



UNIVERSIDADE FEDERAL DO MARANHÃO

Coordenação do Curso de Engenharia da Computação

VISÃO COMPUTACIONAL PARA SEGURANÇA PÚBLICA

Nome do aluno: Lucas Rodrigues Ferreira

Orientador: Wener Borges de Sampaio

São Luís

2020



LUCAS RODRIGUES FERREIRA

VISÃO COMPUTACIONAL PARA SEGURANÇA PÚBLICA

Trabalho de Contextualização e Integralização Curricular II, apresentado a Coordenação do Curso de Engenharia da Computação da Universidade Federal do Maranhão, para obtenção do grau de Bacharel Engenheiro da Computação.

Orientador: Wener Borges de Sampaio

São Luís

2020



LUCAS RODRIGUES FERREIRA

VISÃO COMPUTACIONAL PARA SEGURANÇA PÚBLICA

Trabalho de Contextualização e Integralização Curricular II, apresentado a Coordenação do Curso de Engenharia da Computação da Universidade Federal do Maranhão, para obtenção do grau de Bacharel Engenheiro da Computação.

Aprovado em: / /

BANCA EXAMINADORA

Prof. Dr. Wener Borges de Sampaio (Orientador)

Universidade Federal do Maranhão

Prof. Dr. Sérgio Souza Costa

Universidade Federal do Maranhão

Prof. Msc. Cláudio Manoel Pereira Aroucha

Universidade Federal do Maranhão



AGRADECIMENTOS

Agradeço a Deus por permitir concluir mais esta etapa.

Aos meus pais André e Raissa e a minha irmã Larissa pelo apoio familiar.

Ao professor Wener Sampaio pelo auxílio, orientações e ensinamentos para realização deste trabalho.

Aos professores Bruno Feres e Wener Miranda pelo apoio operacional na pesquisa.

A todos os funcionários da SINFRA que participaram dos testes.

Aos professores Cláudio e Sérgio por aceitarem participar da banca.

A todos meus amigos que fiz nas diversas turmas do curso pela amizade.

Ao CNPq pelo apoio financeiro.

À Universidade Federal do Maranhão pela estrutura física.

RESUMO

A criminalidade custa caro ao Brasil, e com as restrições orçamentárias vigentes no país, é necessário um esforço para desenvolver técnicas de combate à criminalidade mais eficientes. Neste trabalho é feito um apanhado bibliográfico listando diversas técnicas de detecção e reconhecimento facial observadas em diversas literaturas, como também foi idealizado um sistema de reconhecimento facial de baixo custo, capaz de reconhecer rostos em tempo real e precisando apenas de uma imagem frontal do rosto da pessoa a ser reconhecida, utilizando recursos *open source*, além de ser adaptável à infraestrutura de câmeras, caso exista. Foi idealizada uma interface web para o sistema com finalidade de centralizar o processamento pesado em uma máquina servidora mais robusta. Os procedimentos experimentais foram feitos utilizando câmeras de segurança da Universidade Federal do Maranhão e utilizando vídeos disponíveis na internet. Foi percebido que o posicionamento correto das câmeras é de grande importância para o bom funcionamento do sistema, e que a tolerância aplicada no reconhecimento facial deve ser customizada de acordo com a aplicação e a qualidade das câmeras utilizadas.

Palavras-chave: Reconhecimento facial. Visão computacional. Segurança Pública

ABSTRACT

Crime is expensive in Brazil, and with budget restrictions in force in the country, an effort is needed to develop more efficient techniques to fight crime. In this work, a bibliographic overview is made listing several facial detection and recognition techniques observed in several literature, as well as a low cost facial recognition system, able to recognize faces in real time and requiring just one frontal face picture to recognize someone's face, using open source resources, in addition to being adaptable to the camera infrastructure, if any. It was a web interface designed for the system with comforters to centralize heavy processing on a more robust server machine. Experimental procedures were performed using security cameras from the Federal University of Maranhão and using videos available on the internet. It was realized that the correct positioning of the cameras is of great importance for the proper functioning of the system, and that the tolerance applied in facial recognition must be customized according to the application and the quality of the cameras used.

Keywords: Facial recognition. Computer vision. Public security

LISTA DE FIGURAS

Figura 1- Ilustração das regras do nível 1 da técnica de Yang & Huang (1994). Fonte: Yang et al (2002).....	13
Figura 2- Exemplo de projeção horizontal e vertical da imagem de um rosto. Fonte: Costa, 2018.	15
Figura 3- (a) região da face selecionada; (b) cluster de cores no espaço de cores; (c) cluster no espaço cromático. Fonte: Feris et al, 2000.	16
Figura 4– Funcionamento resumido da RNA. Fonte: Rowley et al, 1996.	19
Figura 5 - Os quatro tipos de características utilizadas na abordagem original do detector de objetos Viola-Jones (fonte: Araújo, 2010)	21
Figura 6 - Tecnologias utilizadas na aplicação.....	25
Figura 7 - Tela de login.	25
Figura 8- Homepage do sistema.	26
Figura 9- Cadastramento de rostos.	27
Figura 10- Funcionalidade de gerenciamento de rostos	27
Figura 11- Detalhamento de informações do rosto.	28
Figura 12- Menu de edição das informações dos rostos.....	29
Figura 13- Funcionalidade de listar câmeras disponíveis.....	30
Figura 14- Imagem da câmera com reconhecimento facial ao vivo.....	30
Figura 15- Reconhecimento facial por vídeo local.....	31
Figura 16- Ambiente do administrador do sistema.	32
Figura 17- Testes realizados com as câmeras de segurança da UFMA.....	33
Figura 18- Teste com vídeos disponíveis na internet.	33
Figura 19 - Posicionamento inadequado da câmera atrapalhando a obtenção de imagens dos rostos.....	34



Figura 20- Identificação realizada com tolerância de 0,5 mostrando a mesma pessoa sendo identificada como três pessoas distintas.	35
Figura 21 - Identificação realizada com tolerância de 0,7 mostrando duas pessoas diferentes sendo identificadas como a mesma pessoa.	36

SUMÁRIO

1. INTRODUÇÃO	10
2. OBJETIVOS	10
2.1. Objetivo Geral	10
2.2. Objetivos Específicos.....	11
3. JUSTIFICATIVA	11
4. FUNDAMENTAÇÃO TEÓRICA.....	12
4.1. Detecção de rostos.....	12
4.1.1. Métodos baseados em Conhecimento.....	12
4.1.2. Métodos baseados em Características Invariantes.....	15
4.1.3. Métodos Baseados em Templates.....	16
4.1.4. Métodos baseados na aparência.....	17
4.1.4.1 <i>Eigenfaces</i>	17
4.1.4.2. Redes Neurais.....	18
4.1.5. Abordagem em cascata.....	20
4.1.5.1. Haar Cascade.....	20
4.2. O processo de reconhecimento facial com <i>face_recognition</i>	21
5. METODOLOGIA.....	24
6. PROCEDIMENTOS EXPERIMENTAIS.....	32
7. RESULTADOS	34
8. CONCLUSÃO.....	36
REFERÊNCIAS BIBLIOGRÁFICAS	38

1. INTRODUÇÃO

A visão computacional é um ramo da computação que tem sido cada vez mais presente no dia-a-dia. Dentre as possibilidades de aplicações que a visão computacional proporciona, destaca-se o reconhecimento facial.

De acordo com Wagner (2012), o reconhecimento facial é uma tarefa fácil para humanos. Experimentos mostraram que até recém-nascidos de um a três dias de vida são capazes de distinguir entre rostos conhecidos. Assim, o quão difícil é essa tarefa para um computador? Acontece que sabemos pouco sobre o reconhecimento humano até hoje.

Desta maneira, existem inúmeras implementações para realização desta tarefa, a escolha de qual usar vai depender da finalidade da aplicação que se deseja construir, como também dos recursos de hardware disponíveis. Estas implementações serão estudadas neste trabalho divididas em cinco categorias: Métodos Baseados em Conhecimento, Métodos Baseados em Características Invariantes, Métodos Baseados em *Templates*, Métodos Baseados na Aparência e Abordagem em Cascata.

Cabe ressaltar que para atender as finalidades de utilização pelos órgãos de segurança pública, a aplicação que será proposta tem como requisitos: identificar uma pessoa a partir de uma única foto frontal do seu rosto e realizar esta identificação em tempo real. Assim, o método que mais se encaixa nos requisitos apurados é o de reconhecimento facial utilizando Gradientes de Histogramas Orientados.

2. OBJETIVOS

2.1. Objetivo Geral

- Verificar as possibilidades da utilização da visão computacional para auxiliar no combate à criminalidade.

2.2. Objetivos Específicos

- Levantar as soluções *open source* para reconhecimento facial existentes;
- Idealizar um protótipo funcional de sistema reconhecimento facial;
- Idealizar uma interface visual amigável para utilização do sistema.

3. JUSTIFICATIVA

O problema da segurança pública do Brasil é mundialmente conhecido. De acordo com a Secretaria Especial de Assuntos Estratégicos da Presidência da República (2018), os custos econômicos da criminalidade cresceram de forma substancial entre 1996 e 2015, de cerca de 113 bilhões de reais para 285 bilhões de reais (equivalente a 4,38% do PIB brasileiro de 2015).

Este crescimento dos custos para a repressão da criminalidade é preocupante visto que a capacidade de investimento público no Brasil fica mais reduzida a cada ano. Diante de um cenário de grande restrição orçamentária em que o país vive atualmente, é necessário que sejam desenvolvidas técnicas mais eficientes para reprimir o crime no Brasil.

Visando aumentar tal eficiência no combate ao crime, os sistemas de reconhecimento facial são uma solução possível para isto. Um primeiro motivo que pode ser exposto é a redução na quantidade de mão de obra para realizar atividades de patrulhamento e vigilância ostensiva, já que este tipo de sistema opera de maneira praticamente autônoma, necessitando de pessoas apenas para retornar os resultados em caso de alguma ocorrência.

Outro motivo que pode ser citado é que a infraestrutura de câmeras necessária para implementação da solução é relativamente de fácil acesso, e em muitos lugares essa infraestrutura já é existente necessitando a aquisição de poucos equipamentos para execução do sistema.

Também é importante ressaltar que o desenvolvimento deste tipo de sistema é de grande importância estratégica para o país, já que diversos países estão em busca do aprimoramento deste tipo de tecnologia para fins além da segurança pública, como defesa e soberania nacional.

4. FUNDAMENTAÇÃO TEÓRICA

4.1. Detecção de rostos

O reconhecimento facial é uma tarefa complexa para computadores. De acordo com Lopes (2012), detectar a face antes de detectar cada característica em particular poupa muito trabalho, uma vez que a maioria dos algoritmos se baseia na procura por tais elementos em toda a imagem. Desta maneira, se utilizarmos na imagem sem nenhum pré-processamento, o desempenho da aplicação será aquém do aceitável. Nesta seção serão abordadas diversas técnicas de detecção facial.

4.1.1. Métodos baseados em Conhecimento

Os métodos baseados em conhecimento são o conjunto de técnicas de detecção facial que utilizam regras pré-definidas pelo pesquisador a partir do entendimento deste do que é uma face. Por exemplo, podemos definir que uma face humana possui as seguintes características: dois olhos, um nariz e uma boca, pois sabemos que a maioria das faces humanas possuem tais características.

Segundo Lopes (2012), este método sofre de algumas desvantagens inerentes a construção do conjunto de regras. Se as regras são muito gerais, corre-se o risco de que o sistema que as utiliza apresentar uma alta taxa de falsos positivos, ou seja, elementos erroneamente

identificados como face. Entretanto, se o conjunto de regras for muito específico, a precisão da detecção tende a diminuir bastante.

Como exemplo, Yang & Huang (1994) construíram uma técnica de detecção de faces baseada no conhecimento, através de um conjunto de três níveis de regras hierárquicas. No nível mais alto, todas as faces candidatas possíveis são encontradas por uma janela deslizante que é passada pela imagem de entrada aplicando o conjunto de regras pré-definidas. As regras deste nível são a forma mais genérica de descrever um rosto, por exemplo: a parte central do rosto possui quatro células com intensidade uniforme (Figura 1, parte destacada escura), a parte superior do rosto possui basicamente a mesma intensidade (Figura 1, parte destacada clara), e a diferença entre a média dos valores cinzas da parte central e da parte superior é significativa.

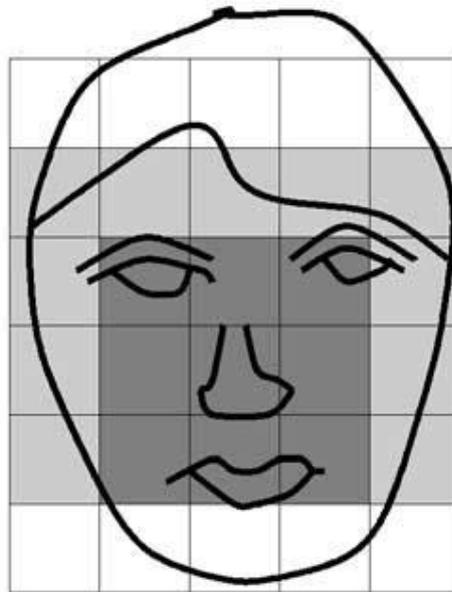


Figura 1- Ilustração das regras do nível 1 da técnica de Yang & Huang (1994). Fonte: Yang et al (2002).

As imagens que passaram são enviadas ao próximo nível. Neste nível é realizado um histograma de equalização seguido por uma detecção de borda. As imagens que restaram são submetidas ao terceiro e último nível, em que são detectados os componentes da face como olhos, boca e nariz.

Outra abordagem deste tipo foi idealizada por Kotropoulos & Pitas (1997). Eles idealizaram um sistema em que as características faciais são definidas como:

$$HI(x) = \sum_{y=1}^n I(x, y) \quad \text{Eq. 1}$$

$$VI(y) = \sum_{x=1}^m I(x, y) \quad \text{Eq. 2}$$

Em que:

n – Altura da imagem;

m – Largura da imagem;

x – Coordenada horizontal da imagem;

y – Coordenada vertical da imagem;

I(x,y) – Intensidade do pixel na posição x,y;

HI(x) – Projeção horizontal da imagem;

VI(y) – Projeção vertical da imagem.

Segundo Costa (2018), analisando-se os mínimos e máximos locais da projeção é possível detectar a face e as regiões dos olhos, boca e nariz (Figura 2). O problema dessa abordagem é que os critérios ficam comprometidos em imagens obtidas com fundo complexo.

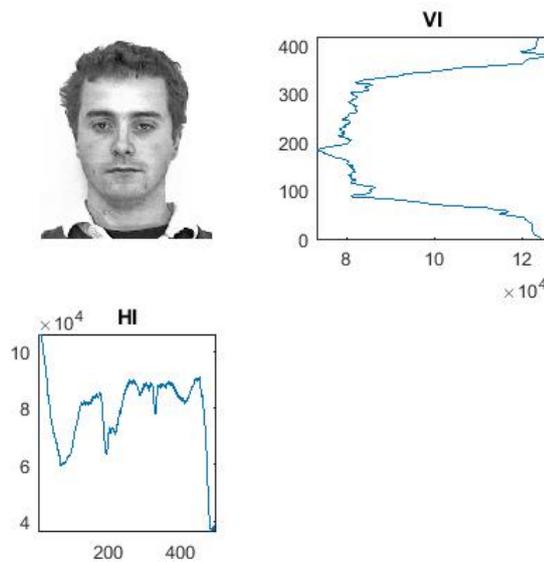


Figura 2- Exemplo de projeção horizontal e vertical da imagem de um rosto. Fonte: Costa, 2018.

4.1.2. Métodos baseados em Características Invariantes

De acordo com Lopes (2012), estes métodos são inspirados na capacidade que os seres humanos possuem de identificar objetos independentes do ponto de vista. A principal desvantagem de tal abordagem é que tais características podem ser corrompidas devido as condições de iluminação ou algum tipo de ruído, comprometendo assim, a eficiência de tal abordagem.

Segundo Yang et al (2002), características faciais como olhos, nariz, boca e cabelos são facilmente extraídas usando detectores de bordas. Um modelo estatístico é construído para descrever seus relacionamentos e verificar a existência de um rosto baseado em extração de características.

Lopes (2012) ressalta que a cor da pele e a textura da face são as principais características invariantes que podem ser utilizadas para separar a face de outros objetos

presentes em uma cena. Costa (2018) complementa dizendo que se constata que a cor da pele independente de suas variações (branca, negra, amarela, etc.), tende a formar um *cluster* no espaço de cores, podendo ser modelada por uma distribuição gaussiana (Figura 3). Assim, esta característica pode ser usada para diferenciar a imagem de um rosto do fundo da imagem.

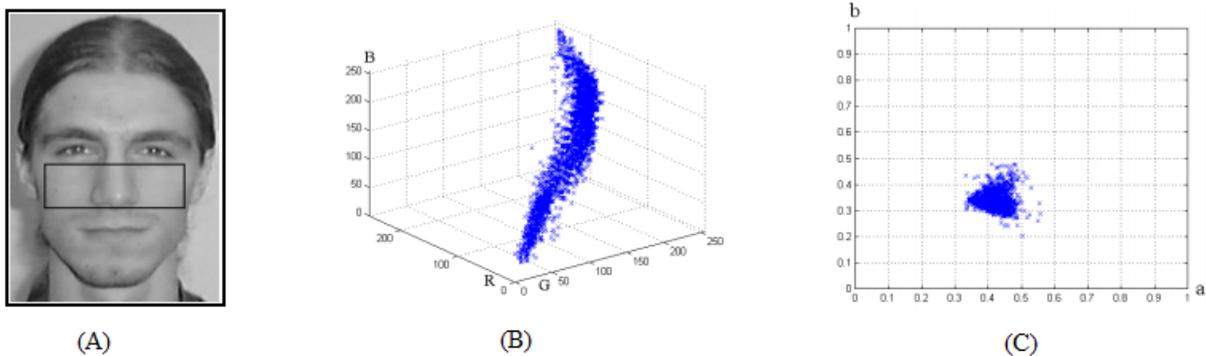


Figura 3- (a) região da face selecionada; (b) cluster de cores no espaço de cores; (c) cluster no espaço cromático. Fonte: Feris et al, 2000.

4.1.3. Métodos Baseados em Templates

Uma maneira simples de buscar um objeto em uma imagem é procurar pela sua forma previamente conhecida. Segundo Lopes (2012), uma das formas mais comuns de modelar a forma de um objeto é descrevê-lo através de seus componentes geométricos básicos, como círculos, quadrados ou triângulos, esta técnica é denominada *template*. A detecção do objeto, portanto, consistirá em achar a melhor correspondência, definida através de uma função de energia, entre o objeto presente na imagem e o seu molde (*template*).

No caso da detecção de faces, segundo Yang et al (2002), dado uma imagem de entrada, os valores de correlação com os formatos padrões são computados para o contorno do rosto, olhos, nariz, e boca independentemente. A existência de uma face é determinada baseada nos valores de correlação. Já Lopes (2012) relata que a *template* mais utilizada é aquela que trata a face como uma elipse.

Este tipo de abordagem possui prós e contras. Segundo Yang et al (2002), a vantagem dessa abordagem é a facilidade de implementação, porém ela não consegue lidar com a variação de escala, pose e formato.

4.1.4. Métodos baseados na aparência

Segundo Costa (2018), os métodos baseados em aparência, não utilizam qualquer conhecimento prévio ou características empíricas do objeto a ser detectado. Nessa classe de algoritmos surgem os conceitos de aprendizado e treinamento, uma vez que as informações necessárias para detectar o objeto desejado são retiradas do conjunto de imagens. Yang et al (2002) complementa dizendo que em geral, os métodos baseados em aparência dependem de técnicas da análise estatística e *machine learning* para encontrar as características relevantes nas imagens com ou sem rostos.

4.1.4.1 *Eigenfaces*

O método *eigenfaces* foi proposto em 1991 por Turk & Pentland. Segundo Turk & Pentland (1991), as imagens de rostos não estão distribuídas aleatoriamente em um espaço de alta dimensionalidade. Assim, a ideia principal desta técnica é utilizar a transformada de *Karhunen-Loeve* ou PCA (*Principal Component Analysis*) para encontrar os vetores que melhor representam a distribuição das imagens de rostos no espaço da imagem completa.

Segundo Lopes (2012), a determinação da nova base de vetores é feita da seguinte maneira: Considerando $I(x,y)$ uma imagem de duas dimensões $N \times N$, ou um vetor de dimensão N^2 . A partir de um conjunto de imagens de testes $I = \{I_1, I_2, \dots, I_M\}$, encontra-se a face média f_m do conjunto definida por:

$$fm_j = \frac{1}{M} \sum_{i=1}^M I_i^j \quad \text{Eq. 3}$$

em que I_i^j refere-se ao pixel j da imagem i.

A diferença de cada face para a média e dada por:

$$\varphi_i = I_i - fm \quad \text{Eq. 4}$$

Assim, submete-se o conjunto I ao PCA que retorna um conjunto M de vetores ortogonais U_k e seus respectivos autovalores (*eigenvalues*) λ_k . Dessa maneira, os vetores U_k e os escalares λ_k são os autovetores e autovalores da matriz de covariância definida por:

$$C = \frac{1}{M} \sum_{n=1}^M \varphi_n \varphi_n^T = AA^T \quad \text{Eq. 5}$$

A dimensão de A é N^2 , tornando complexo o cálculo dos N^2 autovalores e autovetores para imagens de dimensões grandes. Neste momento, faz-se necessário utilizar a técnica PCA para encontrar os autovetores através de uma matriz de tamanho M e realizar combinações lineares dos vetores resultantes, dessa maneira reduzindo os cálculos da ordem de N^2 para M.

Desta forma, segundo Costa (2018), aproveitando-se do fato de que as faces não mudam radicalmente quando projetadas no espaço de *eigenfaces*, mas quando um objeto arbitrário é projetado neste mesmo espaço ele muda sua aparência drasticamente, pode-se utilizar esta técnica para detectar faces apenas calculando a distância do objeto projetado em relação as classes de face projetadas e através de um *threshold* verificar se este é uma face ou não.

4.1.4.2. Redes Neurais

De acordo com Lopes (2012), as Redes Neurais Artificiais (RNA's) são modelos matemáticos biologicamente inspirados, com o intuito de emularem os processos cognitivos do cérebro humano. Já Costa (2018) acrescenta que a RNA é usada nas tarefas de classificação e

reconhecimento de padrões. No caso de detecção de face, usa-se RNA para diferenciar imagens ou regiões da imagem que possivelmente contenham uma face daquelas que não contém.

Segundo Yang et al (2002), a vantagem de usar redes neurais para detecção de rosto é a viabilidade de treinar um sistema para capturar a densidade condicional de classe complexa dos padrões de rosto. No entanto, uma desvantagem é que a arquitetura de rede deve ser amplamente ajustada (número de camadas, número de nós, taxas de aprendizagem, etc.) para obter um desempenho excepcional.

Rowley et al (1996) construíram um sistema de reconhecimento facial usando RNA's (Figura 4). Este sistema consiste em um filtro com uma janela de 20x20 que percorre uma imagem de entrada extraíndo regiões onde podem haver faces. Após isto, tais regiões são submetidas a uma equalização de histograma e normalização da luz. Caso a imagem do rosto seja maior que a janela, a imagem é sub-amostrada iterativamente por um fator 1,2 e o filtro é reaplicado. Então, cada parte extraída da imagem é enviada para a rede neural que decide se há ou não uma face nesta parte. Segundo testes realizados pelo autor, uma rede possuindo 78 neurônios intermediários e 4357 com 24 conexões apresentou uma taxa de acerto na ordem de 92,7%.

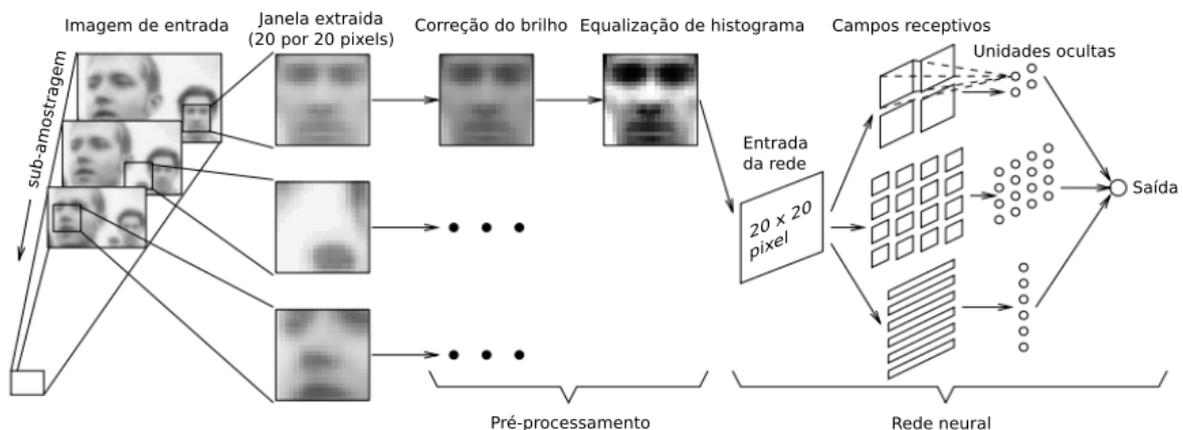


Figura 4– Funcionamento resumido da RNA. Fonte: Rowley et al, 1996.

4.1.5. Abordagem em cascata

Segundo Costa (2018), métodos baseados em conhecimento, características invariantes ou *templates* exigem um conhecimento do objeto a ser detectado para ajustar os parâmetros do detector. Já a abordagem baseada em aparência transfere essa exigência para o próprio algoritmo de detecção (RNA) onde técnicas de aprendizado de máquina são aplicadas para ajustar os parâmetros. A abordagem em cascata permite que os métodos apresentados sejam aplicados de maneira mais inteligente, contribuindo para uma alta taxa de detecção.

4.1.5.1. Haar Cascade

O *Haar Cascade* é um classificador utilizado para detecção de objetos. Segundo Cuimei et al (2017), o classificador *cascade* típico é um método muito bem-sucedido desenvolvido por Viola & Jones (2001) para reconhecimento facial. De acordo com Araújo (2010), o Viola-Jones é capaz de detectar faces com precisão, alta taxa de acerto, baixa taxa de falsos positivos e baixo custo computacional.

Cuimei et al (2017) diz que o classificador *cascade* é uma tecnologia baseada em árvore, em que Viola-Jones utiliza características *Haar-like* para reconhecimento de rostos humanos. De acordo com Araújo (2010), o algoritmo é composto de três partes. A primeira delas é a representação da imagem em um espaço de características baseadas nos filtros de Haar. Isto é feito com o auxílio da “imagem integral” (Figura 5).

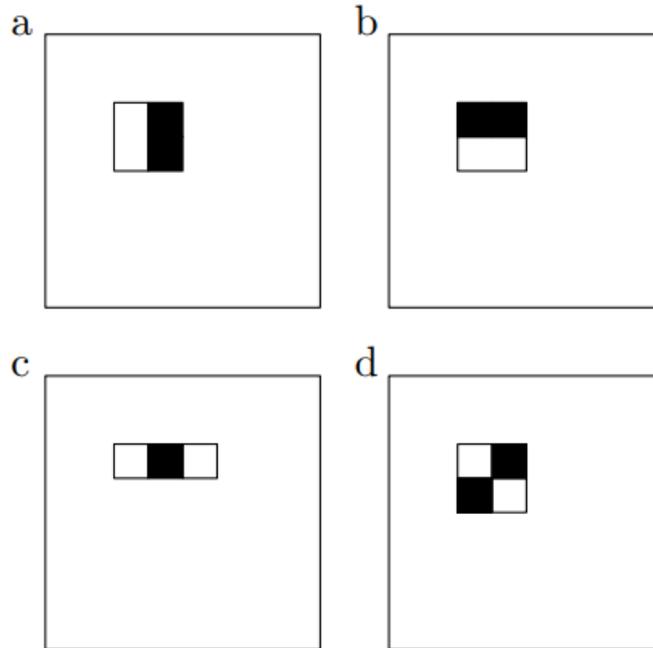


Figura 5 - Os quatro tipos de características utilizadas na abordagem original do detector de objetos Viola-Jones (fonte: Araújo, 2010)

A segunda, de acordo com Araújo (2010), é a montagem de um classificador baseado em *Boosting* capaz de selecionar as características mais relevantes. Por fim é feita uma combinação em cascata destes classificadores de modo garantir bom desempenho e velocidade de processamento.

4.2. O processo de reconhecimento facial com *face_recognition*¹

Segundo Kumar et al (2020), reconhecimento facial é o processo de identificação ou verificação de uma pessoa a partir de alguma imagem, a qual é obtida tanto de um dispositivo de captura de imagem ou de um *frame* individual de um vídeo.

¹ Disponível em <https://face-recognition.readthedocs.io/en/latest/readme.html>.

De acordo com a documentação da biblioteca *face_recogniton*, ela utiliza o estado da arte de reconhecimento facial da biblioteca *Dlib*². Este estado da arte foi testado no banco de imagens *Labeled Faces in the Wild*³ obtendo uma taxa de acerto de 99,38%.

O primeiro passo para o reconhecimento facial, de acordo com Kumar et al (2020), é encontrar os rostos na imagem. Para isto, o estado da arte da biblioteca *Dlib* utiliza uma técnica chamada *Histogram of Oriented Gradients* (HOG).

De acordo com Lopes (2015), O cálculo do HOG é dividido em uma série de etapas: normalização de cor da imagem, cálculo dos gradientes, cálculo dos histogramas e normalização de contraste local.

Na normalização da cor da imagem, segundo Kumar et al (2020), a imagem é convertida para escala de cinzas, pois as cores não são necessárias para este processo.

Feita a normalização, são calculados os gradientes horizontais e verticais. Segundo Lopes (2015), este cálculo é feito da seguinte maneira:

Para uma imagem em escala de cinza I , o gradiente é uma matriz G , com as mesmas dimensões de I , onde cada ponto $G(x,y)$ representa a variação de luminosidade que ocorre em $I(x,y)$. Como a variação é bidimensional, cada ponto do gradiente de imagem será um vetor de duas dimensões, podendo ser representado tanto na forma cartesiana quanto polar. Com base nisso, podem ser definidos:

Gradiente horizontal de uma imagem I é o componente horizontal de G , sendo denotado por G_x e calculado seguindo a equação:

$$G_x(x, y) = I(x + 1, y) - I(x - 1, y) \quad \text{Eq. 6}$$

Gradiente vertical de uma imagem I é o componente vertical de G , sendo denotado por G_y e calculado seguindo a equação:

² Disponível em <http://dlib.net/>.

³ Disponível em <http://vis-www.cs.umass.edu/lfw/>.

$$G_y(x, y) = I(x, y + 1) - I(x, y - 1) \quad \text{Eq. 7}$$

Magnitude de gradiente de uma imagem I é a magnitude de G , sendo denotada por G_ρ e calculada seguindo a equação:

$$G_\rho(x, y) = \sqrt{(G_x(x, y))^2 + (G_y(x, y))^2} \quad \text{Eq. 8}$$

Orientação de gradiente de uma imagem I é a orientação (ou fase) de G , sendo denotada por G_θ e calculada seguindo a equação:

$$G_\theta = \arctan\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \quad \text{Eq. 9}$$

Após isso, é gerado o histograma de gradientes orientados. O objetivo deste histograma, segundo Kumar et al (2020), é saber o quão escuro é um pixel em relação aos pixels que o rodeiam. Em seguida, desejamos desenhar uma seta mostrando a direção em que a imagem está ficando mais escura. Ao fazer isso, o resultado é que convertemos a imagem em uma representação simples de setas chamadas gradientes. Então, geramos um padrão geral coletando gradientes de vários rostos e assim comparar com a imagem que queremos detectar.

O próximo passo, segundo Kumar et al (2020), é codificar as faces. Para realizar isso, é utilizada uma rede chamada *FaceNet* desenvolvida por Schoff et al (2015). Esta rede é uma rede convolucional profunda que codifica um rosto em um *array* com 128 parâmetros.

Assim, segundo Kumar et al (2020), isto permite que o computador descobrirá como fazer medições de maneira confiável para qualquer rosto, uma vez treinado. E a última etapa é usar um classificador como o classificador SVM⁴ para comparar as faces e classificá-las com base em um *threshold* predeterminado.

⁴ Proposto por Boser et al. (1992).

5. METODOLOGIA

O sistema proposto consiste em uma aplicação *web* que obtém imagens geradas por câmeras de segurança via *IP* em tempo real ou por meio de vídeos salvos como arquivos locais, realizando o reconhecimento facial das pessoas que transitam por tais imagens a partir de uma única imagem frontal do rosto da pessoa a ser reconhecida.

Para realização do reconhecimento facial, utilizou-se a biblioteca *face_recognition* para linguagem de programação *Python*⁵. Esta biblioteca utiliza-se do estado da arte de reconhecimento facial da biblioteca *Dlib*, sendo a primeira uma API para a segunda, a qual utiliza-se a linguagem *C++*⁶ e a biblioteca *CUDA*⁷ para ganho de desempenho. A obtenção das imagens foi realizada pela biblioteca *OpenCV*⁸.

A ideia de utilização de uma interface *web* para esta aplicação veio do fato que o reconhecimento facial é uma tarefa que demanda muito processamento, e por isso, seria necessário a utilização de uma máquina mais potente, equipada com uma placa gráfica (*GPU*) com suporte a biblioteca *CUDA*. Esta arquitetura permite que seja centralizado o processamento em uma única máquina, mais potente, funcionando como um servidor, que pode ser acessado por máquinas menos robustas, incluindo dispositivos móveis como *smartphones e tablets*.

Para construção desta interface *web*, foi utilizado o *framework django*⁹. A escolha deste *framework* deu-se pelo motivo que ele utiliza como linguagem de programação *Python*, a mesma que a biblioteca *face_recognition*, assim resumindo o trabalho de integração entre estas partes a uma simples importação. Como banco de dados foi utilizado o *Postgresql*¹⁰. Esta escolha foi feita pela facilidade de integração que há deste banco de dados com o *framework*

⁵ Disponível em <https://www.python.org/>.

⁶ Disponível em <https://isocpp.org/>.

⁷ Disponível em <https://developer.nvidia.com/cuda-downloads>.

⁸ Disponível em <https://opencv.org/>.

⁹ Disponível em <https://www.djangoproject.com/>.

¹⁰ Disponível em <https://www.postgresql.org/>.

django e também pela facilidade que este banco lida com estruturas de dados como *arrays*. Na figura 6 temos um resumo das tecnologias utilizadas.

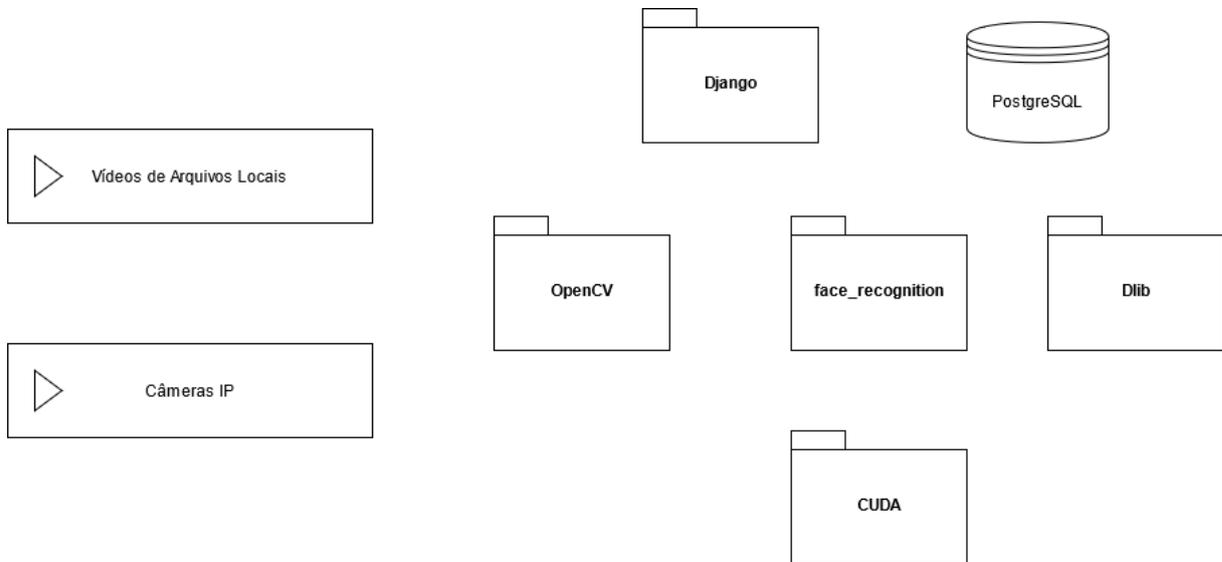


Figura 6 - Tecnologias utilizadas na aplicação.

Considerando que o sistema tem acesso a imagens de segurança, houve a necessidade de restringir o acesso, permitindo este a apenas pessoas devidamente registradas por um usuário administrador do sistema, assim foi criado uma tela inicial de *login* (Figura 7).

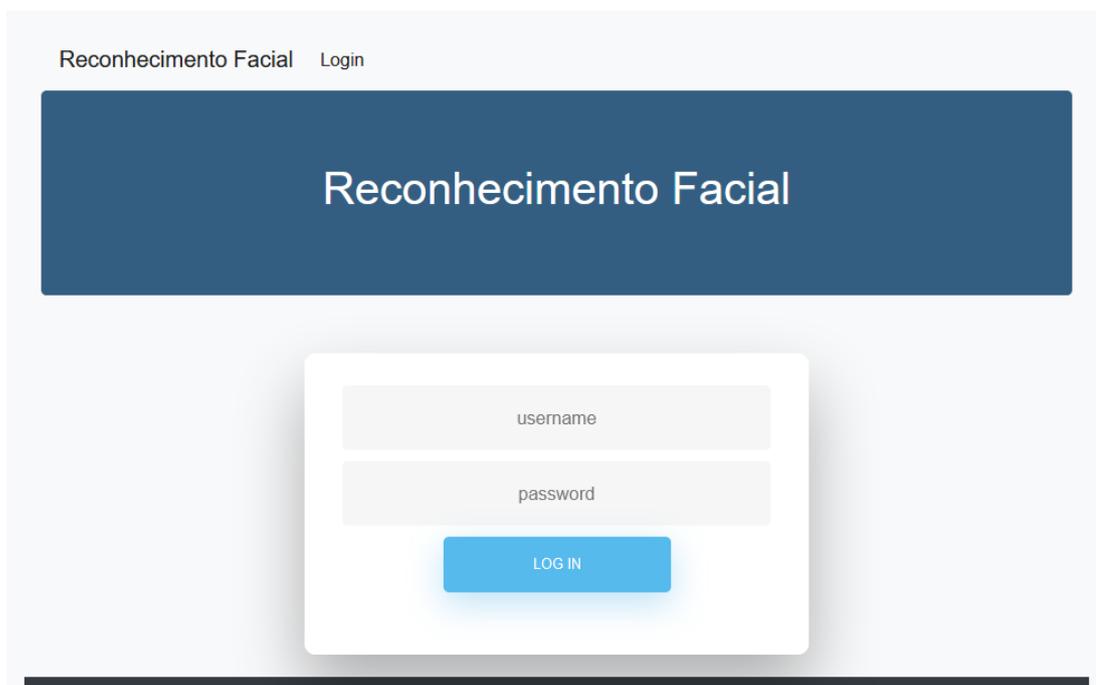


Figura 7 - Tela de login.

Efetuada o *login*, o usuário acessará a *homepage* do sistema (Figura 8), na qual poderá selecionar a funcionalidade do sistema deseja utilizar.



Figura 8- Homepage do sistema.

Clicando na primeira opção exibida, o usuário será redirecionado para a página de cadastramento de rostos (Figura 9). Esta funcionalidade tem como objetivo cadastrar os rostos das pessoas que serão identificadas pelo sistema. Na página há um formulário com três campos de preenchimento obrigatório, no primeiro deles deverá ser colocado o nome da pessoa, no segundo, a foto da pessoa, e no terceiro deverá ser marcado o status da pessoa, este status tem como opções: nenhum, foragido e desaparecido. Para finalizar o cadastro, basta clicar em avançar.

Reconhecimento Facial Usuário: Lucas Sair

Reconhecimento Facial

Insira o nome

Selecionar Foto: No file selected.

Status

Figura 9- Cadastramento de rostos.

Clicando na segunda opção da *homepage*, temos a funcionalidade de gerenciar os rostos cadastrados. Nesta funcionalidade será carregada uma página contendo todos os rostos cadastrados no sistema (Figura 10).

Reconhecimento Facial Usuário: Lucas Sair

Reconhecimento Facial

 <p>Afran Status: N/A</p> <p><input type="button" value="EDITAR INFORMAÇÕES"/></p> 	 <p>Amanda Status: N/A</p> <p><input type="button" value="EDITAR INFORMAÇÕES"/></p> 	 <p>Argemiro Status: N/A</p> <p><input type="button" value="EDITAR INFORMAÇÕES"/></p> 	 <p>Aureliana Status: N/A</p> <p><input type="button" value="EDITAR INFORMAÇÕES"/></p> 
---	--	--	---

Figura 10- Funcionalidade de gerenciamento de rostos

Abaixo de cada rosto será mostrado o nome e o status da pessoa, além de uma opção de editar as informações (Figura 11).

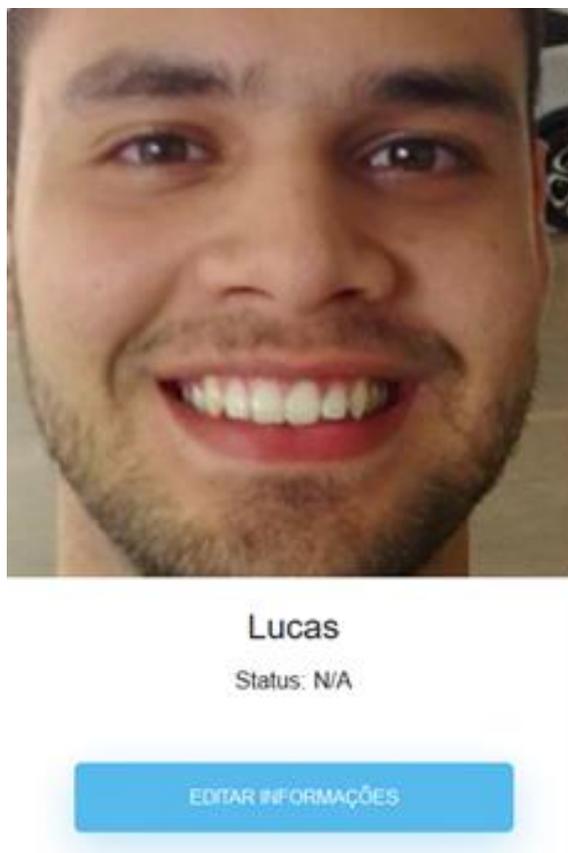
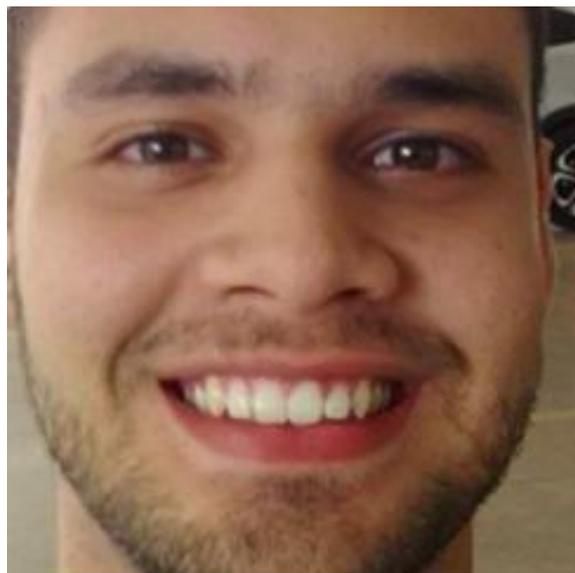


Figura 11- Detalhamento de informações do rosto.

Clicando nessa opção, será mostrado um menu (Figura 12) no qual podem ser editadas as informações desta pessoa, ou até mesmo a exclusão desta.



Nome:

Status:

Figura 12- Menu de edição das informações dos rostos.

Na terceira opção constante na *homepage*, temos a funcionalidade de listar as câmeras disponíveis ao sistema (Figura 13). Para as câmeras que estiverem online, será mostrado o botão de acesso às imagens, clicando neste botão, a aplicação irá exibir a imagem gerada pela câmera ao vivo no navegador (Figura 14).

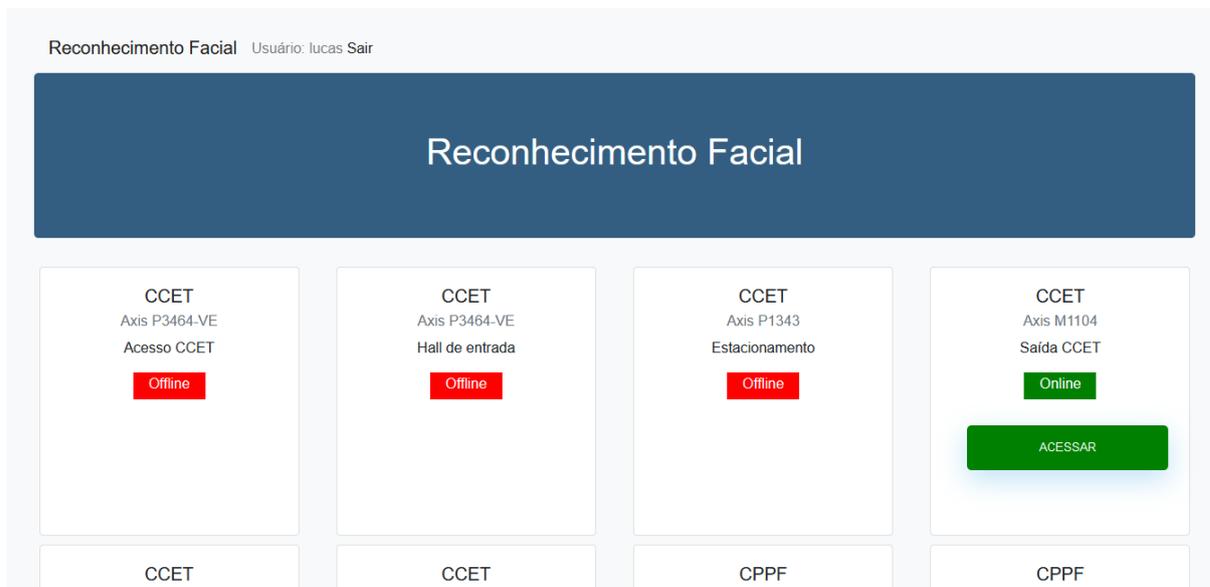


Figura 13- Funcionalidade de listar câmeras disponíveis.

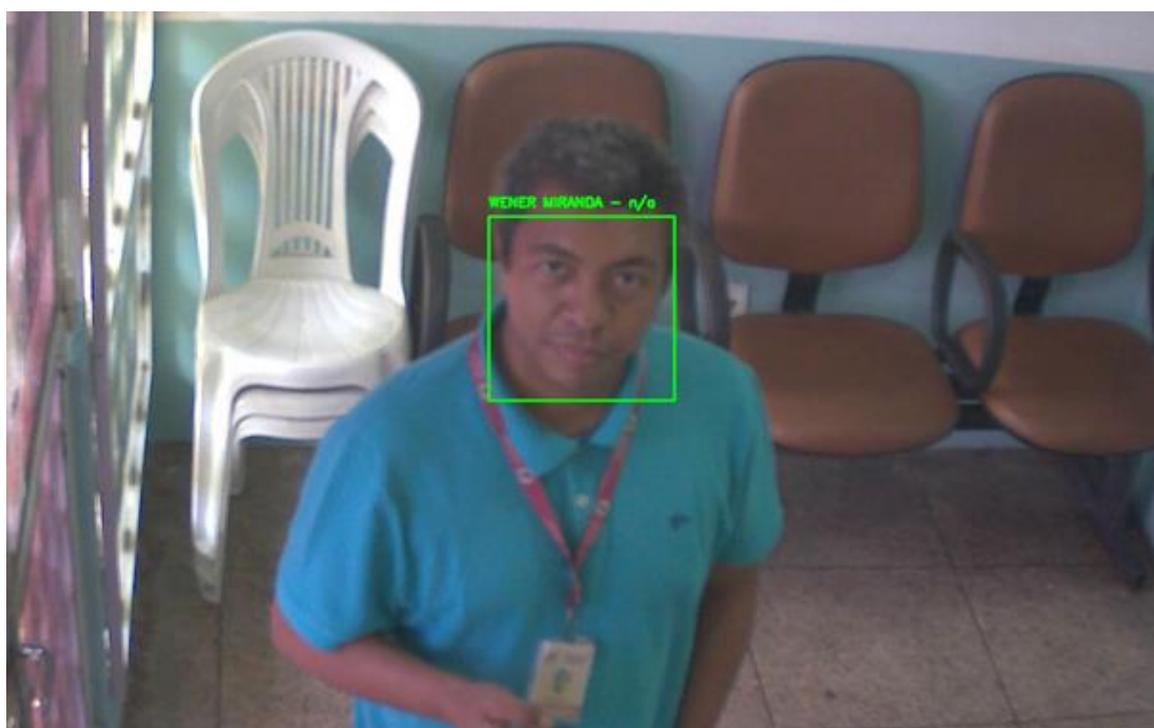


Figura 14- Imagem da câmera com reconhecimento facial ao vivo.

A última funcionalidade contida na *homepage* é a funcionalidade de reconhecimento facial por vídeo local (Figura 15). Nesta funcionalidade, o usuário irá fazer upload de um arquivo de vídeo em formato MP4, AVI ou MKV contida no armazenamento de seu dispositivo,

e o sistema realizará o reconhecimento facial deste vídeo, retornando um novo arquivo de vídeo no formato AVI para download.

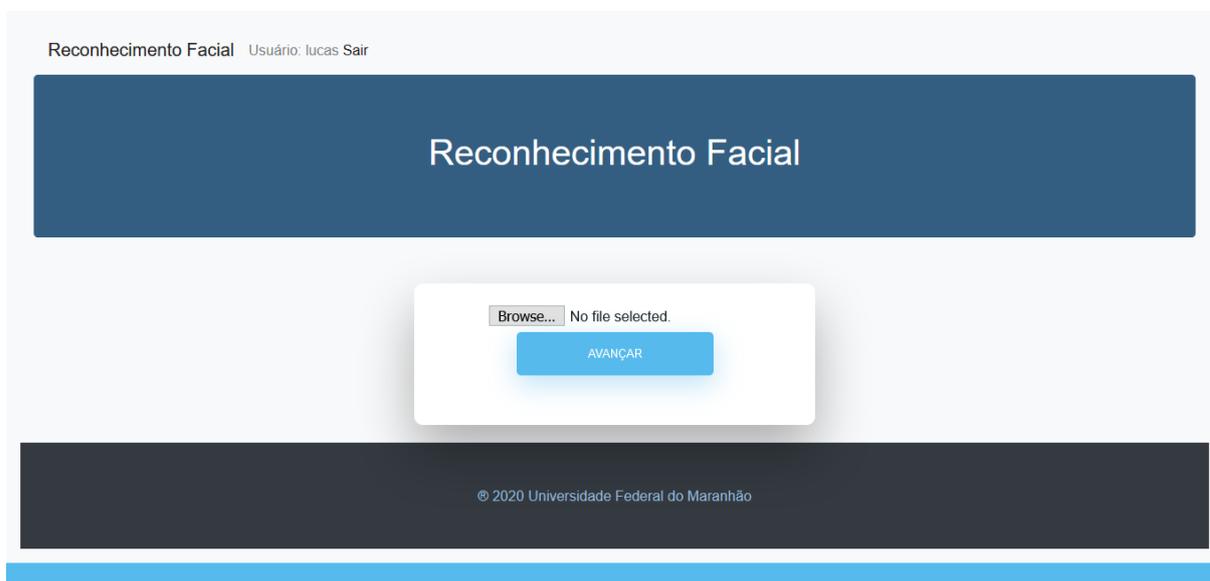


Figura 15- Reconhecimento facial por vídeo local.

Além das funcionalidades listadas na *homepage*, há também um ambiente de gerenciamento do sistema provido pelo *framework django* (Figura 16), o qual pode ser acessado exclusivamente por um usuário administrador. Nesse ambiente poderá ser realizado a criação ou exclusão de usuários, câmeras e rostos.

Administração do Site

AUTENTICAÇÃO E AUTORIZAÇÃO		
Grupos	+ Adicionar	✎ Modificar
Usuários	+ Adicionar	✎ Modificar

FACERECOGNITION		
Cameras	+ Adicionar	✎ Modificar
Procurados	+ Adicionar	✎ Modificar

Ações recentes

Minhas Ações

- ×** desconhecido_22
Procurado
- ×** Guarda da Ufma
Procurado
- ×** Guarda da Ufma
Procurado
- ×** desconhecido_25
Procurado
- ×** desconhecido_26
Procurado
- ×** desconhecido_27
Procurado
- ×** desconhecido_28
Procurado
- ×** desconhecido_29
Procurado
- ×** desconhecido_30
Procurado
- ×** desconhecido_31
Procurado

Figura 16- Ambiente do administrador do sistema.

6. PROCEDIMENTOS EXPERIMENTAIS

Os procedimentos experimentais (Figura 17) foram realizados utilizando câmeras de segurança da Cidade Universitária do Bacanga da Universidade Federal do Maranhão. O acesso a estas câmeras foi concedido pela Superintendência de Infraestrutura (SINFRA/UFMA) por meio de um VPN.



Figura 17- Testes realizados com as câmeras de segurança da UFMA.

Outros experimentos foram realizados utilizando vídeos disponíveis na internet para testar a detecção com muitos rostos na mesma cena como também a funcionalidade de reconhecimento através de arquivos de vídeo (Figura 18).

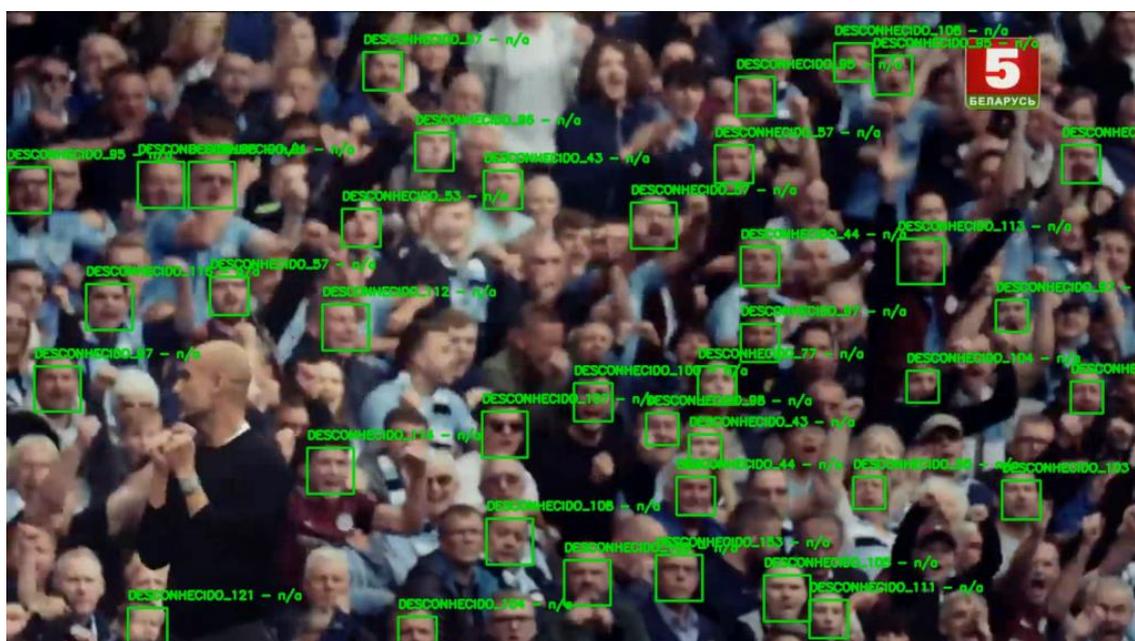


Figura 18- Teste com vídeos disponíveis na internet.

7. RESULTADOS

Com a realização dos testes, foi observado que por mais adaptável que seja o sistema à infraestrutura já existente, é de suma importância que as câmeras tenham seu posicionamento ajustado de maneira correta como pode ser visto na Figura 19, em que a câmera de segurança localizada na recepção da Superintendência de Infraestrutura (SINFRA/UFMA) foi posicionada atrás da porta, dificultando a obtenção de imagens dos rostos das pessoas que transitam naquele recinto.



Figura 19 - Posicionamento inadequado da câmera atrapalhando a obtenção de imagens dos rostos.

Outro ponto que merece destaque é o valor da tolerância para a comparação dos rostos. Esse valor varia de 0 a 1, quando ele se encontra mais próximo de 0, a tendência é que a

aplicação aponte que duas fotos diferentes da mesma pessoa sejam apontadas como duas pessoas diferentes conforme a Figura 20.



Figura 20- Identificação realizada com tolerância de 0,5 mostrando a mesma pessoa sendo identificada como três pessoas distintas.

Quando o valor da tolerância se encontra mais próximo a 1, a tendência é que a aplicação identifique pessoas distintas como a mesma pessoa conforme a Figura 21.



Figura 21 - Identificação realizada com tolerância de 0,7 mostrando duas pessoas diferentes sendo identificadas como a mesma pessoa.

Desta maneira, o valor da tolerância deve ser customizado de acordo com a finalidade da aplicação e a qualidade da infraestrutura, sendo que o valor padrão utilizado pela biblioteca *face_recognition* é de 0,6. Quanto mais imagens diferentes da mesma pessoa, fica mais aplicável a utilização de uma tolerância mais baixa.

8. CONCLUSÃO

Neste trabalho foi realizado um apanhado de técnicas de reconhecimento e detecção facial observadas em diversas literaturas, além de ser proposto um sistema de reconhecimento facial com custo reduzido capaz de ser utilizado pelos órgãos de segurança pública. Em um país com grandes restrições orçamentárias e com grandes problemas de segurança como o Brasil, este sistema é muito bem-vindo para ajudar no combate à criminalidade, já que apenas as licenças de utilização de sistemas existentes no mercado que realizam as mesmas funções do que foi proposto alcançam cifras milionárias, e em muitos casos, o estado não tem condição de arcar com tal ônus.



Coordenação do Curso Engenharia da Computação.
(Cidade Universitária Dom Delgado - São Luís)

Em trabalhos futuros, pretende-se criar uma versão comercial deste sistema com novas funcionalidades, com um desempenho mais aprimorado, além de uma interface ainda mais amigável ao usuário.

REFERÊNCIAS BIBLIOGRÁFICAS

ARAÚJO, Gabriel Matos. **ALGORITMO PARA RECONHECIMENTO DE CARACTERÍSTICAS FACIAIS BASEADO EM FILTROS DE CORRELAÇÃO.**

Dissertação de Mestrado. Universidade Federal do Rio de Janeiro. Rio de Janeiro, fevereiro de 2010.

BOSE, Bernhard E.; GUYON, Isabelle M.; VAPNIK, Vladimir N. **A training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on Computational learning theory.** 1992. p. 144-152.

COSTA, Talison Fernandes. **RECONHECIMENTO FACIAL COM HAAR CASCADE E KNN.** Universidade Federal Rural do Semi-Árido. Pau dos Ferros - RN, 2018.

CUDA. <<https://developer.nvidia.com/cuda-downloads>>, acessado em 16/12/2020 às 23:19.

CUIMEI, Li; ZHILIANG, Qi; NAN, Jia; JIANHUA, Wu. **Human face detection algorithm via Haar cascade classifier combined with three additional classifiers.** 13th International Conference on Electronic Measurement & Instruments. 2017.

DJANGO. <<https://www.djangoproject.com/>>, acessado em 16/12/2020 às 23:21.

DLIB LIBRARY. <<http://dlib.net/>>, acessado em 15/12/2020 às 09:41.

FACE RECOGNITION DOCS. <<https://face-recognition.readthedocs.io/en/latest/readme.html>

>, acessado em 15/12/2020 às 09:41.

FERIS, Rogério Schmidt; CAMPOS, Teófilo Emídio; JUNIOR, Roberto Marcondes Cesar.

Detection and Tracking of Facial Features in Video Sequences. *Lectures Notes in Artificial Intelligence*, vol. 1973, pp 197-206, abril de 2000, Springer-Verlag.T. D.

KOTROPOULOS, Constantine; PITAS, Ioannis. **Rule-Based Face Detection In Frontal Views**, *Em Proc. Int'l Acoustics, Speech and Signal Processing*, vol.4, pp. 2537-2540, 1997.

KUMAR, Praveen; VARMA, Kesava Jayendra; SUBRAMANYA, V; HARISH, Venkata Sai.

A MULTIPLE FACE RECOGNITION SYSTEM WITH DLIB'S RESNET NETWORK USING DEEP METRIC LEARNING. *Journal of Critical Reviews*, vol 7, issue 6, 2020.

LABELED FACES IN THE WILD. <<http://vis-www.cs.umass.edu/lfw/>>, acessado em 15/12/2020 às 09:41.

LOPES, E. Costa. Detecção de Faces e Características Faciais. Pós-graduação em Ciência da Computação - PUCRS, 2012.

LOPES, Marcelo Teider. **HISTOGRAMA DE GRADIENTES ORIENTADOS UTILIZANDO PROCESSAMENTO PARALELO EM GPU.** Trabalho de Conclusão de Curso, Universidade Tecnológica Federal do Paraná, Curitiba, 2015.

OPENCV. <<https://opencv.org/>>, acessado em 16/12/2020 às 23:20.

POSTGRESQL. <<https://www.postgresql.org/>>, acessado em 16/12/2020 às 23:23.

PYTHON. <<https://www.python.org/>>, acessado em 16/12/2020 às 23:15.

ROWLEY, Henry; BALUJA, Shumeet; KANADE, Takeo. **Neural Network-Based Face Detection**, *Computer Vision and Pattern Recognition*, 1996.

SCHOFF, Florian; KALENICHENKO, Dmitry; PHILBIN, James. **FaceNet: A unified embedding for face recognition and clustering**. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 815-823.

SECRETARIA ESPECIAL DE ASSUNTOS ESTRATÉGICOS DA PRESIDÊNCIA DA REPÚBLICA. **CUSTOS ECONÔMICOS DA CRIMINALIDADE NO BRASIL**. Brasília, julho de 2018.

STANDARD C++. <<https://isocpp.org/>>, acessado em 16/12/2020 às 23:17.

TURK, Matthew; PENTLAND, Alex. **Face recognition using eigenfaces**. Computer Vision and Pattern Recognition. IEEE Computer Society Conference, 1991.

VIOLA, Paul; JONES, Michael. **Rapid Object Detection Using a Boosted Cascade of Simple Features**. Conference on Computer Vision and Pattern Recognition, 2001.

Wagner, Philipp. **Face Recognition with Python**. Julho de 2012.

YANG, Guangzheng; HUANG, Thomas; **Human Face Detection in Complex Background.**

Pattern Recognition, vol. 27, N° 1, 1994.

YANG, Ming-Hsuan; KRIEGMAN, David; AHUJA, Narendra. **Detecting Faces in Images: A Survey.** IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol.24, No.1, janeiro de 2002.