DESENVOLVIMENTO DE UM SISTEMA DE RECONHECIMENTO FACIAL PARA CONTROLE DE ACESSO E SEGURANÇA NO CAMPUS IFPE PAULISTA

Gabriel H. Santana Santos

ghss1@discente.ifpe.edu.br

Rodrigo C. L. da Silva

rodrigo.lira@paulista.ifpe.edu.br

RESUMO

Este trabalho aborda a relevância crescente do reconhecimento facial para segurança em instituições de ensino, motivado por recentes episódios de violência escolar, e propõe um sistema de controle de acesso para o campus Paulista do IFPE, cuja metodologia engloba revisão de literatura, seleção de algoritmos de *deep learning* (Res10-SSD para detecção e FaceNet para embeddings) e implementação em Python com OpenCV e face_recognition para operação em tempo real. O protótipo processou 120 faces a taxas entre 17,44 e 31 FPS, confiança média acima do threshold definido. O que permitiu identificar corretamente usuários autorizados e intrusos, embora oscilações de desempenho sob carga e variações de iluminação revelem a necessidade de otimizações no pré-processamento e de diversificação do *dataset*.

Palavras-chave: Controle de acesso; Reconhecimento facial; Visão computacional; Deep learning.

ABSTRACT

This work addresses the growing relevance of facial recognition for security in educational institutions, motivated by recent episodes of school violence, and proposes an access control system for IFPE's Paulista campus, whose methodology includes a literature review, selection of deep learning algorithms (Res10-SSD for detection and FaceNet for embeddings) and implementation in Python with OpenCV and face_recognition for real-time operation. The prototype processed 120 faces at rates between 17.44 and 31 FPS, with average confidence above the defined

Instituto Federal de Educação, Ciências e Tecnologia de Pernambuco. *Campus* Paulista. Curso de Análise e Desenvolvimento de Sistemas. 2025.

threshold. This allowed it to correctly identify authorized users and intruders, although performance fluctuations under load and lighting variations reveal the need for optimizations in pre-processing and diversification of the dataset.

Keywords: Access control; Facial recognition; Computer vision; Deep learning.

---

1 INTRODUÇÃO

O avanço da Inteligência Artificial (IA) e das tecnologias de reconhecimento facial tem desempenhado um papel significativo no panorama da segurança e controle de acesso(Mandru, 2022). A precisão aprimorada do reconhecimento facial transformou a maneira como a segurança é lidada em diversos setores, incluindo ambientes acadêmicos (Santoso; Safitri; Samidi, 2024).

Um relatório liderado por Telma Vinha (GEDDEP-IdEA/Unicamp e GEPEM/Unesp-Unicamp) analisou ataques violentos em escolas brasileiras entre 2001 e outubro de 2023. O estudo destaca que 2023 foi marcado por inúmeros episódios de violência extrema em todo o país. Para compreender o fenômeno e propor medidas de mitigação, os pesquisadores documentaram 36 ocorrências apenas no biênio 2022-2023. Esse acúmulo em tão curto período revela tendência preocupante de escalada da violência em ambientes educacionais (Vinha et al., 2023).

No ano de 2023, um ataque à Universidade Charles, no centro de Praga, capital da República Tcheca, resultou em 14 mortos e 25 feridos, conforme relatado pela BBC News Brasil (2023). O episódio, considerado o mais mortal desde a independência do país há 30 anos, reforça a necessidade de medidas preventivas em ambientes educacionais. Nesse contexto, sistemas de controle de acesso como o reconhecimento facial podem ser cruciais para identificar pessoas não autorizadas e ativar respostas emergenciais rápidas, evitando tragédias semelhantes. Essa tecnologia, se integrada a protocolos de segurança, teria potencial para monitorar entradas/saídas e alertar autoridades sobre comportamentos suspeitos, garantindo maior proteção a alunos e funcionários (Li et al., 2020).

Diante do cenário de ataques a instituições educacionais, este trabalho propõe uma solução para fortalecer a segurança no campus Paulista. A proposta consiste na implementação de um sistema de controle de acesso baseado em reconhecimento facial. Ao integrar tecnologias de IA, o sistema visa validar de maneira rápida e precisa a entrada de pessoas autorizadas, prevenindo o acesso de indivíduos não autorizados. Essa abordagem não apenas contribuirá para a identificação precoce de potenciais ameaças, mas também permitirá uma resposta ágil em situações de emergência, minimizando riscos e protegendo a comunidade acadêmica. A partir da implementação desse trabalho espera-se que pessoas

autorizadas, como alunos, docentes e funcionários, possam acessar o local e serem reconhecidos pela aplicação, já as pessoas não autorizadas vão ser abordadas corretamente e identificadas, aprimorando a segurança e permitindo um monitoramento eficiente das entradas.

Para garantir um monitoramento eficiente das entradas, este trabalho implementa um sistema de reconhecimento facial em tempo real baseado em deep learning, utilizando o algoritmo ResNet-10 SSD (Single Shot MultiBox Detector) (REN et al., 2017). A abordagem foi selecionada por sua robustez em cenários dinâmicos, como os observados no campus do IFPE Paulista.

O trabalho está dividido da seguinte forma: na Seção 2 apresenta-se o referencial teórico; na Seção 3 detalha-se a metodologia, incluindo seleção de algoritmos, implementação do sistema e definição das métricas de avaliação; na Seção 4 são expostos e discutidos os resultados obtidos; e, por fim, na Seção 5 são apresentadas as conclusões e propostas de trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 Controle de acesso

Controle de acesso e uma ferramenta de segurança que por meio de políticas, procedimentos, dispositivos, hardware e software, métodos qualificados de identificação e sistemas de bloqueios, tem o objetivo de controlar e gerenciar a entrada de pessoas em determinados lugares, esse controle é feito por maneiras físicas como catracas, portas, portões e dentre outros (Marcondes, J. S., 2020).

Um sistema de controle de acesso tem como objetivo geral proporcionar proteção a estabelecimentos, ferramentas, dados, bens e pessoas, impossibilitando o acesso de pessoal não autorizado aos ambientes físicos ou lógicos. Ele pode ser classificado em duas categorias quanto ao seu funcionamento ou quanto a sua função, sendo respectivamente dividido em controle manual, semiautomático ou automático e controle de acesso lógico ou controle de acesso físico, o reconhciemnto por exemplo e um meio de controle de acesso biológico (Marcondes, J. S., 2020).

2.2 Autenticação

A autenticação é parte importante no controle de acesso, como descrita em Makhsud (2021), como o processo de verificação da autenticidade de um usuário, processo ou dispositivo. Essa verificação permite que o usuário (ou processo, ou dispositivo) tenha certeza de que é genuíno. Durante o processo de autenticação, a parte examinadora garante que a parte auditada é real, e a parte auditada está ativamente envolvida na troca de informações.

O processo de autenticação acontece em três etapas básicas, essas etapas incluem o estado inicial, onde o requerente não está autenticado, assim sendo impedido de adentrar ou ter acesso a quaisquer recursos, seguido pela etapa de

conexão, onde o usuário fornece ao sistema credenciais válidas de autenticação. Essas credenciais podem ser diversas técnicas como as mais tradicionais como usuário e senha, PINs (Personal Identification Number), bem como podem ser abordagens mais avançadas, como métodos biométricos como impressões digitais, reconhecimento facial, íris até mesmo reconhecimento de ouvidos. Se bem-sucedido, o sistema estabelece uma sessão autenticada, permitindo que o usuário acesse as funções necessárias. Por fim, a etapa de desconexão encerra a sessão, retornando o sistema ao seu estado inicial. Essas etapas são essenciais para garantir a segurança e o controle de acesso adequado, e observam que os sistemas podem implementar diferentes níveis de autenticação para gerenciar diferentes direitos de acesso (Zulkarnain *et a*l., 2013).

2.3 Biometria

Com os avanços tecnológicos surgiram sistemas que são capazes de fazer autenticação baseados em biometria, essa tecnologia utiliza características físicas e comportamentais dos indivíduos para autenticação e identificação. Os meios de autenticação biométricos mais conhecidos são: face, voz, íris e impressões digitais (Yusuf *et al.,* 2020).

A autenticação biométrica teve melhores resultados quando os algoritmos de aprendizagem de máquina apareceram, esses algoritmos ajudam significativamente vários pontos da autenticação como: melhoria da eficiência, coleta de dados, combinação de pistas comportamentais e fisiológicas. Isso mostra como a integração de algoritmos de aprendizado de máquina na biometria não apenas melhora a segurança e a precisão, mas também torna a tecnologia mais acessível e adaptável a diferentes contextos e necessidades (Zhou ; Zhao, 2022).

Os pilares da autenticação biométrica se dividem entre esses conceitos descritos pelos autores Zhou e Zhao (2022):

- Universalidade: todos humanos devem possuir;
- Unicidade: capazes de diferenciar duas pessoas;
- Constância: se manterem mesmo com alterações naturais do organismo;
- Coletabilidade: devem ser fáceis de coletar em um curto espaço de tempo;
- Performance: a diferenciação deve ser possível de ser identificada em tempo hábil;
- Segurança: devem ser difíceis de se copiar.

Esses pontos são fundamentais para a discussão sobre a eficácia e a viabilidade da biometria como um método de autenticação em diversos contextos.

2.4 Reconhecimento facial

Os primeiros registros de trabalhos voltados para o reconhecimento facial começaram a surgir nas décadas de 1950 e 1960, mas é considerado que a

investigação sobre reconhecimento automático de rostos foi iniciada em meados de 1970. Os trabalhos que surgiram nessa década tinham a característica de usar a distância entre as regiões importantes da face. Em 1990, com a crescente evolução de hardware e da necessidade de aplicações relacionadas à segurança, foi onde os estudos de investigação sobre esse método cresceram (Taskiran; Kahraman; Erdem, 2020).

Reconhecimento facial tem sido uma técnica muito investigada nos últimos anos devido a suas várias aplicações que podem ter como exemplo segurança de fronteiras, aplicação da lei e controle de acesso. O reconhecimento e um dos sistemas de biometricos mais complexos usados na área de reconhecimento de padrões isso se deve pois a técnica registra um sucesso notável em ambientes controlados mas em ambientes sem restrições podem falhar uma vez que o processo de aquisição da face pode sofrer uma grande variedade de variações como iluminação, variação de pose, expressões, baixa resolução e mais (Oloyede; Hancke; Myburgh, 2020).

O processo de reconhecimento facial tem semelhança com outros sistemas de biometricos, as fases envolvidas são detecção da face, pré-processamento da imagem da face, a extração das características físicas e por último a classificação das características. Cada uma dessas fases tem funções específicas para compor o reconhecimento de uma face e cada uma possui técnicas e objetivos diferentes (Taskiran; Kahraman; Erdem, 2020).

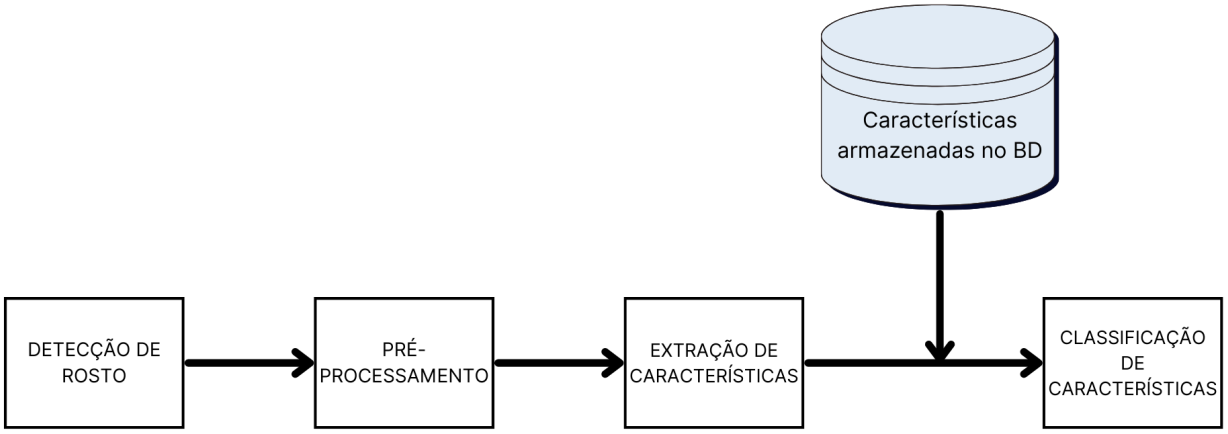
A primeira fase que é a detecção da face que consiste no programa rastrear a presença de uma face no vídeo ou imagem, após a detecção a imagem é pré-processada nessa fase e é onde o programa mapeia a região de interesse e também melhora a qualidade da imagem, onde acontecem processos conhecidos como normalização que consiste em normalizar diferentes escalas da imagem para a mesma escala, com isso o algoritmo de reconhecimento pode operar de maneira mais efetiva, nesta etapa também acontece o alinhamento responsável pelo processo de localizar pontos fiduciais, partes importantes como boca, a pálpebra e o nariz, essa abordagem melhora significativamente a identificação, o melhoramento da imagem por mais que algumas vezes negligenciado é fundamental para obter uma imagem melhorada a partir da original, que consequentemente vai melhorar o desempenho global do sistema.

A extração de características e a segunda fase de um modelo de reconhecimento tem o objetivo de diminuir ou simplificar o número de recursos que descrevem um grande conjunto de dados, além disso essa extração é feita para minimizar o ruído e remover informações irrelevantes na imagem original, é extraído a parte importante das características que o suficiente para descrever uma face através da imagem.

A última fase e a classificação que e responsável tanto por classificar como identificar, a fase de classificação de características que conduz ao reconhecimento de imagens de rostos que envolve tanto a autenticação quanto a identificação de imagens de rostos, a identificação ocorre quando o algoritmo compara uma imagem de face com outras imagens em um banco de dados com objetivo de encontrar alguma face que corresponda corretamente entre as várias possibilidades, enquanto

a autenticação ocorre quando o algoritmo de reconhecimento compara a face encontrada na identificação com a imagem da face para aprovar a identidade solicitada. Em ambos os cenários, as imagens de rostos dos indivíduos reconhecidos são armazenadas num sistema como o de uma galeria, depois disso as imagens de rostos classificadas e podem ser divididas em duas classes: faces registradas ou não registradas, essas informações são usadas para a tarefa de identificação ou reconhecimento, na Figura 1 é possível ver o fluxo completo das etapas que compõem um sistema de reconhecimento facial.

Figura 1. Diagrama de fluxo das etapas envolvidas em um sistema de reconhecimento facial.



Fonte: Adaptado de (Oloyede; Hancke; Myburgh, 2020).

Grande parte dos algoritmos de reconhecimento facial se baseiam nos passos descritos acima, um dos algoritmos mais utilizados nos dias de hoje é o *Haar feature-based cascade*, esse foi proposto por Paul Viola e Michael Jones (2001) no artigo intitulado "*Rapid Object Detection using a Boosted Cascade of Simple Features*".

O algoritmo Viola-Jones começa com a entrada de uma imagem em escala de cinza, que pode ter qualquer dimensão, mas é projetada para trabalhar com janelas de 24x24 *pixels*. Em seguida, a imagem é pré-processada, e é introduzido o conceito de "Imagem Integral". Este conceito permite calcular rapidamente a soma dos *pixels* em uma região retangular da imagem. Para cada *pixel* na imagem integral, o valor armazenado é a soma de todos os *pixels* acima e à esquerda desse *pixel*, incluindo o próprio.

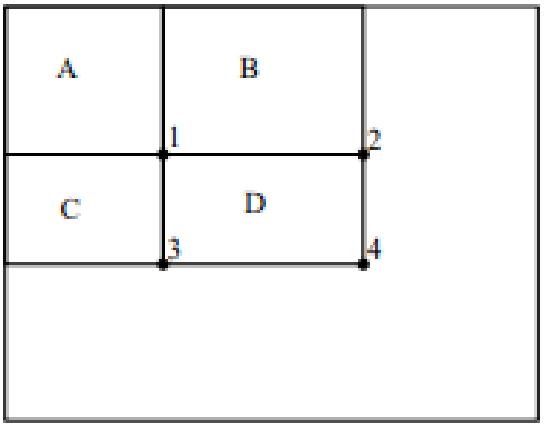
Depois disso, o algoritmo define janelas de 24x24 *pixels* que se movem por toda a imagem. Em cada uma dessas janelas, são avaliadas as características de Haar, que são baseadas nas diferenças de intensidade entre regiões retangulares adjacentes. Essas características capturam informações sobre a presença de bordas, texturas e outras estruturas visuais simples.

O próximo passo envolve o uso de classificadores em cascata para eliminar rapidamente as janelas que não contêm rostos. As janelas descartadas não são processadas mais a fundo, economizando tempo de computação. As janelas que passam por essa primeira etapa são processadas por uma série de classificadores

em cascata, onde cada classificador é mais complexo que o anterior, refinando a detecção entre rostos e não-rostos.

Por fim, as janelas que permanecem após todos os passos são consideradas como contendo rostos, resultando em uma imagem com os rostos destacados, geralmente por retângulos. Além disso, o algoritmo pode fornecer informações sobre a posição e o número de rostos detectados (Viola; Jones, 2001). A Figura 2 demonstra como são as janelas que se movem pela imagem.

Figura 2. Demonstração da definição das janelas utilizando o algoritmo desenvolvido por Viola e Jones.



Fonte: Viola; Jones, (2001).

Outro algoritmo muito conhecido foi criado por Navneet Dalal e Bill Triggs (2005) no artigo intitulado de “Histograms of Oriented Gradients for Human Detection”, ele apresenta uma abordagem diferente no algoritmo Viola-Jones, o algoritmo também conhecido por HOG, utiliza uma metodologia de histogramas de gradiente orientados para conseguir descrever a forma e aparência de objetos dentro e uma imagem, contribuindo para o desenvolvimento de métodos de detecção robustos.

O algoritmo HOG começa com a entrada de uma imagem, que pode conter rostos em diversas poses e condições. A imagem é pré-processada, o que pode incluir o redimensionamento e a conversão para escala de cinza. Em seguida, durante a extração de características, a imagem é dividida em pequenas células. Em cada célula, o algoritmo calcula o gradiente de intensidade, que fornece informações sobre a direção e a magnitude das bordas. O resultado dessa análise é um vetor de características que representa a distribuição das orientações de borda. Esses vetores são normalizados para reduzir a sensibilidade a variações de iluminação e melhorar a robustez da extração de características, formando assim um vetor que representa a imagem como um todo.

Na etapa de classificação, o vetor de características é alimentado em um classificador SVM (do inglês, *Support Vector Machine*). O SVM é uma técnica de aprendizado de máquina que separa diferentes grupos de dados ao traçar a melhor

linha (ou fronteira) que os mantém afastados. O classificador SVM é previamente treinado com imagens de rostos e de não-rostos, o que permite que ele determine se um rosto está presente na imagem analisada. O algoritmo HOG também pode utilizar janelas deslizantes para detectar rostos em várias escalas e posições, retornando a localização dos rostos na imagem, geralmente representada por caixas delimitadoras. Além disso, pode fornecer uma pontuação indicando a probabilidade de acerto da detecção. Apesar das semelhanças entre o algoritmo HOG e o Viola-Jones, a principal diferença entre eles reside na maneira como eles detectam e representam características visuais para a detecção de objetos, como rostos ou pessoas.

Os algoritmos citados acima trabalham bem na fase de detecção de rostos, e, quando falamos sobre extração de características, outro passo crucial para o reconhecimento facial, surgem outros algoritmos importantes. Os principais, conhecidos como Eigenfaces e Fisherfaces. Ambos seguem a *holistic approach*, que trata o rosto como um todo, utilizando a imagem completa para o reconhecimento. No entanto, existem também abordagens híbridas (*hybrid approach*), que combinam as técnicas *holistic* e *feature-based* para capturar tanto as características globais quanto as locais do rosto, oferecendo uma solução mais robusta e precisa para o reconhecimento facial em condições variadas.

Após a consolidação de métodos clássicos como Viola-Jones (baseado em *Haar cascades*) e HOG (*Histogram of Oriented Gradients*), que dependem da extração manual de características e são eficazes apenas em ambientes controlados, as Redes Neurais Convolucionais (do inglês, *Convolutional Neural Networks* - CNNs) surgiram como um avanço revolucionário, superando limitações como variações de iluminação e pose. Um trabalho importante para esse cenário é o de Schroff, Kalenichenko e Philbin (2015). Nele, os autores propõem uma arquitetura de CNN que mapeia imagens faciais diretamente em um espaço euclidiano de alta dimensão, onde a distância entre *embeddings*[1] reflete a similaridade facial. Diferentemente de técnicas tradicionais, que dependem de redução linear de dimensionalidade e são sensíveis a variações não lineares, o FaceNet utiliza uma rede convolucional profunda treinada com *triplet loss*, otimizando a representação das características faciais em vez de camadas intermediárias. Essa abordagem permite que o sistema alcance 99,63% de acurácia no *benchmark* LFW (*Labeled Faces in the Wild*) e 95,12% no YouTube Faces DB, superando métodos anteriores (Schroff et al., 2015).

A superioridade do FaceNet reside na sua capacidade de generalização, enquanto Eigenfaces e Fisherfaces exigem condições ideais (*e.g.*, iluminação uniforme, poses frontais), as CNNs aprendem representações hierárquicas diretamente dos *pixels*, capturando padrões complexos como texturas, contornos e relações espaciais. Isso as torna robustas mesmo em cenários dinâmicos, como os de um campus universitário, onde a diversidade de ambientes e a necessidade de processamento em tempo real são críticas.

[1] Embeddings são representações vetoriais numéricas de informações complexas, como imagens ou palavras, projetadas em um espaço de características.

2.5 Bibliotecas de reconhecimento facial

A OpenCV é uma biblioteca de código aberto consolidada na área de visão computacional, originalmente desenvolvida pela Intel em 1999. Com mais de duas décadas de evolução, ela reúne algoritmos otimizados para tarefas como processamento de imagens, captura de vídeo em tempo real, detecção de objetos e integração transparente com modelos de deep learning. Sua arquitetura é projetada para baixa latência, sendo uma característica interessante para sistemas que exigem processamento contínuo de *frames*.

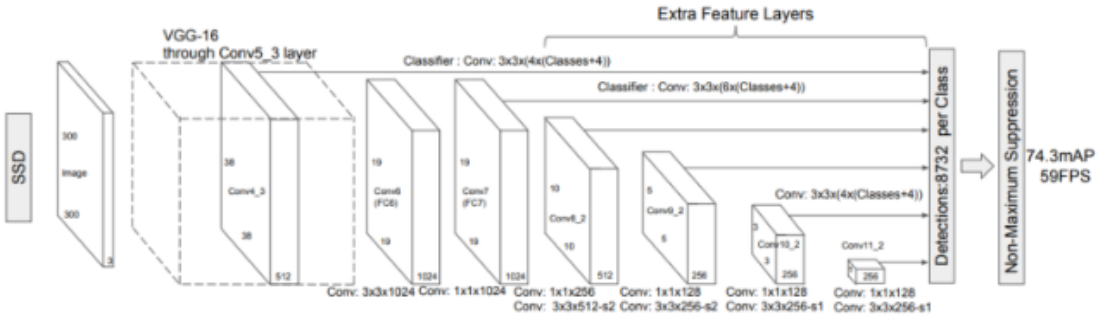
Além da eficiência, a OpenCV destaca-se pela versatilidade: oferece suporte nativo a Python e interfaces com *frameworks* populares, como o TensorFlow, e o PyTorch, simplificando a implementação de modelos pré-treinados em *pipelines* de visão computacional. Essa flexibilidade, aliada a uma comunidade ativa e documentação robusta (Bradski; Kaehler, 2008), com essas características tornou-se uma escolha adequada para esse artigo que demanda rapidez e confiabilidade.

Para o algoritmo de detecção facial, optou-se pelo Res10-SSD (*Single Shot MultiBox Detector* com ResNet-10), um modelo pré-treinado que integra duas arquiteturas complementares: a ResNet-10 e o SSD (*Single Shot MultiBox Detector*). A ResNet-10 é uma variante compacta da Residual Network, que emprega conexões residuais mecanismos que permitem "atalhos" entre camadas da rede neural para mitigar o problema de degradação de gradiente em redes profundas, garantindo treinamento estável mesmo com poucas camadas. Já o SSD destaca-se por unificar as etapas de localização e classificação de objetos em uma única passagem pela rede, o que reduz significativamente a latência do sistema.

Essa combinação permite ao Res10-SSD operar com eficiência computacional em tempo real, processando imagens redimensionadas para 300x300 *pixels*, uma resolução que equilibra carga de processamento e retenção de detalhes críticos para detecção facial. O modelo foi treinado no dataset WIDER FACE, que contém mais de 32 mil imagens de rostos em condições desafiadoras, como oclusões, variações bruscas de iluminação e múltiplos ângulos de captura. Essa diversidade de dados confere ao sistema robustez para cenários dinâmicos, com taxa de detecção de 84% para rostos pequenos (Yang *et al.,* 2016).

A escolha do Res10-SSD está na sua capacidade de harmonizar velocidade e precisão, requisitos críticos para aplicações de segurança. Por um lado, a arquitetura single shot do SSD assegura baixa latência (Liu *et al.,* 2016), viabilizando processamento contínuo de fluxos de vídeo. Por outro, as conexões residuais da ResNet-10 preservam *features* faciais relevantes mesmo em redes neurais compactas, assegurando precisão em condições adversas, como mudanças súbitas de iluminação ou movimento rápido. Além disso, sua eficiência em consumo de memória GPU (até 40% menor que modelos similares) torna-o adequado para hardware moderado, como sistemas embarcados ou dispositivos sem aceleradores dedicados, ampliando sua aplicabilidade prática. Na Figura 3 pode-se ver a arquitetura original do modelo *Single Shot MultiBox Detector*.

Figura 3. Exemplo das camadas do modelo SSD, desde a entrada da imagem até a detecção.



Fonte: Adaptado de (Liu *et al*., 2016)

A biblioteca *face_recognition* foi selecionada para o reconhecimento facial devido à sua implementação eficiente de algoritmos baseados em redes neurais profundas, capazes de gerar *embeddings* faciais representações vetoriais que codificam características únicas do rosto em um espaço multidimensional. Esses embeddings são gerados por uma rede convolucional pré-treinada, inspirada no trabalho seminal de Schroff *et al*. (2015) com o FaceNet, que alcança mais de 99% de acurácia em *benchmarks* consolidados, como o *Labeled Faces in the Wild* (LFW). A escolha da biblioteca tem a fácil integração com o OpenCV, além de sua reputação em oferecer alta precisão em comparações faciais.

3 METODOLOGIA

Este trabalho é constituído por três abordagens: técnica, experimental e aplicada. Inicialmente, foi realizada uma pesquisa de literatura sobre o tema, com o objetivo de identificar os principais algoritmos utilizados para reconhecimento facial e sua implementação em Python. Na vertente experimental, esses algoritmos foram de fato executados e avaliados em termos de desempenho. Por fim, na abordagem aplicada, desenvolveu-se um protótipo funcional que integra detecção e reconhecimento em um fluxo real de controle de acesso no campus.

Para desenvolver o sistema de reconhecimento facial, com o intuito de ser aplicado no controle de acesso do campus Paulista. O software foi implementado utilizando Python 3.9, com bibliotecas de Visão computacional e Aprendizado de máquina como *OpenCV* e *face_recognition*.

3.1 Seleção de Algoritmos

A primeira etapa do trabalho consistiu em construir uma revisão bibliográfica abrangente, com o objetivo de identificar os algoritmos mais adequados para a detecção e o reconhecimento em tempo real. Os artigos foram selecionados e analisados no período de (2020 - 2024) que abordaram aplicações similares. A busca desses artigos tiveram a função de buscar informações sobre as principais bibliotecas e técnicas de reconhecimento usadas e as diferentes abordagens para cada tecnologia.

Como métrica de similaridade, adotou-se a distância cosseno, que mede o ângulo entre dois vetores de *embeddings*, garantindo independência de variações de

escala (*e.g.*, brilho da imagem). O *threshold* definido em 0,55, quanto menor a distância, maior a similaridade, foi ajustado empiricamente durante testes preliminares, equilibrando sensibilidade, detecção de rostos autorizados e redução de falsos positivos. Essa configuração prioriza a segurança em ambientes críticos, onde a identificação errada de intrusos é inaceitável.

3.2 Interface e sistema

O sistema desenvolvido para ser operado por porteiros ou responsáveis pela segurança do campus, dispõe de uma interface gráfica simples e intuitiva criada com Python 3.0 e a biblioteca PyQt5. No menu principal, o usuário pode abrir o formulário de cadastro para inserir nome, matrícula e imagem facial de novos alunos; iniciar o sistema, acionando a captura contínua de vídeo para detecção e reconhecimento em tempo real; remover alunos, eliminando registros do banco de dados e atualizando a lista de autorizados; e, por fim, sair, encerrando a aplicação de forma segura e salvando todas as alterações realizadas.

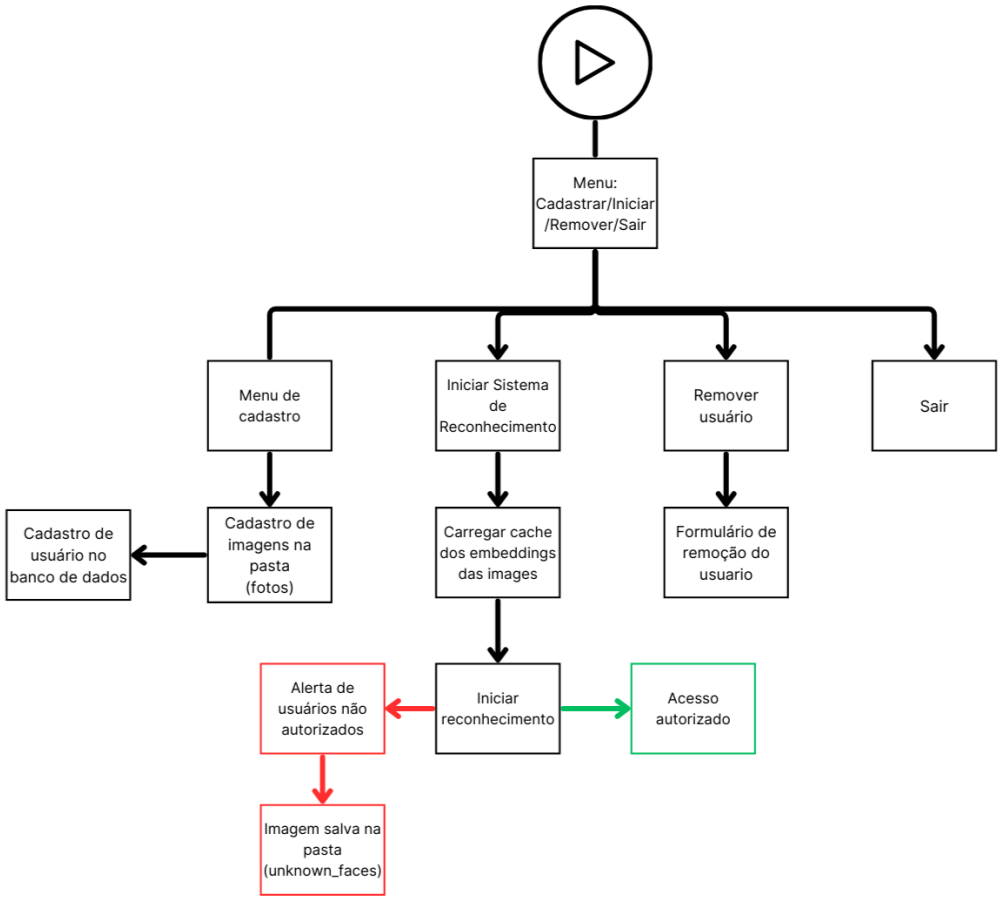
As informações dos alunos são gerenciadas em um banco de dados SQLite, escolhido por sua portabilidade e eficiência em aplicações de pequeno a médio porte. As imagens faciais são armazenadas em uma pasta dedicada no sistema operacional (*known_faces*), na pasta em questão para aumentar o grau de dificuldade.

Para os experimentos realizados, a base de fotos foi criada com 250 imagens aleatórias extraídas do dataset 1 Million fake faces[2]. Esse procedimento exigiu que o algoritmo detectasse e identificasse, em paralelo, rostos pertencentes à base (“conhecidos”) e rostos não cadastrados (“desconhecidos”) em um único *frame*, validando sua robustez em cenários com múltiplos sujeitos e alto nível de ruído visual. Em caso de detecção de rostos não autorizados, o sistema emite alertas visuais e sonoros, garantindo que o usuário seja notificado imediatamente. Simultaneamente, o *frame* contendo o rosto desconhecido é salvo na pasta criada no sistema operacional chamada de *unknown_faces*. O código-fonte completo do sistema, incluindo a interface gráfica e os scripts de detecção, está disponível publicamente no repositório GitHub[3].

Existe também uma implementação de cache no sistema para agilizar o início diário da aplicação. Nessa memória, os cálculos dos embeddings são armazenados em formato binário, acelerando as comparações durante o reconhecimento. Quando novos alunos são cadastrados ou alguma face é alterada, o sistema compila novamente o cache automaticamente. Para informar o usuário sobre esse processo que pode ser demorado , uma barra de progresso foi implementada. A Figura 6 demonstra o fluxo completo do sistema e seu comportamento.

Figura 6. Fluxo completo do sistema desenvolvido neste trabalho.

[2] Disponível em https://www.kaggle.com/datasets/tunguz/1-million-fake-faces-7.
[3] Disponível em https://github.com/GabrielHenriquedev/Face_recoginition_if.



Fonte: Próprio Autor.

3.3 Métricas de avaliação

As métricas de avaliação foram escolhidas para analisar o desempenho do sistema, algumas métricas conhecidas foram usadas para avaliar o sistema, como confiança e *frames* por segundo, dessa forma garantindo a robustez e podendo gerar um relatório detalhado para analisar o algoritmo e sua performance de acordo com o desafio, portanto modelo de reconhecimento e avaliado por meio de meios que quantificam sua precisão, eficiência e confiabilidade. As métricas foram selecionadas com base em estudos anteriores de reconhecimento facial (Oloyede; Hancke; Myburgh, 2020; Feng *et al.*, 2022) .

Originalmente planejava-se utilizar métricas clássicas de classificação como acurácia, precisão, entretanto durante a fase de testes iniciais foi identificada inconsistência no cálculo automático de acurácia e falsas detecções, pois o sistema não tem uma base real para comparação é devido a o código não contabilizar o erros onde rostos desconhecidos são classificados como conhecidos, assim acabou inflando artificialmente a acurácia, motivando a adoção de métricas mais diretas e facilmente interpretáveis para avaliar a detecção em tempo real. Em aplicações de segurança em tempo real, métricas operacionais como FPS (*Frame Per Second*) e confiança média foram utilizadas para avaliar eficiência operacional, garantindo que o sistema funcione sem atrasos e com decisões confiáveis, mesmo em condições

dinâmicas (Oloyede *et al.,* 2020). Na abela 1 são mostradas as métricas que foram utilizadas e o objetivo de cada.

Tabela 1. Métricas adotadas para avaliação do sistema.

Métrica	**Definição**
FPS (*Frames/s*)	Mede quantos quadros, que são imagens, o sistema processa por segundo. É crítica em aplicações em tempo real, garantindo fluidez e resposta imediata.
Confiança média	Quantifica a certeza das identificações, calculada a partir da distância cosseno entre embeddings faciais, representação matemática do rosto.
Faces detectadas	Quantidade total de faces, conhecidas e desconhecidas capturadas durante o teste.
Faces conhecidas	Número de rostos reconhecidos como “autorizados”.
Faces desconhecidas	Rostos detectados que não constavam na base de autorizados.

A confiança no sistema é avaliada por meio de duas abordagens complementares. A confiança média histórica funciona como um indicador global da eficácia do sistema, resultando da média de todas as tentativas de reconhecimento ao longo da execução. Já a confiança instantânea representa a precisão pontual de cada reconhecimento individual, refletindo o grau de similaridade entre o rosto detectado e os registros da base de dados. A confiança média, portanto, é calculada a partir das confianças instantâneas e sofre influência direta do *threshold* definido e da distância entre os *embeddings* faciais. A confiança instantânea é obtida a partir da distância entre embeddings, utilizando a relação confiança = 1 - distância. O processo de reconhecimento pode ser visto em um exemplo na Tabela 2.

Tabela 2. Métricas adotadas para avaliação do sistema.

embeddings	Confiança	Threshold (0,55)	Resultado
0,40	0,60	< 0,55	Reconhecido
0,55	0,45	== 0,55	Desconhecido
0,60	0,40	> 0,55	Desconhecido

O FPS garante que o sistema processe fluxos de vídeo sem atrasos perceptíveis, já a proporção de faces conhecidas/desconhecidas valida a capacidade do sistema de filtrar acessos indesejados, essencial para ambientes como o campus IFPE Paulista.

4 Resultados

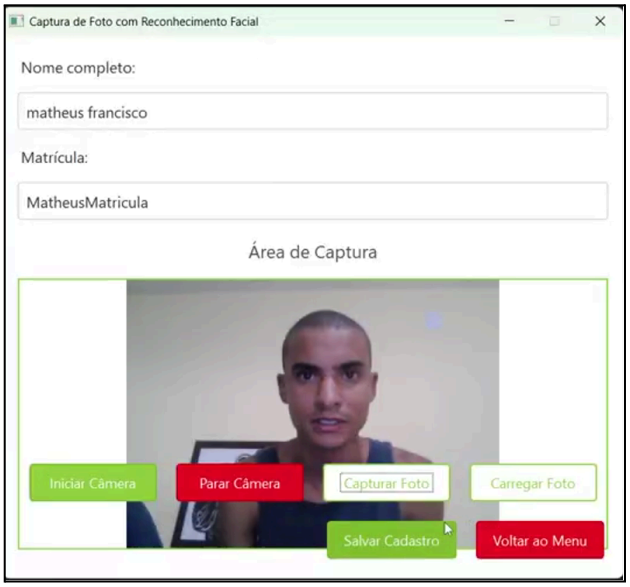
Nesta seção apresentam-se os resultados de um ensaio controlado projetado para avaliar a robustez do sistema ao reconhecer duas faces-alvo simultaneamente capturadas pela *webcam*. Durante o teste, foram exibidos, em um único *frame*, rostos já cadastrados e rostos não registrados, exigindo que o algoritmo realizasse, em paralelo, a detecção e a identificação de ambas as categorias. Esse procedimento permitiu verificar o comportamento do sistema em cenários com múltiplos sujeitos, avaliando sua precisão de rotulação e estabilidade de desempenho sob condições de maior complexidade visual.

O primeiro passo consiste no cadastro do rosto do aluno é realizado pelo gestor do sistema por meio de interface gráfica em PyQt5. Ao selecionar a opção “Abrir formulário” como mostrado na Figura 6, é exibido um formulário que solicita nome completo e matrícula, como mostrado na Figura 7, que serão utilizados como rótulos para identificação futura. Em seguida, o gestor pode optar pela captura da imagem facial diretamente pela câmera ou pelo upload de um arquivo de imagem armazenado localmente. Após a captura ou seleção do arquivo, o sistema apresenta uma pré-visualização da imagem, permitindo ajuste de enquadramento e confirmação antes do envio. Concluída a validação visual, a foto é redimensionada e submetida a um algoritmo de detecção facial para garantir que apenas o rosto seja armazenado. Logo após essa verificação, o registro com os dados do aluno é gravado no banco de dados SQLite, e o arquivo de imagem é salvo na pasta de “Fotos”, nomeado com a matrícula, o *cache* é automaticamente atualizado quando algo na pasta e atualizado, seja um novo cadastro ou a remoção de algum rosto. Por fim, a interface exibe uma mensagem de sucesso, indicando que o novo usuário está pronto para autenticações subsequentes.

Figura 6. Exemplo do menu inicial do programa desenvolvido de reconhecimento facial.

Fonte: Próprio Autor.

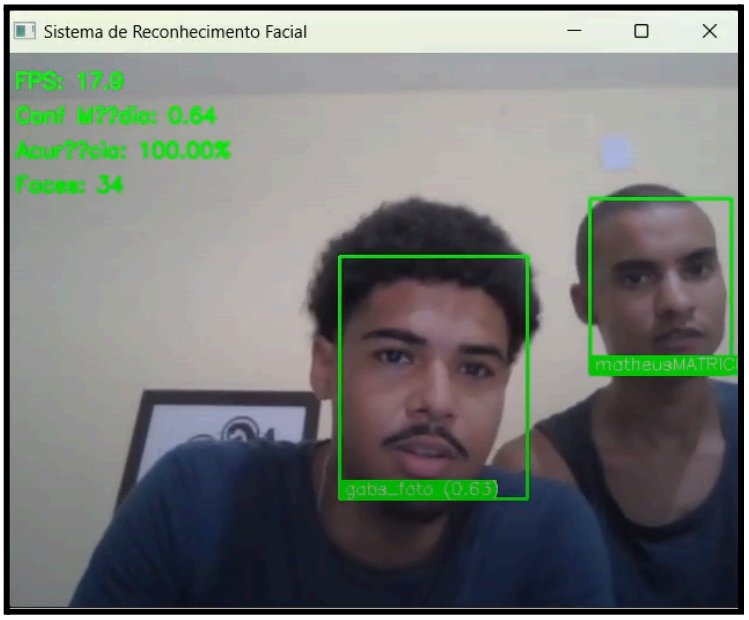
Figura 7. Formulário de cadastro do usuário no programa desenvolvido.



Fonte: Próprio Autor.

Após o cadastro do novo usuário, iniciou-se o teste com a apresentação isolada de um rosto conhecido, seguido pelo rosto recentemente inserido no sistema. Em seguida, duas faces registradas foram posicionadas simultaneamente em frente à câmera, e o algoritmo os reconheceu com êxito, como mostrado na Figura 8. Observou-se, entretanto, dificuldade na rotulação, imprimir corretamente qual era o nome do rosto reconhecido, possivelmente devido a semelhanças faciais ou variações de iluminação; após alguns segundos, porém, o sistema distinguiu corretamente cada indivíduo, após um tempo da execução.

Figura 8. Imagem do sistema identificando rostos conhecidos corretamente.



Fonte: Próprio Autor.

No teste subsequente, um rosto não cadastrado foi exibido isoladamente e identificado com sucesso como “desconhecido”. Por fim, três sujeitos, dois “conhecidos” e um “desconhecido” foram apresentados em um único quadro. Após breve demora na rotulação precisa dos usuários autorizados, o sistema detectou o intruso e salvou automaticamente o *frame* na pasta de “rostos desconhecidos”. Todos os experimentos foram conduzidos em ambiente controlado, com o objetivo de demonstrar a conformidade do protótipo aos requisitos de detecção e classificação propostos, as métricas do ensaio estão descritas na Tabela 3.

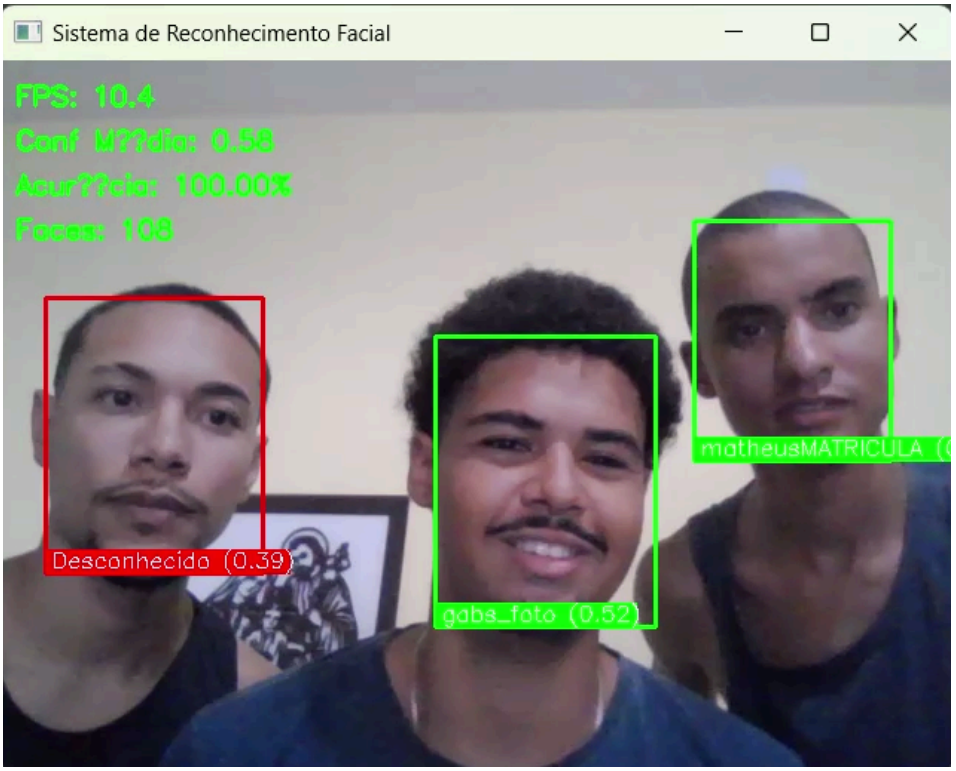
Tabela 3. Métricas de Performance do Experimento realizado utilizando rostos.

Métrica	**Valor**
Tempo de execução	91,99 s
FPS Mínimo	17,44
FPS Máximo	31,00
Confiança média	0,5779
Faces detectadas	120
Faces conhecidas	2
Faces desconhecidas	1

O FPS variou entre 17,44 e 31,00 quadros por segundo, em função do ciclo de processamento configurado em 0,75 s para detecção e 1,5 s para reconhecimento facial. Essa periodicidade reduz o consumo de recursos quando o sujeito permanece em cena, permitindo ao hardware utilizado Ryzen 7 7500U, 12 *gigabytes* de memória ram e ssd de 256 gb, manter desempenho aceitável, ainda que apresente limitações sob maiores cargas de rostos simultâneos. A confiança média atingiu 0,578, valor superior ao limiar de 0,5 e consistente com detecções confiáveis mesmo em condições de iluminação não uniforme, vale destacar que a confiança é um termômetro de similaridade, não um indicador absoluto de reconhecimento. O sistema só afirma "conhecer" um rosto quando a similaridade ultrapassa o *threshold* definido de 0,55. Valores baixos indicam apenas que nenhum rosto conhecido foi encontrado dentro do limite de tolerância configurado, a confiança imediata variou entre 0,63 e 0,71 demonstram que, em várias detecções, a similaridade ficou bem acima do limite configurado, evidenciando uma boa margem de segurança e aceitabilidade nas decisões de autenticação. No total, foram detectadas 120 faces durante o ensaio, o número alto de faces detectadas se dá, pois as faces são calculadas a cada ciclo de reconhecimento (intervalo de 0,75) dado a isso o número elevado de faces, das quais 2 foram corretamente

identificadas como “conhecidas” e 1 como “desconhecida”, é importante apontar que o teste foi realizado em ambiente controlado e com uma limitação de apenas 3 faces. O teste realizado é mostrado na Figura 9, onde se pode ver os rostos corretamente rotulados e as informações das estatísticas do sistema[4].

Figura 9. Imagem do funcionamento do teste de reconhecimento.



Fonte: Próprio Autor.

Os ensaios foram realizados em ambiente controlado, com no máximo três rostos simultâneos, o que limita os resultados para cenários de maior complexidade, já que em ambientes abertos existe muito mais influência de iluminação, movimentação e pessoas. Observou-se dificuldade na rotulação de rostos cadastrados, possivelmente causada pela elevada similaridade entre os sujeitos ou por variações de iluminação; entretanto, após alguns segundos de processamento, o algoritmo estabilizou e identificou corretamente cada indivíduo. Pode ser visto também que o FPS apresentou oscilações consideráveis à medida que o número de rostos em cena aumentava, revelando um potencial gargalo computacional em situações de maior quantidade de faces, mas o sistema se mostrou capaz de mostrar um resultado rápido e devidamente preciso mesmo com essa lentidão.

Para uma validação mais robusta do sistema, fica a recomendação da execução de testes suplementares em cenários mais desafiadores (baixa luminosidade, movimentação de múltiplos sujeitos, uso de câmeras externas). Do ponto de vista de infraestrutura, a adoção de uma GPU dedicada via CUDA

[4] Um vídeo do experimento realizado está disponível em https://www.youtube.com/watch?v=RR8o4J1tXT8.

(*Compute Unified Device Architecture*) poderia diminuir o uso da CPU e elevar o FPS, ampliando a viabilidade do sistema para operações em tempo real. Vale a pena ressaltar que o protótipo atende aos requisitos básicos de controle de acesso, mas carece de validação em escala real. A combinação de ajustes algorítmicos, expansão do *dataset* e infraestrutura adequada pode transformar a solução em uma ferramenta robusta para a segurança do *campus*, alinhada às demandas tecnológicas contemporâneas.

5 Conclusão

Este trabalho apresentou a implementação de um sistema de reconhecimento facial para controle de acesso no IFPE *campus* Paulista, com o propósito de reforçar a segurança em ambientes educacionais. Baseado em algoritmos de *deep learning*, Res10-SSD para detecção e FaceNet para geração de embeddings e desenvolvido em Python com OpenCV e *face_recognition*, o protótipo mostrou-se capaz de operar em tempo real, variando de 17,44 a 31 FPS, com confiança média de 0,578.

Foram empregadas exclusivamente ferramentas gratuitas e de código aberto, de modo a minimizar custos futuros de manutenção. As funcionalidades principais incluem: (i) cadastro de usuários (i.e. alunos); (ii) detecção e identificação de indivíduos não cadastrados; e (iii) emissão de alertas diante de tentativas de acesso indevido.

Os ensaios realizados em ambiente controlado confirmaram a eficácia do sistema em reconhecer indivíduos na base e desconhecidos. No entanto, oscilações de FPS sob carga moderada e dificuldades iniciais na rotulação de sujeitos com traços semelhantes evidenciam a necessidade de aprimorar o pré-processamento de imagens e diversificar a base de faces cadastradas, para garantir precisão igualitária em diferentes etnias, idades e gêneros. Ademais, a falta de testes em condições reais de baixa luminosidade, multidões, câmeras externas limita a generalização dos resultados.

Em síntese, o protótipo satisfaz os requisitos básicos de controle de acesso, mas sua maturidade e escalabilidade dependem de fatores como investimentos em infraestrutura e refinamento algorítmico. A proposta alinha-se às demandas atuais por soluções de segurança inteligente, apontando caminhos promissores para reduzir riscos e proteger a comunidade acadêmica.

Como trabalhos futuros, fica sugerido a integração de aceleração em GPU para elevar o FPS, o ajuste dinâmico de *thresholds* de confiança e janelas de detecção para otimização de recursos, a ampliação da base de usuários com diversidade étnica, etária e de expressões e o desenvolvimento de protocolos de privacidade e segurança para armazenamento e transmissão de dados biométricos. Dessa forma, espera-se estender o protótipo a um sistema de controle de acesso escalável e confiável, apto a operar em instalações de grande porte como o campus IFPE Paulista.

REFERÊNCIAS

BBC NEWS BRASIL. **O que se sabe sobre o ataque a tiros que matou 14 pessoas em universidade em Praga.** 22 dez. 2023. Disponível em: https://www.bbc.com/portuguese/articles/c2jy4z81j5lo. Acesso em: 19. jun. 2024

BRADSKI, G.; KAEHLER, **A. Learning OpenCV: Computer vision with the OpenCV library.** Sebastopol: O'Reilly Media, Inc, 2008. ISBN: 978-0-596-51613-0

FENG, Y. et al. Detect Faces Efficiently: A Survey and Evaluations. IEEE **Transactions on Biometrics, Behavior, and Identity Science**, v. 4, n. 1, p. 1–18, jan. 2022. Doi: 10.1109/TBIOM.2021.3120412

LI, L. et al. A Review of Face Recognition Technology. **IEEE Access**, v. 8, p. 139110–139120, 21 jul. 2020.

LIU, W.; ANGUELOV, D.; ERHAN, D. et al. SSD: Single Shot MultiBox Detector. In: **EUROPEAN CONFERENCE ON COMPUTER VISION (ECCV)**, 14. Amsterdam: Springer, Cham. 2016. p. 21-37.

MAKHSUD, U. Identification and Authentication. **International Journal Of Academic Pedagogical Research (IJAPR)**, p. 39, jan. 2021. ISSN: 2643-9123

MANDRU, SRIKANTH. How AI can improve identity verification and access control processes. **Journal of Artificial Intelligence & Cloud Computing,** v. 1, n. 4, p. 1-5, dez. 2022.

OLOYEDE, M. O.; HANCKE, G. P.; MYBURGH, H. C. **A review on face recognition systems: recent approaches and challenges.** Multimedia Tools and Applications, v. 79, n. 37–38, p. 27891–27922, jul. 2020.

SANTOSO, W.; SAFITRI, R.; SAMIDI, S. Integration of Artificial Intelligence in Facial Recognition Systems for Software Security. **Sinkron : jurnal dan penelitian teknik informatika**, v. 8, n. 2, p. 1208–1214, 30 abr. 2024.

SCHROFF, F.; KALENICHENKO, D.; PHILBIN, J. FaceNet: A unified embedding for face recognition and clustering. In: 2015 **IEEE Conference on Computer Vision and Pattern Recognition** (CVPR), jun. 2015.

TASKIRAN, M.; KAHRAMAN, N.; ERDEM, C. E. Face recognition: Past, present and future (a review). **Digital Signal Processing**, v. 106, p. 102809, nov. 2020.

VINHA, TELMA, et al. **Ataques de Violência Extrema Em Escolas No Brasil Causas E Caminhos.** 1. ed. São Paulo: D3e, 2023. ISBN: 978-65-995856-8-5

VIOLA, P.; JONES, M. Rapid Object Detection using a Boosted Cascade of Simple Features. **Conference on Computer Vision and Pattern Recognition**, 2001.

YUSUF, NUHU; MARAFA, Kamalu Abdullahi; SHEHU, Kamila Ladan; MAMMAN, Hussaini; MAIDAWA, Mustapha. A survey of biometric approaches of authentication. **International Journal of Advanced Computer Research**, v. 10, n. 47, p. 96–104, 2020.

ZHOU, F.; ZHAO, T. **A survey on biometrics authentication**. arXiv, 15 dez. 2022.

ZULKARNAIN, S.; et al. A review on authentication methods. **Australian Journal of Basic and Applied Sciences**, v. 7, n. 7, p. 95–107, 2013.