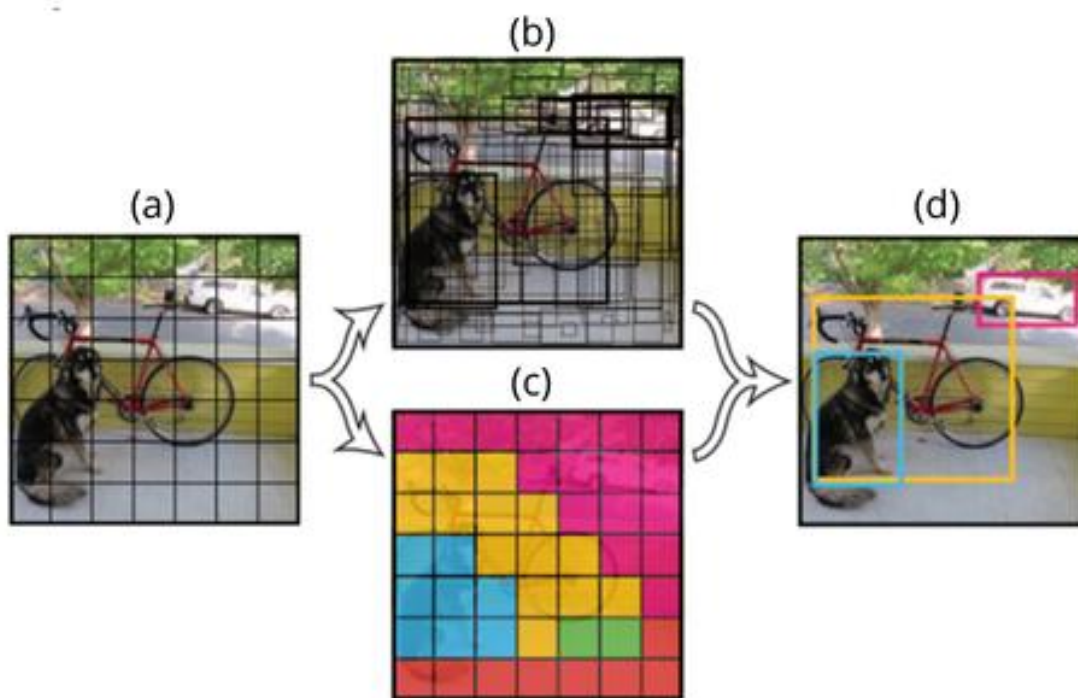
A Figura 5 mostra a seleção da matriz 2x2 da camada *Maxpool* representada pelos valores 1, 6, 3 e 7 e registra o maior valor na matriz resultante. Então este processo se repete percorrendo toda a matriz tendo um passo 2, ou seja, a próxima análise será na seção 2, 4, 5 e 9, resultando no valor 9.

A detecção Yolo é considerada uma das mais rápidas, podendo detectar objetos e pessoas em tempo real com uma latência de 25ms e ao observar a imagem

como um todo, consegue compreender alguns contextos que até alguns algoritmos de ponta tem mais dificuldade (Bonela, 2021).

A Figura 6 exemplifica o processo de identificação do algoritmo, onde uma imagem  $S \times S$  é dividida em grades (a) e para célula são previstas *bounding boxes* e seus graus de confiança  $C$  (b) que representa a certeza do objeto. Em paralelo à predição de *bounding boxes* ocorre o mapeamento de classe (c) que indica quais objetos tem maior probabilidade de estar presentes em cada célula. Ao mesclar os resultados de previsão de *bounding boxes* com o mapeamento de classes, o Yolo consegue detectar os diferentes itens da foto (d).

Figura 6 — Funcionamento YOLO

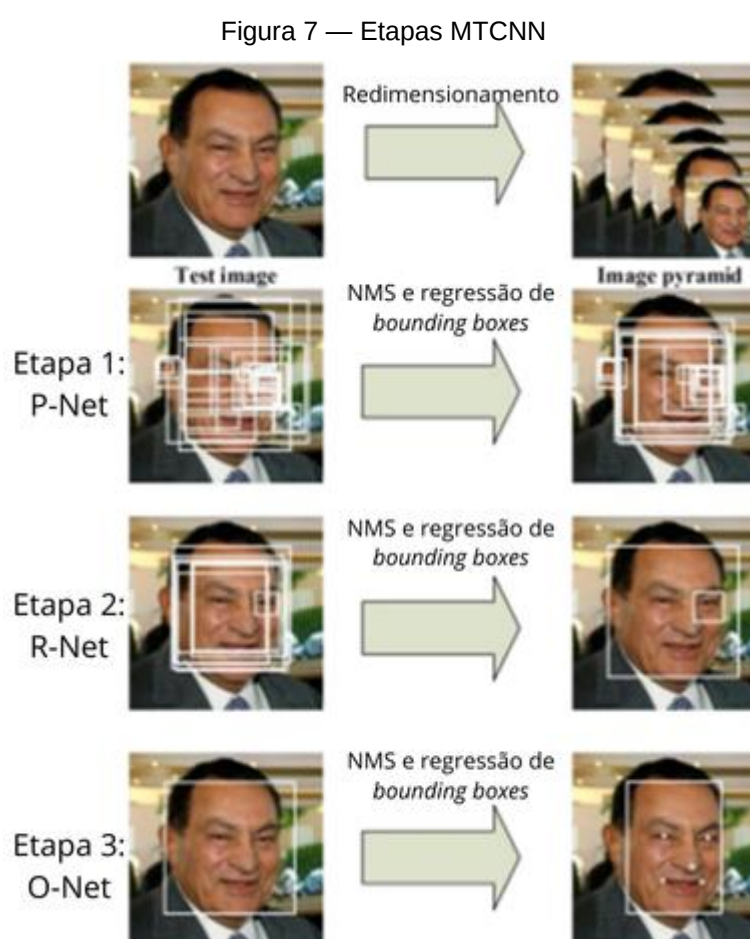


Fonte: adaptada de Redmon *et al.* (2016).

### 2.3.2 MTCNN (Multi-task Cascaded Convolutional Networks)

Outro modelo de detecção facial que utiliza de redes neurais é o MTCNN ou Redes Neurais Convolucionais em Cascata Multitarefa. Este modelo é baseado em aprendizado profundo que se refinam cada vez mais à medida que as camadas de redes neurais avançam, tornando-as mais complexas e precisas. (Pereira Junior, Teixeira & Homem, 2021).

Ao todo o modelo contém 3 redes neurais cada um responsável por um estágio da detecção. Inicialmente a P-Net ou rede de proposta identifica possíveis faces e cria *bounding boxes* na imagem. Já o estágio 2 conhecido com R-Net ou camada de refinamento é responsável por ajustar as *bounding boxes* do estágio anterior e eliminar falsos positivos. Por fim, a O-Net ou rede de saída, onde ocorre o último refinamento e é determinada a posição final da *bounding box*. É nesta camada também que são marcados 5 pontos faciais previstos entre olhos boca e nariz (Zhang et al., 2016). Estas etapas podem ser vistas na Figura 7:



Fonte: Adaptado de Zhang, Zhang e Li (2016).

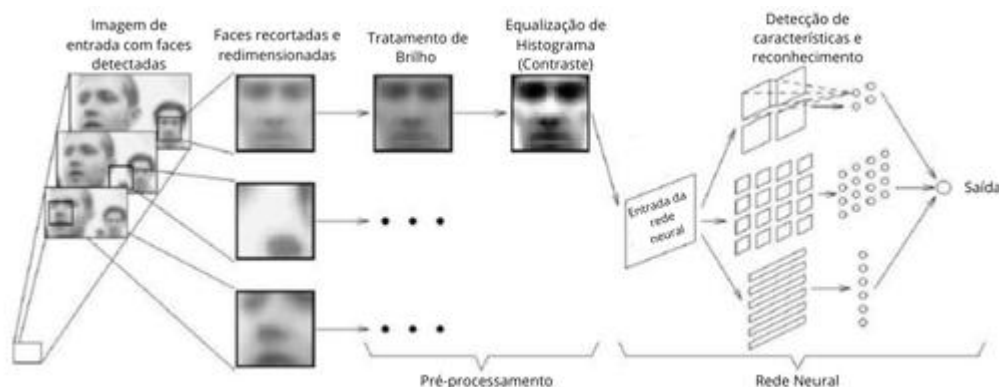
A detecção da face antes de outros processos facilita as etapas seguintes e diminui a carga necessária dos algoritmos de reconhecimento, onde uma vez a ocorre a detecção o algoritmo passará a focar apenas nos pixels delimitados pela face.

## 2.4 PRÉ-PROCESSAMENTO

O redimensionamento da imagem faz parte do pré-processamento, etapa onde vários tratamentos são feitos na imagem. Outros tratamentos comuns são: o alinhamento facial utilizando dos pontos faciais, redução de ruídos em imagens de menor qualidade, tratamentos de iluminação e conversão em escalas de cinza que se torna preferencial ao utilizar certos algoritmos. (Rowley, 1996).

A Figura 8 representa bem o processo de tratamento da imagem antes de ser enviada para uma rede neural que fará o reconhecimento de características. Nela pode-se observar as entradas de diferentes faces detectadas por uma imagem e elas são recortadas, redimensionadas, ocorre a correção de brilho e equalização de histograma para ajustar o contraste. Estes tratamentos são exemplos dentre vários que auxiliam nas detecções de características dos modelos de reconhecimento.

Figura 8 — Pré-processamento pós detecção facial e antes da rede de reconhecimento



Fonte: Adaptada de Rowley (1997).

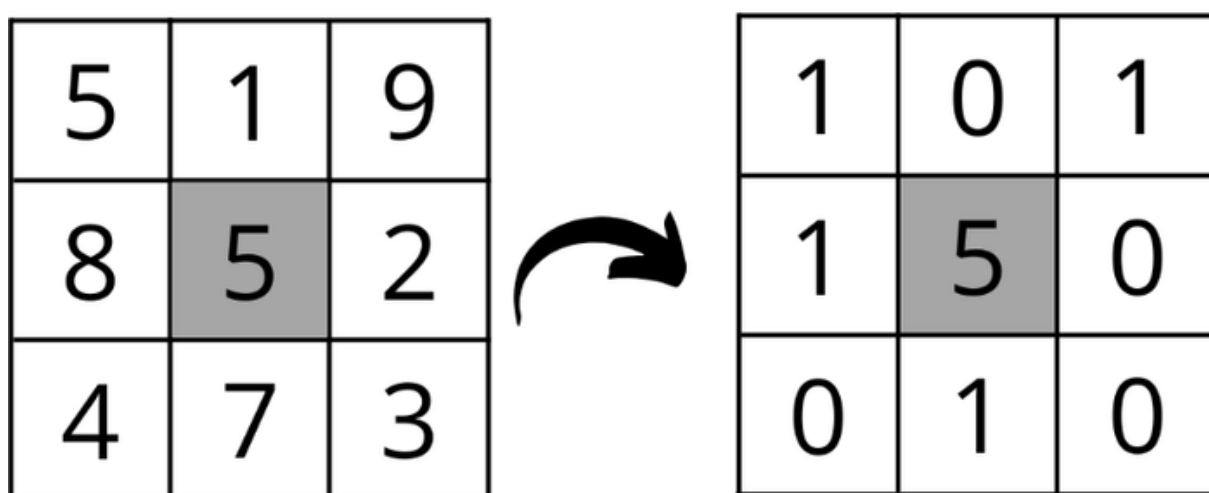
## 2.5 LBPH (*LOCAL BINARY PATTERNS HISTOGRAMS*)

Converter a imagem em escalas de cinza é crucial para que algoritmos como o *Local Binary Patterns Histograms* (LBPH) possam extrair as características da imagem. LBPH, ou Histograma de padrões binários locais, é um modelo que gera histogramas de características através da comparação entre pixels vizinhos que lhe permite identificar bordas com precisão tendo uma taxa de acertos até mesmo sob adversidades luminosas. (Ojala, Pietikäinen & Harwood, 1996)

O LBPH divide a imagem em diversas regiões de 3x3 pixels. Estes pixels recebem valores conforme sua intensidade na escala de cinza e o valor central é comparado com os valores vizinhos (Ojala, Pietikäinen & Harwood, 1996).

Os valores comparados caso sejam maiores que o valor central, se tornam 1 e caso sejam menores, se tornam 0 criando uma matriz binária. Combinando sentido horário começando da primeira posição da matriz, pode-se obter um número binário que ao ser convertido em decimal se torna o valor do histograma (Mahboob & Qadir, 2020). Este processo pode ser visto na Figura 9 a seguir onde o número binário é 10100101.

Figura 9 — Comparação de pixels e nova matriz binária

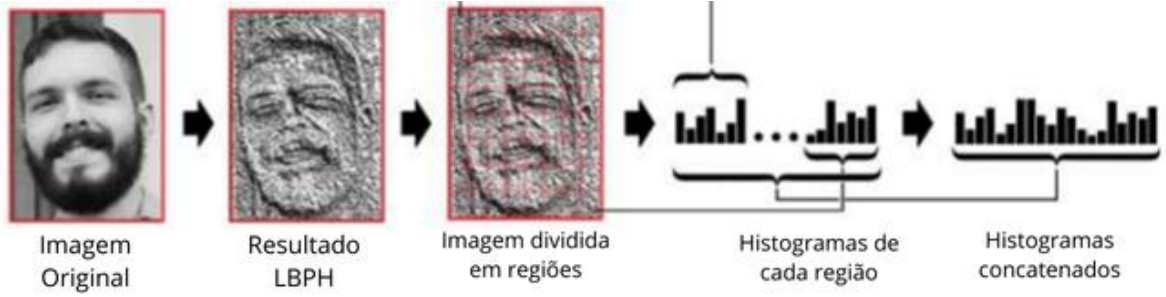


Fonte: adaptado de Ojala, Pietikainen e Hardwood (1996).

Por fim, os valores da matriz são somados para compor o histograma referente a cada área da imagem.

A Figura 10 representa este processo de tornar a imagem em um histograma de características, onde a imagem tem suas bordas bem destacadas e divididas em regiões e cada uma dessas regiões é representada por um histograma que em conjunto geram dados da imagem que serão salvos para o reconhecimento futuro ou comparados com a base de identidades.

Figura 10 — Extração de histogramas da imagem



Fonte: Adaptado de Mahboob e Qadir (2024).

## 2.6 FACENET

O *FaceNet* é um modelo de reconhecimento que utiliza de uma CNN de aprendizado profundo que processa imagens para gerar *embenddings*. *Embbendings* são vetores que carregam características específicas da face e as projetam em um espaço euclidiano de maneira que a distância entre elas indica o quão parecidas elas são (Dias, 2022). Para o cálculo da distância utiliza-se (1), onde  $p$  e  $q$  representam os valores individuais de cada vetor comparado e  $n$  é a quantidade de características de cada vetor.

$$d(p, q) = \sqrt{\sum_{i=1}^N (p_i - q_i)^2} \quad (1)$$

Seu treinamento é baseado no conceito de *Triplet Loss*, onde três imagens são comparadas, a âncora sendo uma imagem de referência, uma imagem da mesma pessoa que a âncora chamada de positiva e uma imagem negativa sendo uma pessoa diferente (Dias, 2022). O modelo de treinamento é representado matematicamente por (2):

$$d(p, q) = \sum_{i=1}^N \left[ \left| \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right| \right] \quad (2)$$



Deste jeito a função de extração de características  $f()$  faz com que as *embeddings* da âncora ( $x_i^a$ ) e positiva ( $x_i^p$ ) tem sua distância minimizada enquanto a distância da negativa ( $x_i^n$ ) tem sua distância maximizada. A ideia deste treinamento é que as três imagens selecionadas sejam um caso “difícil” e assim naturalmente será gerada uma distância menor tornando o modelo mais preciso.

## 2.7 MÉTRICAS DE AVALIAÇÃO

Segundo Goodfellow, Bengio e Courville (2016), o desempenho de modelos de aprendizado profundo não deve ser apenas medido por acertos gerais, mas também pela capacidade de distinguir classes distintas. Métricas como acurácia, F1-score, *recall*, precisão e tempo de processamento são cruciais para se ter uma visão detalhada do desempenho do modelo.

A acurácia é a proporção entre os acertos e o número total de amostras dado pela fórmula (3), já a precisão é porcentagem de verdadeiro positivo pela soma de verdadeiro positivo com falsos positivos, definida por (4), o *recall* é a sensibilidade do sistema, caracterizado pelos verdadeiros positivos dividido pela soma de verdadeiro positivo e falsos negativos (5) e por fim o F1-score é a média harmônica entre a precisão e o *recall* (6).

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \quad (3)$$

$$Precisão = \frac{VP}{VP + FP} \quad (4)$$

$$Recall = \frac{VP}{VP + FN} \quad (5)$$

$$F1 - score = \frac{Precisão \cdot Recall}{Precisão + Recall} \quad (6)$$

Onde VP é verdadeiro positivo, VN é verdadeiro negativo, FP é falso positivo e FN é falso negativo.

## 2.8 BANCOS DE DADOS DE FACES

Com o desenvolvimento das tecnologias de detecção e reconhecimento facial, foi fundamental a criação de bancos de dados com conjuntos de faces que possam servir para avaliar e comparar diferentes modelos. Esses bancos possuem um conjunto de fotos de determinados indivíduos com variações de posição, iluminação, objetos como óculos e barba especialmente criados para tentar aproximar a simulação de aplicações com a realidade e tornar a avaliação mais assertiva.

O *Yale Face* é um banco de dados de faces desenvolvido na universidade de San Diego e possui imagens de 15 pessoas diferentes, sendo 11 imagens de cada, totalizando 165 imagens, todas elas em escala de cinza e está disponível em: <https://vision.ucsd.edu/datasets/yale-face-database>. A Figura 11 mostra parte destas imagens (BELHUMEUR, 1997).

Figura 11 — Banco de faces Yalefaces



Fonte: Adaptado de BELHUMEUR (1997).

O banco de faces ORL ou AT&T é um banco desenvolvido pelo *AT&T Laboratories Cambridge* e é um pouco maior com um total de 400 imagens sendo dividida em 10 imagens de 40 indivíduos com variações de iluminação, posição, expressões faciais e outros detalhes como óculos e barba. Este banco está disponível em: <https://www.kaggle.com/datasets/kasikrit/att-database-of-faces>. A Figura 12 contém algumas das faces presentes nestes bancos de dados, vale notar que estas imagens já possuem certa troca de posição das faces e alguns acessórios como óculos (AT&T LABORATORIES CAMBRIDGE, 2002).

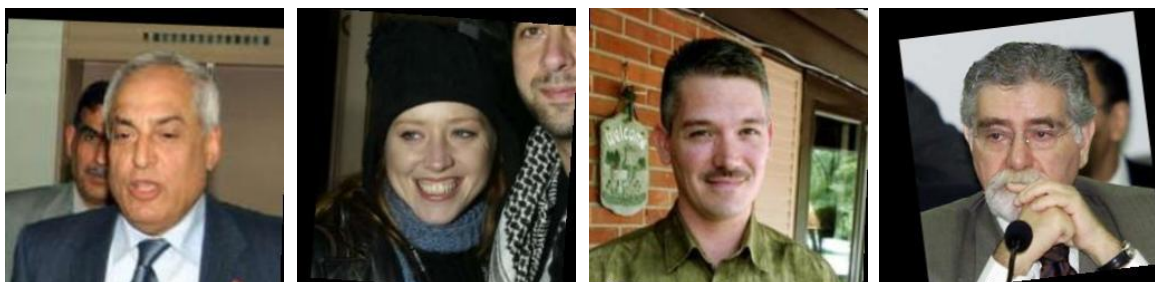
Figura 12 — Banco de faces AT&T



Fonte: Adaptado de AT&T Laboratories Cambridge (2002).

O *Labelled Faces in the Wild* (LFW) é um banco de faces com 13233 imagens de 5749 pessoas diferentes, criado por Gary B. Huang, Manu Ramesh, Tamara Berg e Erik Learned-Miller para a pesquisa *Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments*. University of Massachusetts, Amherst, Technical Report 07-49 (2007). Este banco possui uma grande quantidade de imagens com as mais diversas variações o que possibilita gerar melhores resultados para a maioria dos algoritmos de verificação. Ele está disponível em: <https://www.kaggle.com/datasets/jessicali9530/lfw-dataset> (HUANG et al., 2007). A Figura 13 representa algumas das faces no banco de dados LFW, nelas é possível reparar uma série de fatores que dificultam a detecção e o reconhecimento facial, como a iluminação, acessórios, posicionamento, e partes da face escondidas.

Figura 13 — Banco de faces LFW



Fonte: Adaptado de HUANG (2007).

### 3 DESENVOLVIMENTO DO TRABALHO

O sistema foi desenvolvido em *python* utilizando a IDE Pycharm com o ambiente Conda que facilita o gerenciamento e das bibliotecas utilizadas como *Tensorflow*, *Pytorch* e *OpenCV*, para que elas não entrem em conflito principalmente em casos que a aplicação exigia versões específicas. A tabela 1 contém todas as bibliotecas utilizadas pelo projeto, para sua instalação basta abrir o terminal adicionar o comando *pip install nome\_da\_biblioteca* para o interpretador *python* ou *conda install nome\_da\_biblioteca* para o interpretador *conda*.

Quadro 1 — Bibliotecas utilizadas

| Biblioteca      | Versão | Função                                                                         |
|-----------------|--------|--------------------------------------------------------------------------------|
| OpenCV          | 4.7    | Processamento de imagens, redimensionamento e desenho de <i>bounding boxes</i> |
| Numpy           | 1.23.5 | Manipulação de <i>arrays</i> e operações matemáticas                           |
| Torch           | 2.0.1  | Utilizada para <i>deep learning</i>                                            |
| Facenet-pytorch | 2.5.3  | Fornecer modelos pré-treinados de MTCNN e FaceNet                              |
| ONNXRuntime     | 1.14.1 | Inferência com modelos ONNX                                                    |
| scipy           | 1.10.1 | Utilizado para o cálculo de distância das <i>embeddings</i>                    |
| scikit-learn    | 1.2.2  | Possui métricas de avaliação                                                   |
| mtcnn           | 0.1.1  | modelo de detecção facial                                                      |
| json            | 2.0.9  | Leitura e escrita de arquivos JSON                                             |
| os              | -      | Interações com o sistema operacional (não requer instalação)                   |
| time            | -      | manipulação de tempo (não requer instalação)                                   |

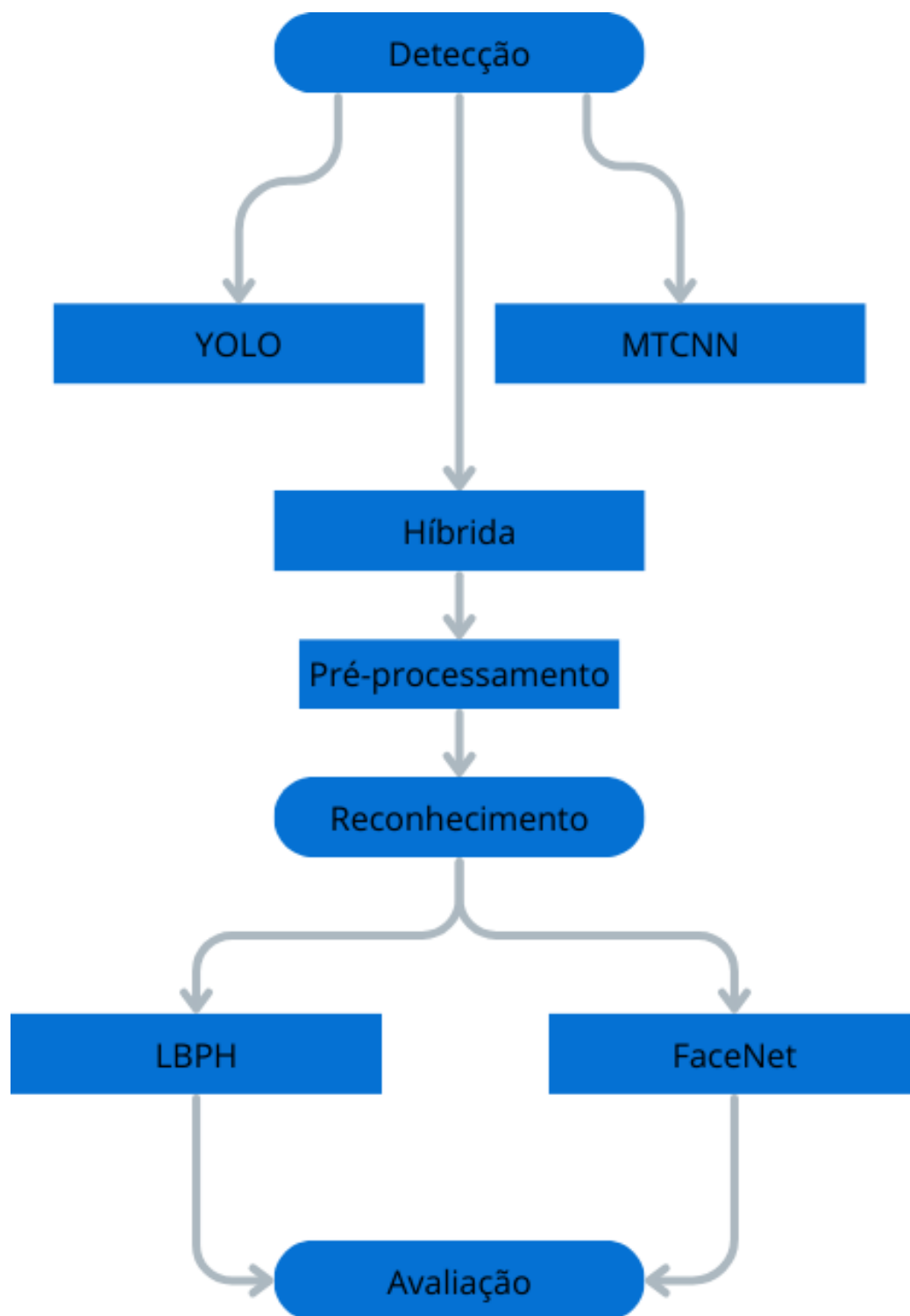
Fonte: O autor (2025).

Durante o desenvolvimento das etapas de detecção e reconhecimento, os testes foram realizados com o banco de dados Yalefaces que possui uma diversidade menor de imagens e com fotos do próprio autor por promoverem desafios um pouco menores para o sistema e permitia identificar ajustes necessários com mais facilidade.

O desenvolvimento do trabalho foi estruturado em 3 etapas principais: detecção de faces com modelos YOLO e MTCNN seguido reconhecimento facial e os métodos

LBPH e *FaceNet* e por avaliação do sistema e aquisição de resultados para comparação, a Figura 14 representa estas etapas.

Figura 14 — Diagrama das etapas do projeto



Fonte: O autor (2025).

### 3.1 DETECÇÃO

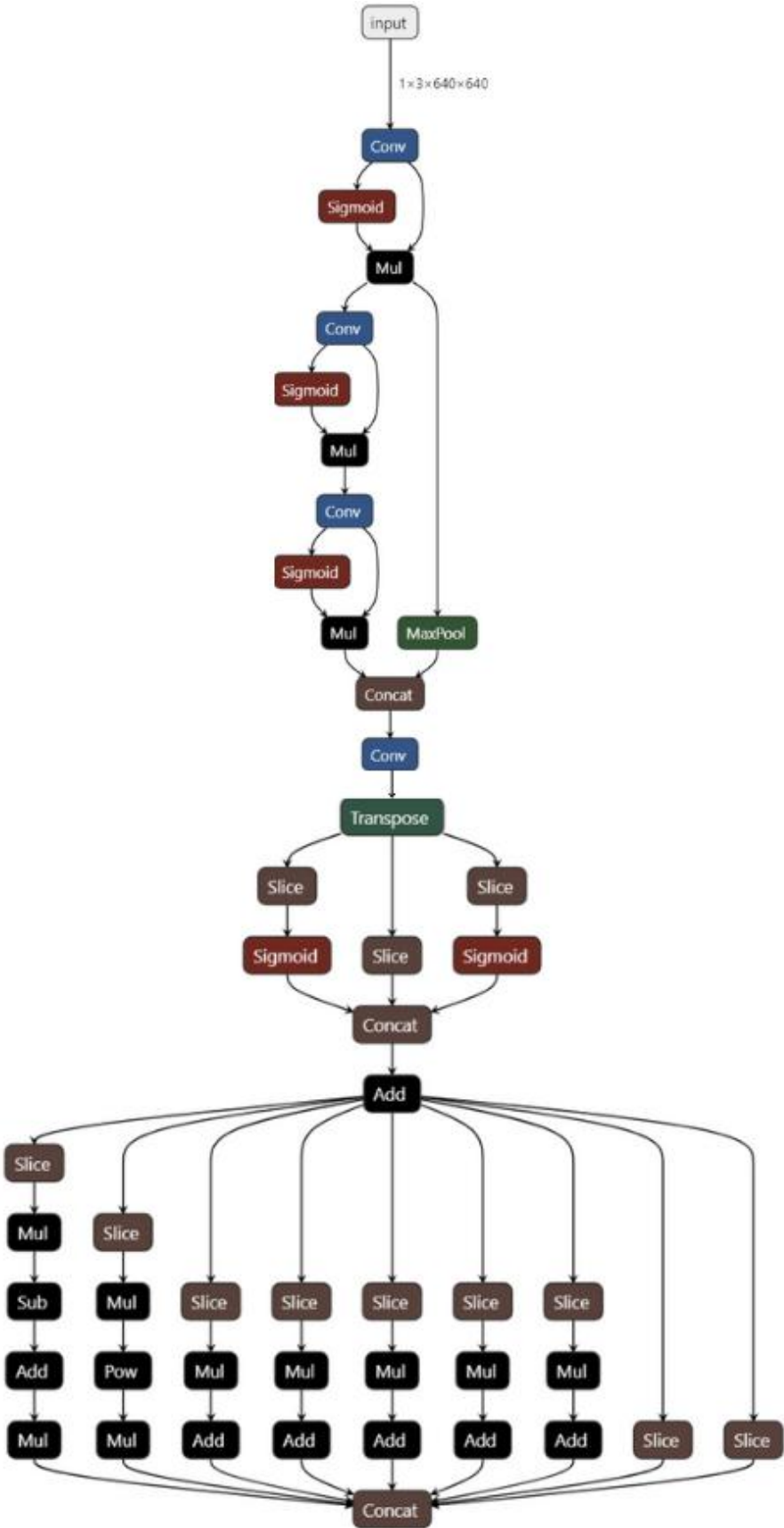
Inicialmente, a detecção de faces foi realizada de maneira independente do resto do projeto para o teste de funcionamento de ambos os modelos MTCNN e YOLO. Com a implementação e o bom funcionamento deles, iniciou-se a implementação da detecção híbrida que seria utilizada como fonte para os modelos de reconhecimento facial.

#### 3.1.1 YOLO

O modelo YOLO utilizado é fornecido e desenvolvido pela Ultralytics disponível no repositório do GitHub <https://github.com/ultralytics/yolov5.git> possuindo modelos pré-treinados e disponibilizando toda documentação. Para o bom funcionamento do YOLO é necessário que o Python do ambiente em que se encontra esteja entre as versões 3.7 e 3.10 e que a versão do PyTorch seja maior que 1.7

O modelo pré-treinado utilizado é de uma ramificação do repositório anterior e pode ser obtido em: <https://github.com/akanametov/yolo-face.git>. Este modelo foi escolhido por possuir um treinamento específico para faces. O arquivo de treinamento fornecido possui a extensão *onnx* portanto foi necessário utilizar a biblioteca *onnxruntime* para leitura do arquivo pelo código. A Figura 15 mostra um "resumo" da estrutura da rede neural YOLO utilizada

Figura 15 — Modelo YOLO Simplificado



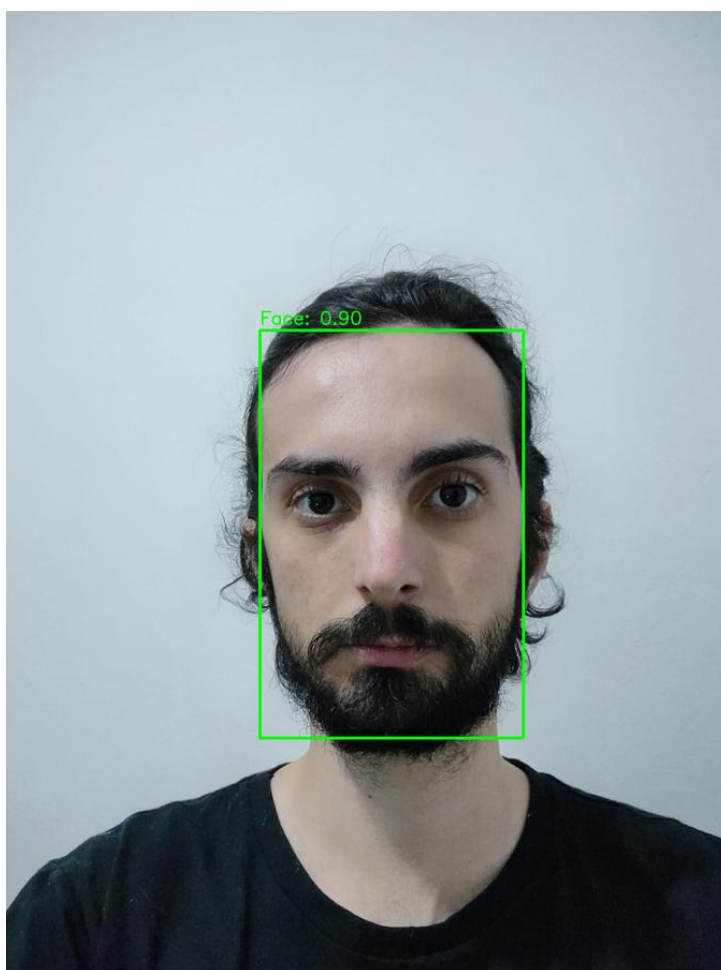
Fonte: O autor (2025).

O processo de desenvolvimento de detecção de faces pelo YOLO é simples, onde foi necessário apenas garantir que a imagem enviada para o modelo fosse 640x640. Para iniciar a detecção é necessário iniciar o modelo pré treinado com comando `start = onnxruntime.InferenceSession(path)` onde `path` é o caminho para o modelo. Então é utilizado o comando `start.run(imagem)` para realizar a detecção da imagem.

Como resultado, o modelo retorna as coordenadas de onde foram encontradas faces na imagem e com estas coordenadas é possível desenhar a *bounding box* utilizando o comando *rectangle* do OpenCV.

A Figura 16 é a saída os testes iniciais para o modelo YOLO. Acima da *bounding box* é possível observar o valor 0,90. Este valor é o grau de confiança C da detecção, foi determinado que apenas *bounding boxes* com grau de confiança maiores do que 0,6 sejam validados.

Figura 16 — Teste da detecção de faces pelo YOLO



Fonte: O autor (2025).



Já a Figura 17 mostra o mesmo Yolo detectando diversas faces, contendo alguns falsos positivos onde *bounding boxes* não possuem faces e falsos negativos, onde faces não foram detectadas.

Figura 17 — Teste Yolo em múltiplas faces

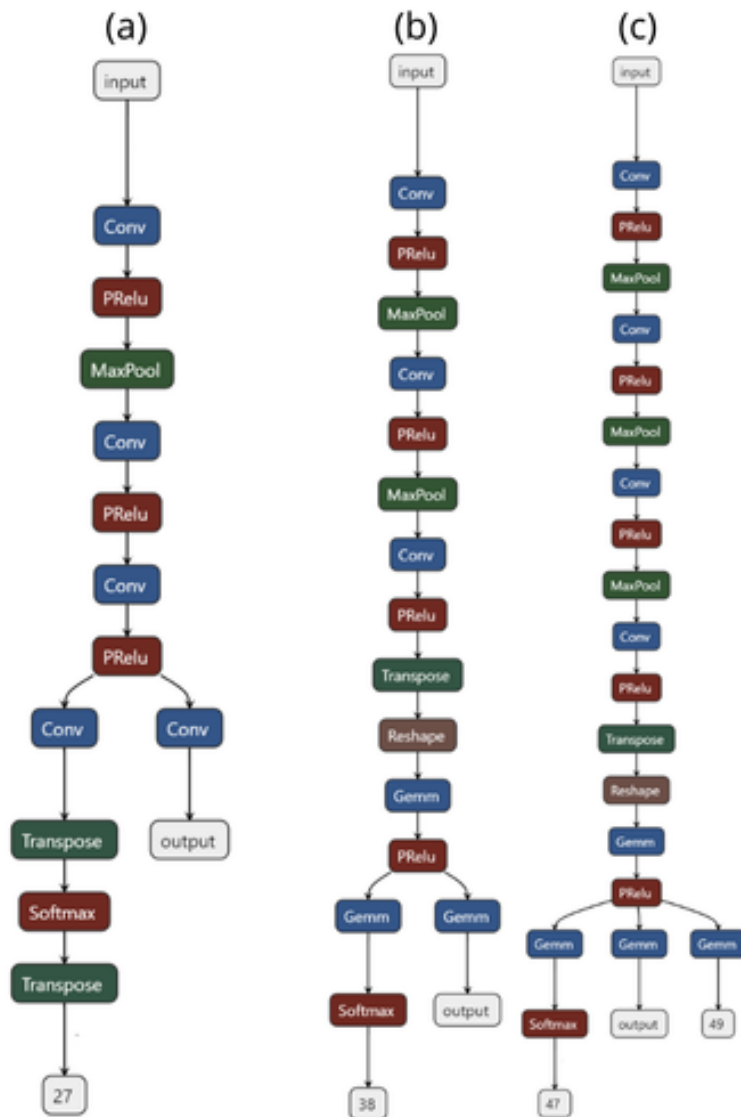


Fonte: O autor (2025).

### 3.1.2 MTCNN

O Modelo MTCNN está disponível como uma biblioteca em Python podendo ser instalado tanto pelo comando *pip install mtcnn* quanto pelo comando *conda install mtcnn* sendo necessário checar sua compatibilidade de versão com a biblioteca tensorflow que deve ser maior ou igual a 2.12. Uma maneira alternativa de adquirir o detector facial mtcnn é utilizando da biblioteca *facenet-pytorch* que se torna interessante por não depender de determinada versão do *tensorflow* e em testes, ambos obtiveram resultados semelhantes. A estrutura da rede neural MTCNN é dividida entre P-Net (a), R-Net (b) e O-Net (c) e estas são mostradas na Figura 18.

Figura 18 — Estrutura das redes neurais MTCNN



Fonte: O autor (2025).

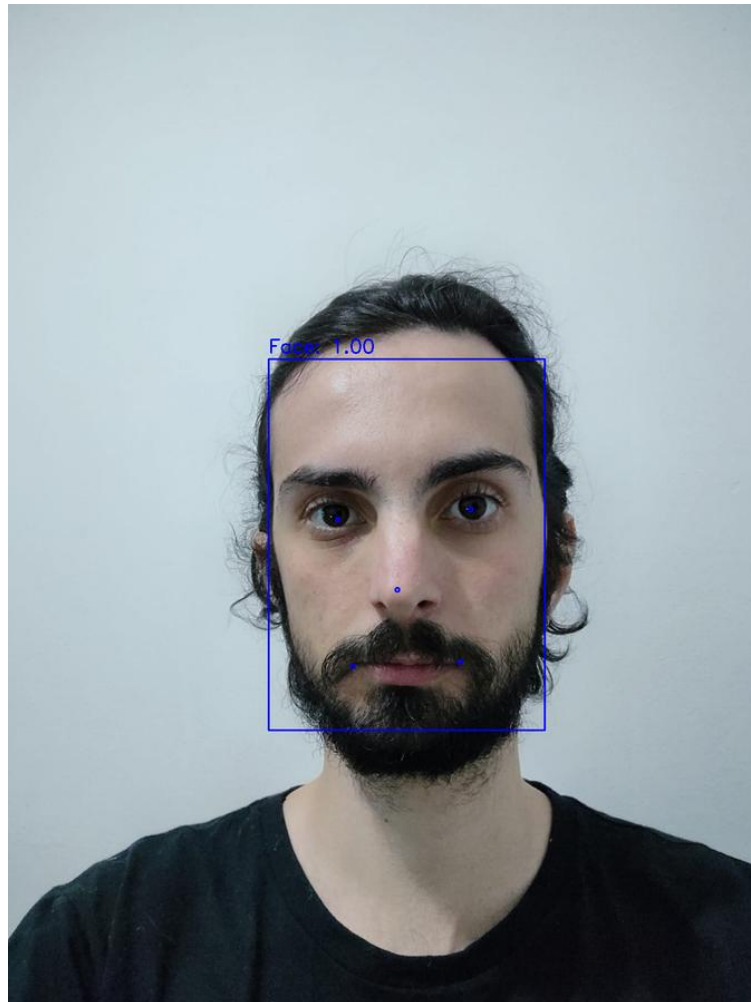
Para implementação é necessário da mesma maneira que o YOLO, iniciar o modelo mtcnn, para isso é utilizado esta linha de código: `detector = MTCNN()` e em sequência para a detecção é utilizado o código: `detections = detector.detect_faces(imagem)` onde imagem é o caminho para a imagem testada.

Diferente do modelo anterior, além das coordenadas da face, o MTCNN também retorna 5 pontos faciais, sendo eles um em cada olho, um no nariz e dois na região da boca.

Então além de utilizar o mesmo comando do OpenCV para desenhar as caixas delimitadoras, também é utilizado o comando `circle` para desenhar os pontos faciais.

O resultado do teste de detecção do modelo MTCNN pode ser visto na Figura 19 onde é possível ver a *bounding box*, os 5 pontos faciais e o grau de confiança da detecção desta imagem que foi máximo (1.0).

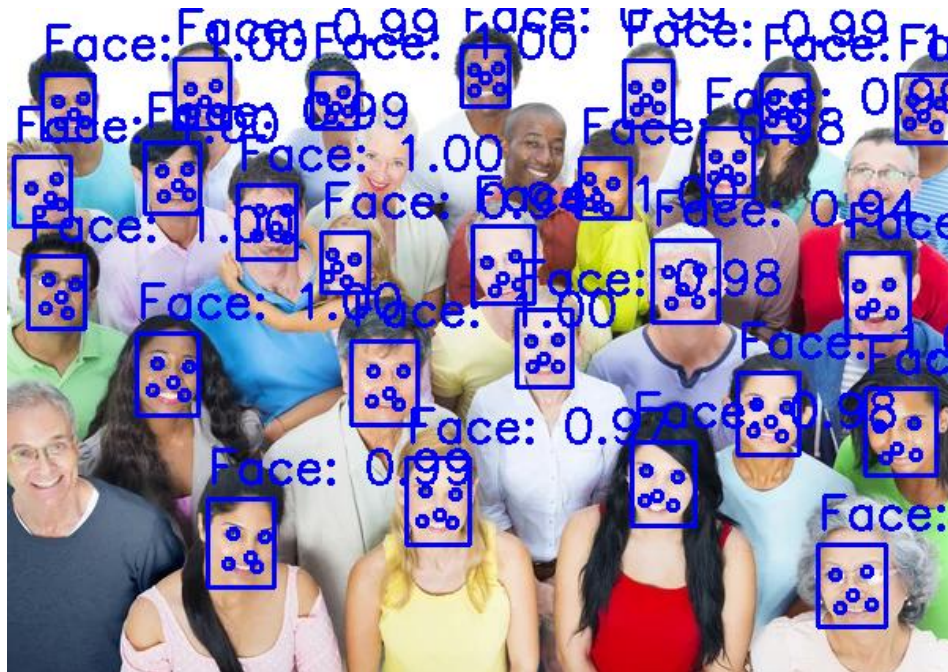
Figura 19 — Teste da detecção por MTCNN



Fonte: O autor (2025).

Já a Figura 20 mostra novamente a imagem com múltiplas faces, onde o modelo conseguiu identificar a maioria, com altos graus de confiança e adicionando as bounding boxes. Entretanto é possível observar alguns falsos negativos na imagem.

Figura 20 — Teste MTCNN em múltiplas faces



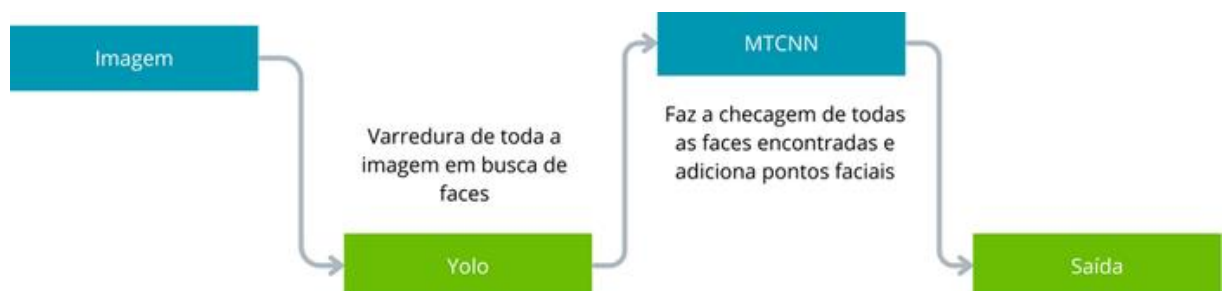
Fonte: O autor (2025).

### 3.1.3 DETECÇÃO HÍBRIDA

A detecção híbrida foi desenvolvida com o intuito de aproveitar a velocidade da detecção YOLO com a robustez da detecção MTCNN fazendo com que o sistema tenha uma menor taxa de falsos positivos nas imagens detectadas.

Como é possível ver na Figura 21 a detecção YOLO é feita inicialmente analisando completamente a imagem de maneira rápida e em seguida o MTCNN confere as faces detectadas e desenha as *landmarks* faciais.

Figura 21 — Processo de detecção de faces híbrida

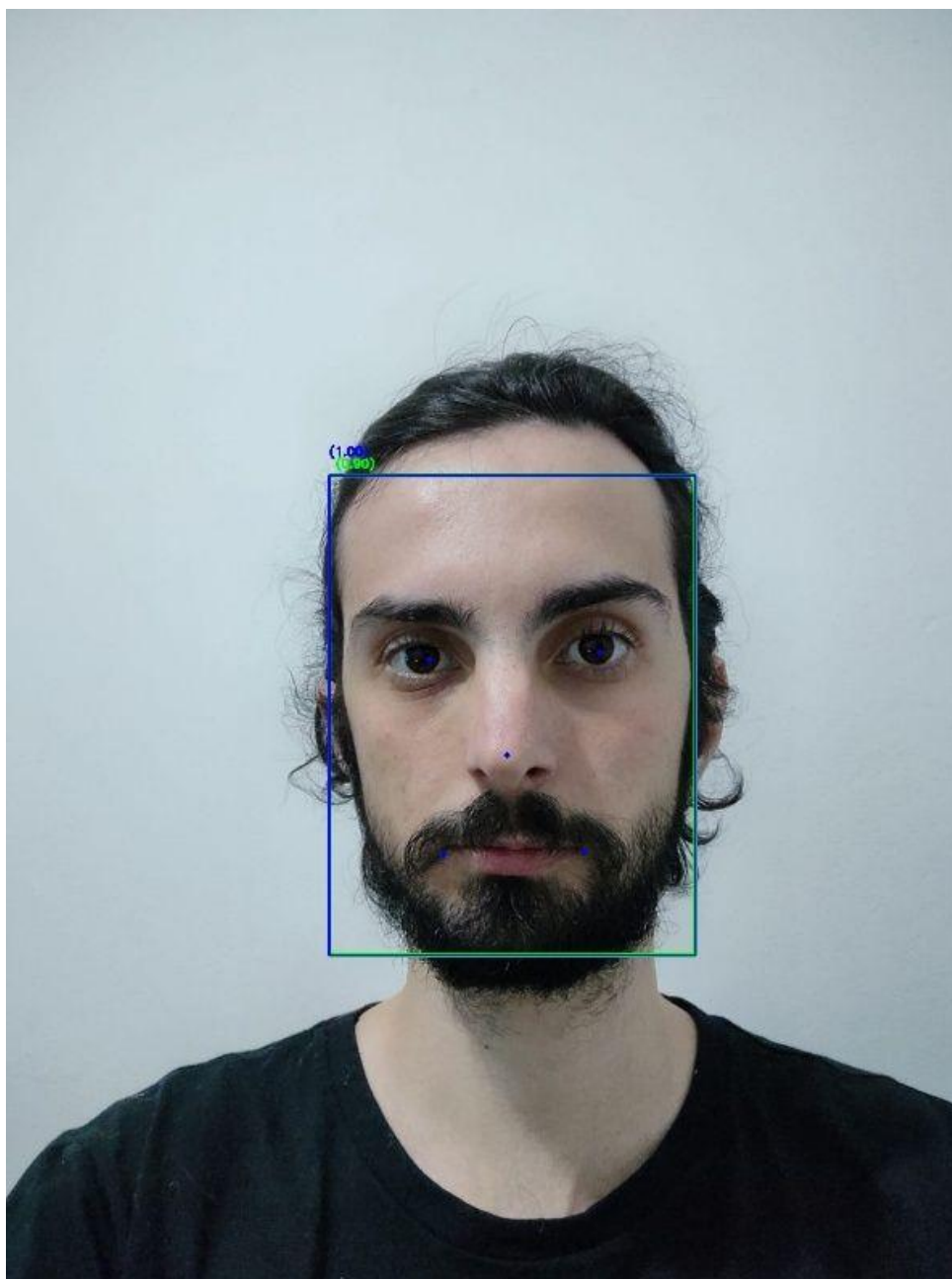


Fonte: O autor (2025).



A Figura 22 mostra a detecção de ambos os modelos em conjunto, nela em azul pode-se ver que ambos os modelos detectaram a face na posição correta e o MTCNN marcou os pontos faciais

Figura 22 — Teste da detecção híbrida dos modelos YOLO e MTCNN

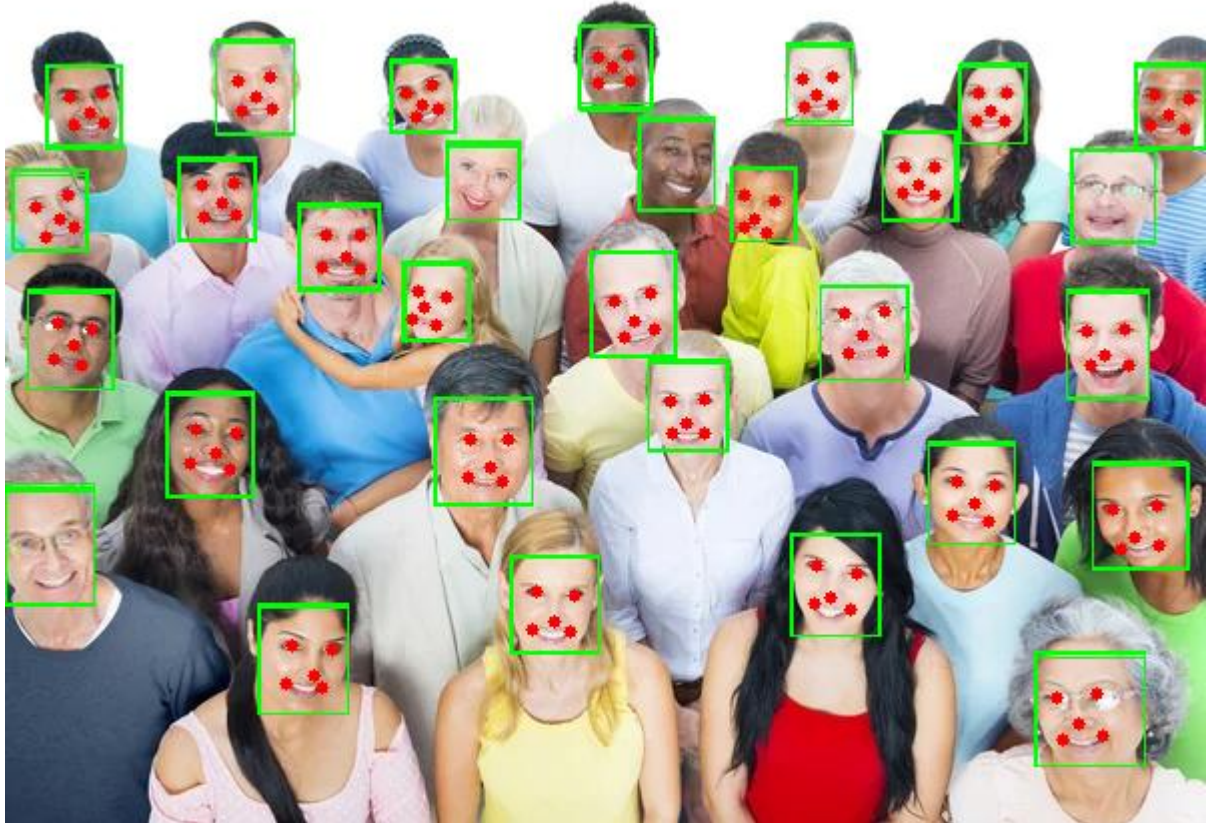


Fonte: O autor (2025).

A Figura 23 combina os modelos em uma imagem de múltiplas faces, entretanto utilizando de ambos os algoritmos em paralelo, o que abrange e verifica as faces, aumentando a precisão ao custo de velocidade de processamento. Nela é

possível observar que todas as faces foram identificadas sem que houvesse falsos positivos.

Figura 23 — Teste da detecção híbrida em múltiplas faces



Fonte: O autor (2025).

Esta será a detecção considerada nos próximos tópicos para os treinamentos e identificação dos modelos de reconhecimento facial.

### 3.2 PRÉ-PROCESSAMENTO

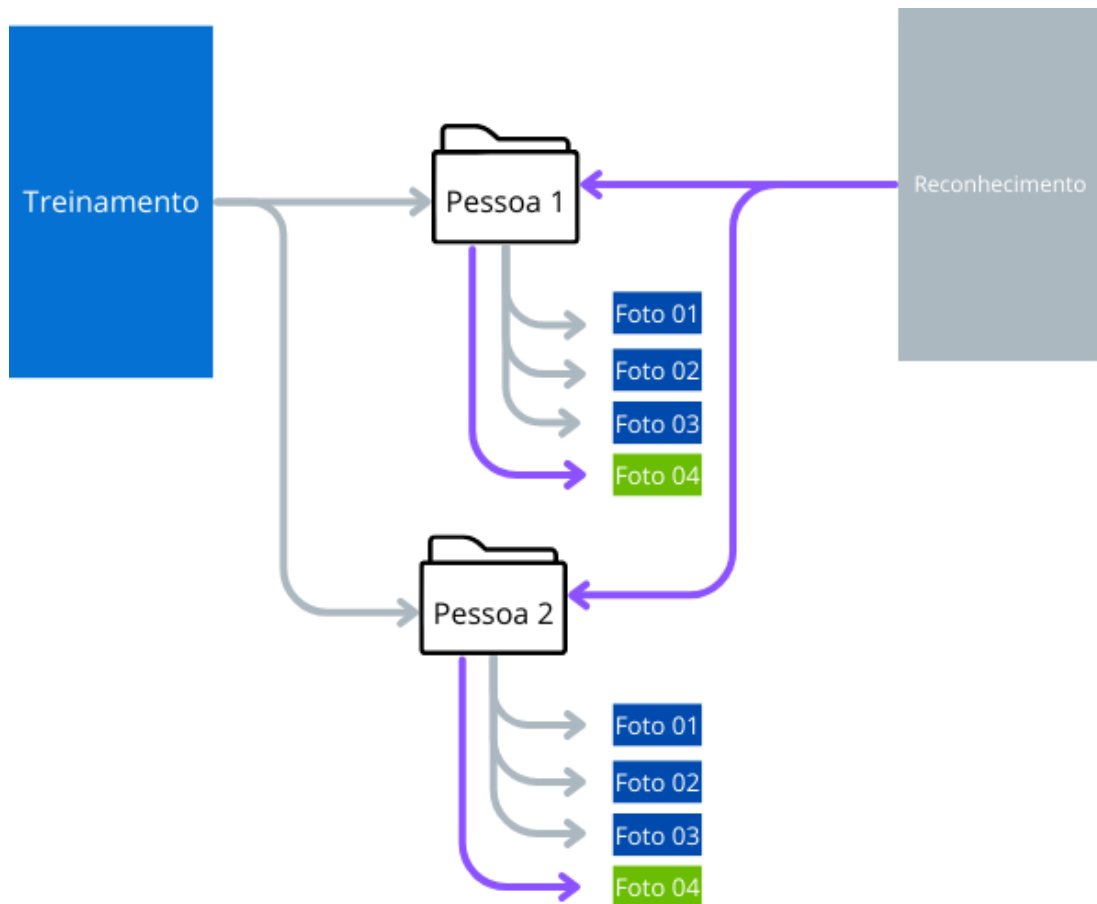
Com a detecção de faces finalizada, começou-se a preparar os modelos de reconhecimento facial. Os modelos LBPH e Facenet esperam dados de entrada diferentes então foi necessário de pré-processamento foram feitas individualmente em seus respectivos códigos. Enquanto para o LBPH foi necessário converter a imagem para tons de cinza e redimensionar a imagem para 100x100, o FaceNet requisitava de uma imagem de 3 canais (RGB), com dimensões de 160x160 e que fosse normalizada de -1 a 1.

### 3.3 RECONHECIMENTO FACIAL

Os treinamentos inicialmente foram realizados utilizando faces do banco de dados Yalefaces e com faces do autor, entretanto estes testes não serão levados em consideração nos resultados devido ao baixo grau de dificuldade das imagens. Outro ponto a ser levantado foi o erro de algumas *bounding boxes* que ao serem geradas na imagens dos bancos de dados acabavam ficando em posições inválidas devido a face estar muito próxima das bordas, por isso houve um ajuste mecânico no código para forçar elas a sempre estarem dentro dos limites válidos da imagem.

Devido ao número de imagens elevados destes bancos, foi desenvolvido um método para acessar todos os dados e dividi-los entre imagens de treinamento e imagens de reconhecimento que seriam testadas. Para isso, caso o processo sendo executado fosse treinamento, as primeiras imagens da pasta seriam utilizadas e caso o processo executado fosse o de reconhecimento, as imagens lidas seriam as últimas. Este processo pode ser visto na Figura 24

Figura 24 — Processo de divisão de imagens no banco de dados entre treinamento e reconhecimento



Fonte: O autor (2025).

### 3.3.1 LBPH (*Local Binary Patterns Histograms*)

O método de reconhecimento facial LBPH está disponível na biblioteca OpenCV e não necessita de nenhum modelo pré-treinado visto que ele funciona através da comparação de histogramas, então para adquiri-lo basta utilizar o comando `pip install opencv-python` no terminal ou `conda install opencv-python` para a variante conda do ambiente. Inicialmente foi verificado que o LBPH espera como entrada uma imagem 100x100 em escala de cinza, então foi criada uma função que recebe a face recortada da detecção, redimensiona a imagem para o tamanho necessário com o comando `resize(imagem,100,100)` e a converte em tons de cinza utilizando o comando `cv2.COLOR_BGR2GRAY`.

Para o treinamento, as imagens do banco de dados foram atreladas a rótulos e salvas através dos comandos `cv2.face.LBPHFaceRecognizer_create()` que cria um objeto para treinar o modelo e `train(faces, np.array(labels))` que vincula as faces à



seus respectivos rótulos. Estes dados são salvos em um arquivo .yml e os rótulos são salvos em um dicionário .json. A Figura 25 mostra este processo em que o modelo treinado rotula as imagens para a comparação futura.

Figura 25 — Processo de rotulação das imagens de treinamento



Fonte: O autor (2025).

Para o reconhecimento inicialmente são carregados os dados do treinamento e do dicionário e então ocorre o processo de pré-processamento semelhante ao do treinamento, porém para a imagem que será identificada.

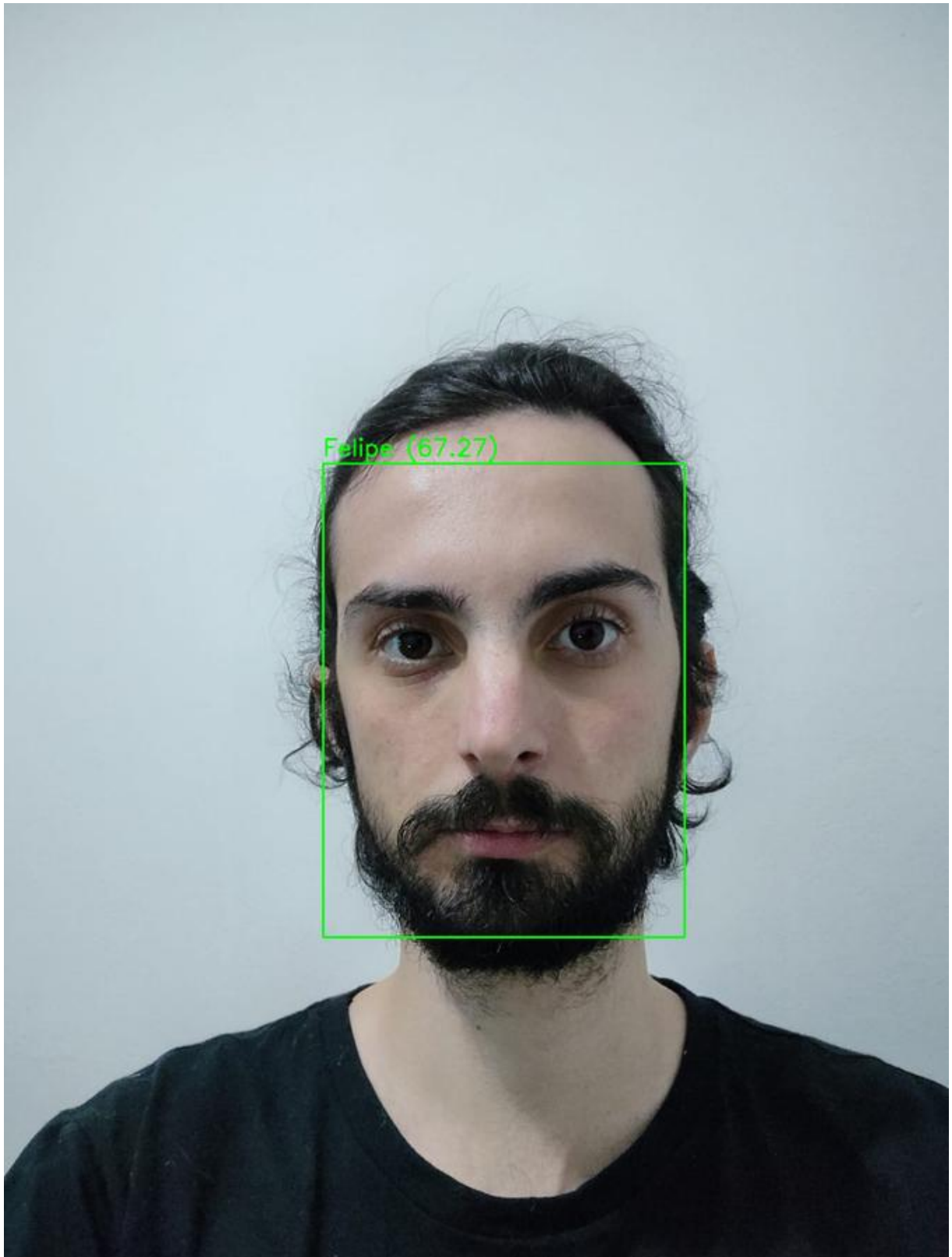
Em seguida é determinada a taxa de confiança. Diferente dos modelos anteriores, quanto menor o limite de confiança do LBPH maior é a certeza do modelo. A partir de testes realizados foi o valor do limite de confiança foi ajustado para 70. Caso a imagem ultrapasse este limite, o sistema retornará o rótulo desconhecido. A Figura 26 abaixo mostra a saída do sistema de reconhecimento LBPH e a Figura 27 é a imagem após reconhecida

Figura 26 — Saída do sistema de reconhecimento LBPH

```
Rostos detectados: 1
LBPH: Rosto em [405, 585, 864, 1187] - Label: Felipe, Confiança: 67.2676542768995
```

Fonte: O autor (2025).

Figura 27 — Imagem reconhecida pelo sistema LBPH



Fonte: O autor (2025).

### 3.3.2 FACENET

O método de redes neurais convolucionais para reconhecimento *FaceNet* é disponibilizado através da biblioteca *facenet-pytorch* que pode ser incorporada ao projeto através dos comandos *pip install facenet-pytorch* ou *conda install facenet-pytorch*. Por ser um método baseado em redes neurais convolucionais ele necessita de um pré-treinamento e o conjunto de dados utilizado foi o VGGFace2 que possui uma grande variedade de rostos já treinada tornando o algoritmo eficaz ao extrair características.

Para dar início ao treinamento, foi carregado o modelo FaceNet através do comando `InceptionResnetV1(pretrained='vggface2').eval().to(device)`, onde *pretrained* é o modelo de pré-treino, *eval* é o comando para ativar o modo avaliação que garante resultados mais eficientes na hora de salvar *embeddings* e *to(device)* permite a seleção de processamento do modelo na cpu ou na gpu. O FaceNet espera imagens no tamanho 160x160 então as imagens das faces coletadas nas etapas anteriores são redimensionadas para este tamanho e normalizadas. Então estas faces têm suas características são convertidas em *embeddings* e rótulos são adicionados a elas para identificações posteriores. Ao final do treinamento todas estas *embeddings* são salvas em um arquivo .npz

Com as *embeddings* criadas, é possível testar o modelo de reconhecimento FaceNet. Para isso a imagem a ser reconhecida passa pelo pré-processamento e uma *embedding* é criada, esta nova *embedding* é comparada com todas as outras previamente treinadas para encontrar a *embedding* mais próxima, caso a distância entre elas não ultrapasse grau de confiança determinado, a nova *embedding* recebe o rótulo a *embedding* da base de dados. Com o decorrer dos testes realizados foi estipulado que o limite ideal seria de 0.8 para esta aplicação.

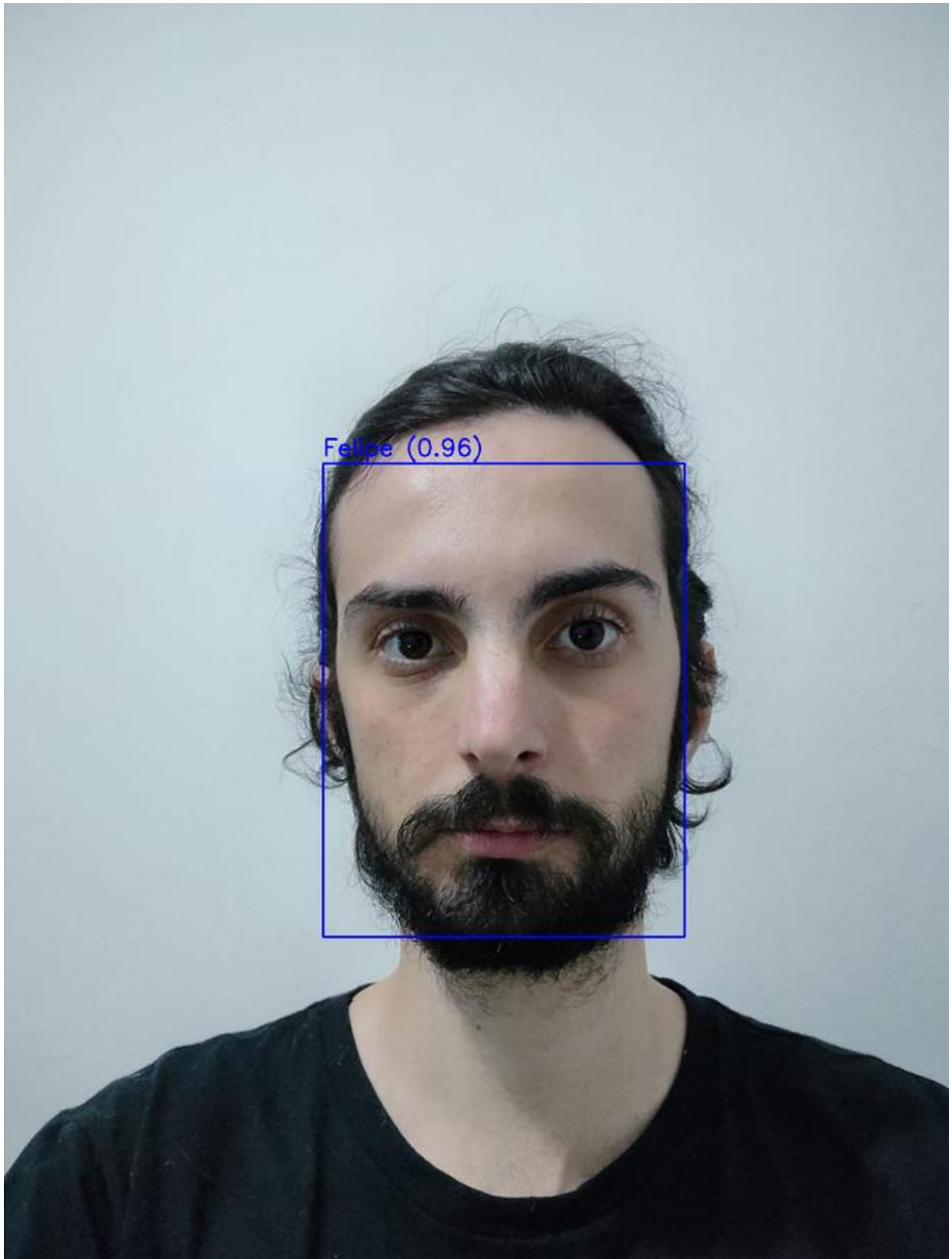
A Figura 28 mostra a identificação da face, o rótulo e o grau de confiança de um dos testes. Já a Figura 29 é a imagem de saída

Figura 28 — Dados da imagem identificada pelo método FaceNet

```
Rostos detectados: 1
FaceNet: Rosto em [405, 585, 864, 1187] - Rótulo: Felipe, Confiança: 0.9563337227404504
```

Fonte: O autor (2025).

Figura 29 — Imagem reconhecida pelo modelo FaceNet



Fonte: O autor (2025).

### 3.4 AVALIAÇÃO

Com os modelos de detecção e reconhecimento implementados, a etapa seguinte é pôr a prova o desempenho destes modelos. Para isso foi levado em consideração as métricas de acurácia, precisão, *recall*, *F1-score*, tempo médio de detecção e reconhecimento, total de rostos detectados e reconhecidos, falsos positivos e falsos negativos

A primeira avaliação foi feita com o banco AT&T que disponibiliza 10 faces de 40 pessoas distintas totalizando 400 imagens. Destas 320 foram utilizadas para o treinamento e 80 foram testadas.

Em seguida a segunda avaliação foi feita utilizando o banco de *faces Labelled Faces in the Wild (LFW)* que possui ao todo 13,233 imagens de 5749 pessoas diferente. Para estes testes foi desenvolvida uma lógica em que os treinamentos aprenderiam partes das imagens de cada identidade e ignorariam 1 ou 2 imagens da identidade para que estas fossem utilizadas para testar o modelo na etapa de reconhecimento.

## 4 RESULTADOS

Ambos os modelos tiveram êxito ao reconhecerem todas as imagens testadas pelo banco de dados *Yale Faces*, obtendo 100% de acurácia, precisão, recall e F1-score e como mencionado anteriormente este banco foi utilizado como teste por ter um certo grau de facilidade então ele não será considerado nos resultados.

Os primeiros testes com resultados relevantes foram utilizando o banco de dados AT&T. Por possuir características mais diversas acabou gerando alguns falsos positivos no modelo LBPH. O resultado de ambos os modelos para o banco de dados AT&T é representado pela Figura 30 onde (a) é o modelo LBPH e (b) é o modelo FaceNet. Nele pode-se observar que o modelo FaceNet não sentiu nenhuma dificuldade para identificar corretamente as faces ao ter acertado todas as imagens com um tempo de 0.39 segundos por imagem bem abaixo do tempo do modelo LBPH.

Figura 30 — Avaliação dos modelos LBPH e FaceNet para o banco AT&T

(a)

```
Resultados para LBPH:  
Acurácia: 0.80  
Precisão: 0.95  
Recall: 0.80  
F1-score: 0.85  
Tempo médio de detecção: 0.73 segundos por imagem  
Tempo médio de reconhecimento: 0.39 segundos por imagem  
Total de rostos detectados: 80  
Total de rostos reconhecidos corretamente: 64  
Falsos positivos: 16  
Falsos negativos: 0
```

(b)

```
Resultados para FaceNet:  
Acurácia: 1.00  
Precisão: 1.00  
Recall: 1.00  
F1-score: 1.00  
Tempo médio de detecção: 0.73 segundos por imagem  
Tempo médio de reconhecimento: 0.39 segundos por imagem  
Total de rostos detectados: 80  
Total de rostos reconhecidos corretamente: 80  
Falsos positivos: 0  
Falsos negativos: 0
```

Fonte: O autor (2025).

Para o banco de dados LFW os resultados mostram que ambos os modelos tiveram uma dificuldade um pouco maior. O modelo FaceNet ainda teve um alto desempenho enquanto o LBPH teve uma grande queda principalmente em sua precisão e F1-score. A Figura 31 mostra os resultados da avaliação LBPH (a) e FaceNet (b) para o banco de dados mais complexo LFW

Figura 31 — Resultados dos modelos de reconhecimento facial para o banco de dados LFW



Fonte: O autor (2025).

A Tabela 1 mostra de uma maneira mais clara a comparação entre os dois modelos e faz uma média do desempenho de ambos onde o modelo de redes neurais FaceNet se mostrou superior ao modelo clássico LBPH.

Tabela 1 — Resultados dos modelos de reconhecimento

| Modelo   | AT&T |         | Labelled Faces in the Wild |         | Média |         |
|----------|------|---------|----------------------------|---------|-------|---------|
|          | LBPH | FaceNet | LBPH                       | FaceNet | LBPH  | FaceNet |
| Acurácia | 80%  | 100%    | 72%                        | 91%     | 76%   | 95,5%   |
| Precisão | 95%  | 100%    | 68%                        | 89%     | 81,5% | 94,5%   |
| Recall   | 80%  | 100%    | 72%                        | 91%     | 76%   | 95,5%   |



|                 | AT&T  |        | <i>Labelled Faces in the Wild</i> |          | Média  |       |
|-----------------|-------|--------|-----------------------------------|----------|--------|-------|
| <i>F1-Score</i> | 85%   | 100%   | 69%                               | 90%      | 77%    | 95%   |
| <i>Tempo</i>    | 1.1 s | 0.39 s | 0.5664 s                          | 0.3351 s | 0.83 s | 0.36s |

Fonte: O autor (2025).

Outra comparação interessante é a de mesmos modelos em diferentes pesquisas. O artigo de Reddy, Ramesh, Praveen, Surekha e Sharma (2023) propõe o desenvolvimento de um sistema de reconhecimento facial a base de MTCNN e FaceNet. A Tabela 2 compara os valores adquiridos por este estudo com os valores do artigo de Reddy, Ramesh, Praveen, Surekha e Sharma e é possível reparar na proximidade entre parâmetros avaliados.

Tabela 2 — Comparação entre o modelo desenvolvido por este estudo e o desenvolvido pelo artigo de Reddy, Ramesh, Praveen, Surekha e Sharma (2023)

| Modelo          | Autor | Reddy  |
|-----------------|-------|--------|
| Precisão        | 91%   | 89,74% |
| <i>Recall</i>   | 89%   | 94,59% |
| <i>F1-Score</i> | 91%   | 92,06% |
| Acurácia        | 90%   | 86,67  |

Fonte: Autor e Reddy *et al.* (2023).

A precisão de ambos os modelos está próxima, entretanto, o modelo de reconhecimento desenvolvido neste estudo é ligeiramente superior o que indica um maior acerto na previsão de classes reduzindo a quantidade de falsos positivos.

Já o modelo de Reddy *et al.* (2023) possui a métrica de *recall* melhor, o que indica uma maior capacidade de identificar corretamente as classes previstas também minimizando falsos negativos

A métrica de *F1-score* foi a mais próxima entre os dois modelos com uma variação de 1,06%, o que sugere que ambos os modelos têm uma capacidade semelhante de classificar corretamente amostras, minimizando falsos positivos e negativos.

Por fim, a acurácia do modelo desenvolvido neste estudo foi maior que o desenvolvido por Reddy, indicando uma maior capacidade geral de previsões corretas pelo modelo.

## 5 CONSIDERAÇÕES FINAIS

Ao decorrer do projeto foi possível trazer características interessantes sobre dois modelos de reconhecimento facial além de aplicar uma técnica não muito usual de combinação de detecção de faces. Comparando os resultados, o FaceNet se provou aproximadamente 20% melhor que o do LBPH em todos os parâmetros avaliados e pode-se dizer que os resultados eram previstos, visto que o modelo LBPH já é um modelo mais antigo enquanto o FaceNet já possui tecnologias mais avançadas que lhe permitem desempenhar melhor em casos mais difíceis.

Entretanto, o LBPH se mostrou eficiente em situações simples onde não há muita variação de posição, iluminação ou objetos cobrindo partes do rosto, como ocorreu no banco *Yalefaces* em que ele não teve erros ou até mesmo no banco AT&T onde ele teve uma precisão de 95% mostrando que pode ser muito bem aproveitado em situações de ambiente controlado, onde iluminação, posicionamento, acessórios, entre outros fatores que podem atrapalhar o reconhecimento possam ser controlados e variações minimizadas. Já o FaceNet se mostrou mais completo, robusto e eficiente onde no banco de faces AT&T ele não errou nenhuma das imagens testadas e foi mais de 2 vezes mais rápido que o LBPH e no banco de faces LFW manteve seus parâmetros de avaliação por volta de 90% o que deixa claro que os avanços nas áreas de redes neurais são extremamente benéficos para o processamento de imagem.

Já na comparação com o modelo desenvolvido em Reddy *et al.* (2023), o modelo FaceNet desenvolvido apresentou uma maior precisão e acurácia (91% contra 89,74% e 90% contra 8,67% respectivamente) o que demonstra uma maior capacidade de identificar corretamente as classes e a proximidade no parâmetro avaliativo *F1-score* (diferença de 1,06%) indicando que ambos têm boa relação entre taxas de acertos em previsões e identificação de classes.

Embora diferentes, a média de variação entre os parâmetros de avaliação foi de aproximadamente 3%. Essa proximidade não só valida os resultados obtidos por esta pesquisa como também indica que o modelo FaceNet possui uma certa linearidade e eficácia em diferentes implementações mostrando ser um modelo confiável para desenvolvimento.

Com os resultados obtidos também pode-se afirmar que a tecnologia de reconhecimento facial tem um grande espaço no mundo moderno sendo um sistema biométrico rápido e eficiente.

## REFERÊNCIAS

AT&T Laboratories Cambridge. The Database of Faces. 2002. Disponível em: <<https://cam-orl.co.uk/facedatabase.html>>. Acesso em: 28 fev. 2025.

BELHUMEUR, P. N.; HESPANHA, J. P.; KRIEGMAN, D. J. Yale Face Database. 1997. Disponível em: <<https://cvc.cs.yale.edu/cvc-projects/yalefaces/yalefaces.html>>. Acesso em: 28 fev. 2025.

BONELA, Bruno Novelli. **Contagem de pessoas no interior de edificações por técnicas de visão computacional**. Vitória, 2021. 46 p Tese (Engenharia Elétrica) - Universidade Federal do Espírito Santo.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. MIT Press, v. 3, f. 401, 2016. 801 p.

HUANG, G. B.; MATTAR, M.; BERG, T.; LEARNED-MILLER, E. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. 2007. Disponível em: <<http://vis-www.cs.umass.edu/lfw/>>. Acesso em: 28 fev. 2025.

JUNIOR, Alexandre Pereira; TEIXEIRA, Fabio Oliveira; HOMEM, Thiago Pedro Donadon. **Aplicação de Visão Computacional para o Monitoramento do Uso de Máscaras de Proteção**. 2021. 6 p Tese (Sistemas de Informação) - Instituto Federal de São Paulo.

LOPES, Eduardo Costa. **Detecção de Faces e Características Faciais**. Porto Alegre, f. 41 Tese (Ciência da Computação) - Pucrs.

MAHBOOB, Mariyam ; QADIR, Nounman Mohsin Abdul Qadir. **Attendance Management System Using Face Recognition**. 2024. 222 p Tese.

MURPHY, Kevin P.. **Machine Learning: A Probabilistic Perspective**. MIT Press, v. 3, f. 551, 2012. 1102 p.

OJALA, Tomo; PIETIKAINEN, Matti ; HARDWOOD, David. **A Comparative Study Of Texture Measures With Classification Based On Feature Distributions**. 1996. 59 p Tese (Department of Electrical Engineering) - University of Oulu/university of Maryland.

PIVA JUNIOR, Dilermando *et al.* **Detecção e reconhecimento facial: uma abordagem prática**. Salto, SP: FoxTablet, 2023. 321 p.

REDDY, K. Shyam Sunder *et al.* **A Real-time Automated System for Object Detection and Facial Recognition.** ICMPC, 2023. 10 p Tese.

REDMON, Joseph *et al.* **You Only Look Once: Unified, Real-Time Object Detection.** 2016. 10 p Tese (Allen Institute for AI) - University Of Washington.

RODRIGUES, Diego Alves. **DEEP LEARNING E REDES NEURAIAS CONVOLUCIONAIS: RECONHECIMENTO AUTOMÁTICO DE CARACTERES EM PLACAS DE LICENCIAMENTO AUTOMOTIVO.** 2018. 37 p Trabalho de Conclusão de Curso (Ciência da Computação) - Universidade Federal da Paraíba.

ROWLEY, Henry A. **A Trainable View-Based Object Detection System Thesis Proposal.** 1997. 20 p Tese.

VELOSO DIAS, M. **Otimização de algoritmo de Reconhecimento Facial via aplicação de arquitetura FaceNet Residual baseada em Deep Learning.** 10 p Trabalho de Conclusão de Curso (Engenharia Elétrica) - Cefet-mg.

ZHANG, Aston *et al.* **Dive into Deep Learning.** Cambridge University Press, v. 1, f. 291, 2023. 581 p.

ZHANG, Kaipeng ; ZHANG, Zhanpeng; LI, Zhifeng. **Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks.** 2016. 5 p Tese.