

Reconhecimento facial utilizando OpenCV na segurança Pública

Anastácio Kemerich¹, Mirkos Ortiz Martins²

¹ Acadêmico do Curso de Ciência da Computação – Universidade Franciscana – Santa Maria – RS – Brasil.

² Professor do Curso de Ciência da Computação - Universidade Franciscana – Santa Maria – RS – Brazil.

{anastacio.kemrich,mirkos}@ufn.edu.br

Abstract. *This project presents a solution to the process of authenticating individuals through face recognition for use in public safety. One of the main current security problems is the result of the release of the prison of dangerous individuals through semi-open mode, where civil society mixes with inmates without the possibility of recognizing them and understanding their dangerousness. To solve this problem, it is proposed to implement a facial pattern recognizer using Python language integrated with the Pandas, OpenCV libraries, building a neural network as artificial intelligence to highlight faces registered in the prison system in order to search, identify and identify them. meet them in a public environment.*

Resumo. *Esse projeto apresenta uma solução para o processo de autenticação de indivíduos, através do reconhecimento facial, para utilização na segurança pública. Um dos principais problemas atuais na segurança é o resultado da liberação do cárcere de indivíduos perigosos através da modalidade semia-berto, onde a sociedade civil mistura-se com apenados sem a possibilidade de reconhecê-los e entender sua periculosidade. Para resolver esse problema, propõe-se implementar um reconhecedor de padrões faciais utilizando linguagem Python integrada com as bibliotecas Pandas, OpenCV, construindo uma rede neural como inteligência artificial para destacar rostos cadastrados no sistema prisional a fim de procurá-los, identificá-los e encontrá-los em ambiente público.*

1. Introdução

No ano de 2015, engenheiros chineses em coparticipação com a Universidade Tsinghua (Pegim) e a empresa de segurança *Tzekwan Technology* criaram um protótipo de caixa eletrônico com reconhecimento facial com a finalidade de redução nos crimes envolvendo o uso de serviços bancários [Carneti 2015]. Um outro exemplo de tecnologia facial foi utilizada em Santa Maria-RS, onde para evitar fraudes no uso das passagens de ônibus, foi instalado um sistema de reconhecimento facial para certificar que a carteira de passagem está sendo usada pela pessoa certa [Globo 2010].

Em 2018, um caso mostra o uso da tecnologia a favor de segurança no norte do Brasil, quando um indivíduo foragido há 21 anos era acusado da morte de um publicitário na cidade de Natal-RN, foi localizado após as filmagens de um estabelecimento

comercial serem analisada por um softwares do Laboratório forense Computacional do (Gaeco). [POTIGUAR 2018]

Câmeras instaladas em óculos dos policiais são capazes de identificar, a câmera e capas de identificar todos os rostos dos transeuntes, e esta informação e comparada com uma base de dados que buscas coincidências e esta busca aparece instantânea em um dispositivo móvel semelhante a um *tablet*. Com ajuda da desta tecnologia sete pessoas foram presas em fevereiro por diversos crimes, e 26 passageiros fixados por portar documentação falsa o sistema foi testado com sucesso segundo as autoridades na província de *Henan*, na estação do trem-bala [Carneti 2015]

2. Proposta

Este trabalho tem como objetivo desenvolver um sistema que irá auxiliar no reconhecimento de rostos de pessoas junto a base de dados fotográfica de procurados da segurança publica, em particular da cidade de Santa Maria-RS, a partir do cadastro efetuado tanto na SUSEPE quanto na emissão de documentos e registro de Boletins de Ocorrência (BO) pela Polícia Civil (PC). A base de informações de procurados pela PC é sigilosa, logo a proposta do trabalho irá validar a ideia com imagens de livre acesso, obtidas pela internet ou mesmo geradas pelo próprio autor, consistindo em uma base de dados de treinamento para o algoritmo de Inteligência Artificial, na área de Visão Computacional. O desenho é mostrado na 1, representando de modo abstrato, os componentes da solução proposta.

2.0.1. Objetivos Específicos

- Desenvolver um software capaz de extrair imagens estáticas de vídeos, para tratamento e reconhecimento de padrões de formas geométricas;
- Mapear os padrões de formatos geométricos que formam uma face capturada em imagem;
- Gerar uma base de padrões geométricos reconhecidos e classificados, utilizando técnica de visão computacional;
- Comparar vídeos ao vivo (ou gravado), *frame a frame*, com as coordenadas da base de padrões;
- Avaliar a acurácia de software em relação ao reconhecimento.
- Construir uma funcionalidade de alerta, quando houver identificação positiva de face no sistema, aos agentes de segurança pública

3. Referencial Teórico

Este capítulo tem como objetivo descrever as etapas do processo de reconhecimento facial, que podem ser divididas nos seguintes grupos: Detecção de faces, Extração de características e Representação da Face, Reconhecimento e Verificação. Um sistema automatizado é capaz de receber como entrada uma imagem ou vídeo, identificar as faces presentes, caracterizá-las matematicamente, compará-las com outras previamente cadastradas em um banco de dados e caso haja alguma correspondência, reconhecimento facial totalmente informar tem como saída, qual face do banco de dados condiz à imagem de entrada. A Figura 2 descreve um sistema de reconhecimento facial.

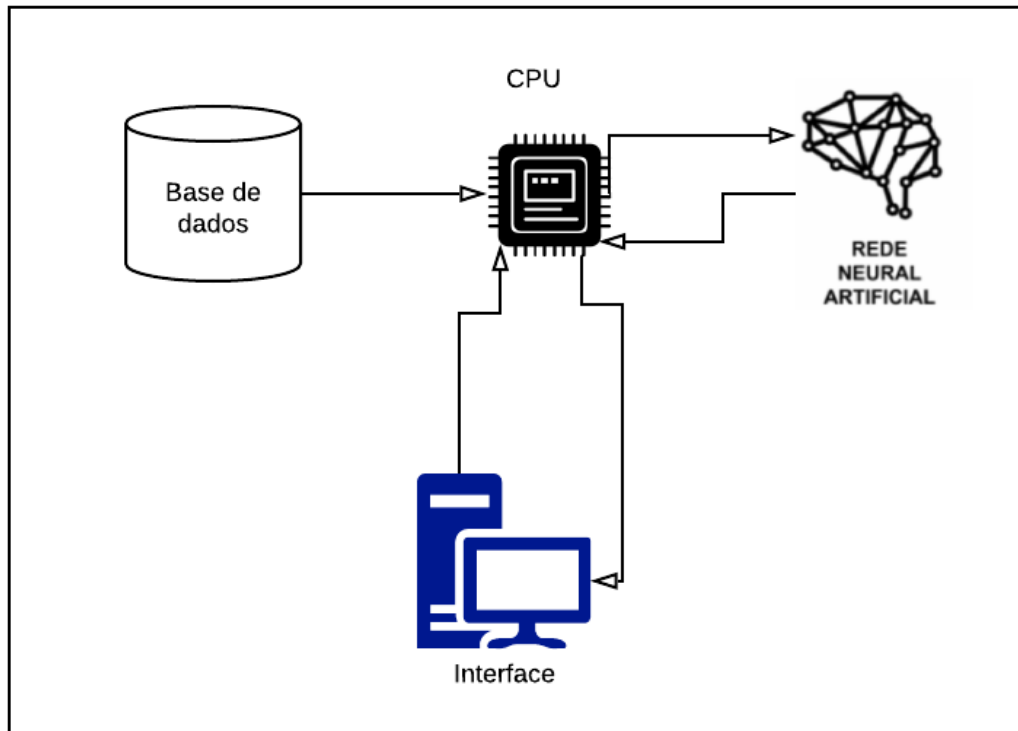


Figura 1. Proposta do Sistema

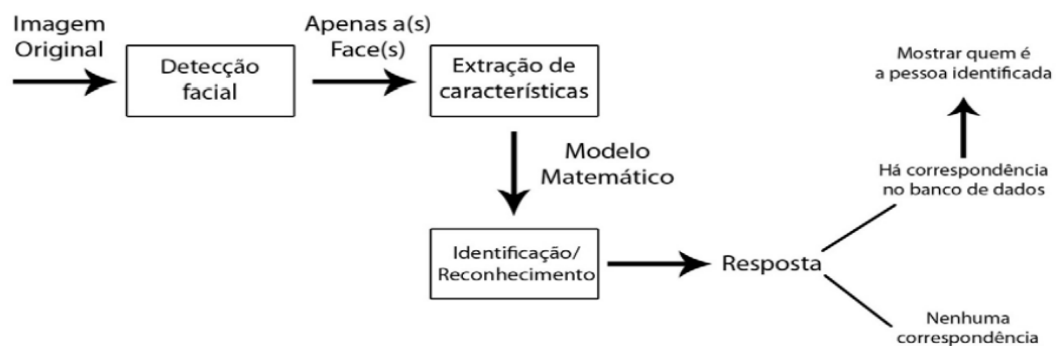


Figura 2. Sistema de reconhecimento facial

3.0.1. Reconhecimento de Padrão por *Machine Learning*

Reconhecer significa conhecer de novo, e isto implica num processo onde existe algum conhecimento prévio e algum tipo de armazenamento do conhecimento sobre o assunto esta é a parte onde os sistemas de visão computacional possuem uma intersecção para fazer o reconhecimento um sistema de visão computacional necessita de uma base de conhecimento dos objetos a serem reconhecidos, esta *database* um conjunto de amostras dos objetos a serem reconhecidos utilizando técnicas de aprendizado de máquina [Marenconi and Stringhini 2009]. Reconhecimento de objetos é uma área fundamental na visão computacional e está profundamente relacionada com o reconhecimento de padrões apresentam padrões de textura, formas, cores, dimensões entre outros e com o reconhecimento individual desses padrões é possível reconhecer o objeto como um todo, os processos de reconhecimento de padrões podem ainda ser uma mistura das apresentado em [Cândido et al. 2009], é utilizado um modelo estrutural padrão de reconhecimento das regiões da boca, olhos e testa.

3.0.2. Python e OpenCv

Python é uma linguagem de programação de uso geral iniciada por Guido van Rossum, que se tornou muito popular em pouco tempo, principalmente devido à sua simplicidade e legibilidade do código [Songini 2005]. Permite ao programador expressar suas ideias em menos linhas de código sem reduzir a legibilidade.

Python se tornou popular por vários motivos, um dele e seu modo simples e uma sintaxe que é como um pseudocódigo, sua modularidade, seu design orientado a objetos, é criação de perfil, portabilidade, teste e recursos de auto documentação, e a presença de um grande número biblioteca permitindo o armazenamento e manuseio efetivo de enormes quantidades de informação numérica. Python é gratuito, como muitas outras linguagens de *script* e pode ser executado na maioria dos computadores modernos. Além disso, fornece estruturas de dados de alto nível, como matrizes associativas, listas, ligação dinâmica, módulos de digitação dinâmica, memória automática gerenciamento, classes, exceções etc. Ele possui um pequeno kernel e pode ser estendido importando bibliotecas externas [Martins and Mould 2017]. Sua distribuição contém uma grande biblioteca de extensões padrão, escrita em python e outras linguagens como C ou C ++, para operações como como expressões regulares do tipo Perl, manipulação de strings, utilitários relacionados à Web, serviços de sistema operacional, testes e ferramentas de criação de perfil, depuração etc. O idioma pode ser estendido criando novos módulos. Existem muitos algoritmos complexos envolvidos na IA, *deep learning* etc. Python envolve muito menos codificação, em termos de linhas de código, entre muitas linguagens de programação, que podem ser usado para o desenvolvimento de muitas dessas aplicações. Esse recurso permite testes fáceis e, portanto, os desenvolvedores podem se concentrar mais em programação real. Python usa tanto quanto 1/5 de código em comparação com outras linguagens OOPs para implementar o mesmo lógica. Mostra o tamanho médio do projeto em milhões de linhas de código para vários idiomas. tamanho médio do projeto por idioma milhões de linhas de código omo visto na Figura python é apenas o segundo após ruby com respeito às linhas a codificar. Portanto, esta é a primeira razão por que o Python é preferido por muitas empresas e Projetos baseados em IA. Possui palavras-chave menores, estrutura simples

e uma sintaxe claramente definida. Assim, o código python é altamente visível para o olhos. Além disso, a alta legibilidade incentiva a colaboração levando a mais contribuição para um projeto Python de código aberto, levando a um rápido ciclo de desenvolvimento. Também tem sido visto que os usuários de *python* são mais fiéis à linguagem como comparado aos usuários da linguagem R(*ruby*) [Nagpal and Gabrani 2019] . As tendências de lealdade em caso de usuários python são bem mais expressivas que em outras linguagens como as citadas acima.Figura 3.

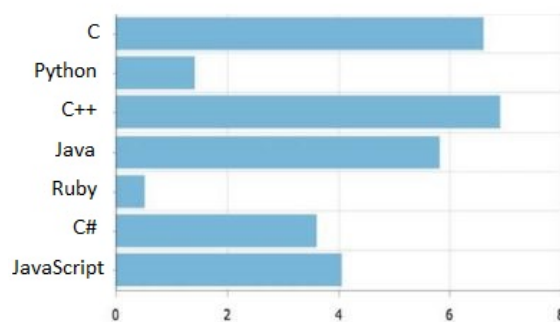


Figura 3. Milhoes de linhas de Codigo

3.1. Trabalhos Correlatos

Esta subseção apresenta os trabalhos correlatos que serão utilizados como base para o desenvolvimento deste trabalho. Os trabalhos, em questão, utilizam diversas abordagens para realizar a captação, identificação e segmentação por meio de técnicas de processamento de imagem.

3.1.1. Uma Análise do Processo Reconhecimento Facial

Neste projeto [Queiroz de Santana et al. 2014] faz um relato comparativo de reconhecimento da face humana por um ser humano e através de sistemas computacionais. O processo de reconhecimento resume-se na execução de vários passos como aquisição de imagens, detecção de faces, a segmentação, extração de características e somente após estas etapas faz a classificação da face. As características mais exploradas serão os olhos, boca e as sobrancelhas, para resolver o problema da dimensionalidade foi utilizado o método PCA (Análise de Componentes Principais) um método descrito como o melhor para extração de características linear, e ainda melhor se utilizado várias imagens do mesmo indivíduo, também foi utilizado o método de classificação K-NN ou popularmente conhecido como K-vizinhos onde existem diversas métricas para calcular a distância entre dois pontos como distância *Mahalanobis*, Euclidiana entre outras.

Para verificação dos resultados foi analisado um banco de dados com 230 imagens com 10 imagens de diferentes posições como o tamanho de 92x112 de 23 indivíduos, e foi aplicada para este teste distância *Mahalanobis* e *Euclidiana* onde obtiveram um índice médio nas duas classes de 76,42% neste projeto foi usado um *framework* Web que segundo [Queiroz de Santana et al. 2014] e um índice aceitável pelo tipo de dados coletados.

3.1.2. Um Método de Aquisição de Imagens para Reconhecimento de Rosto e implementação de um atendimento automático Sistema de Eventos

Em [Lung et al. 2019] propõe um método de aquisição de imagem para criar uma base de dados chamado *Smart Event Faces* que contém quadros de vídeo capturados. possui duas pastas a primeira é a *event-aces* e a segunda é *asmart-aces*. Cada pessoa está contida nas duas pastas para um total de 52 pessoas.

Os vídeos dos *smartphones* são para treinamento e os vídeos do *Raspberry Pi* são para o teste, após localizada os rotos as fotos são recortadas e redimensionado para 140x140, este conjunto de dados contém 52 pessoas. Quarenta e nove homens e três mulheres. Todas elas eram estudantes de graduação que tinham entre 18 e 30 anos. Para cada pessoa, há 50 imagens na pasta *smart faces* e 50 imagens na pasta *event faces*. Ambos os envelopes têm as mesmas etiquetas. Em total, a base de dados *Smart Event Faces* tem 5200 fotos.

3.1.3. Um sistema integrado de detecção e reconhecimento de faces

[Koh et al. 1999] mostrou em seu trabalho que com uso do *haar cascade* juntamente co *Python*, *OpemCv* na alise de *selfie* de usuários da rede social *Instagram*, onde o autor utilizou os marcadores chamado *hashtag*, *Instagram* é uma das mais rápido crescimento mídias sociais nos últimos anos com 1 bilhão de usuários em junho 2018, que aumentou de 600 milhões de usuários em abril 2017, Indonésia onde foi feita a pesquisa é um país com os maiores usuários do Instagram na Ásia-Pacífico, com um total de 45 milhões de usuários. Com isso o projeto consiste em detectar se em *Selfie* marcadas com as respectivas *hashtag* #selfie, #selfiee, #selfies, #candiprambanan, #tamansarijogja, #tebengbrel. Para extração de imagem foi utilizada uma biblioteca chamada *Beautifulsoup*, dela é obtido um arquivo JSON com os dados necessário para o estudo. O método de *Haar* cada recurso produz um valor único que é calculado subtraindo o número de *pixels* sob o retângulo branco do número de *pixels* sob o retângulo preto. Para detectar rostos humanos a digitalização e feita da parte superior esquerdo e termina do inferior direita, para apurar a precisão do sistema foi propostos os seguintes parâmetro

- TP (verdadeiros positivos) indica a quantidade de faces detectadas corretas (faces detectadas como faces)
- TN (negativos verdadeiros) indica a quantidade de faces detectadas corretas (sem faces detectadas como não faces)
- FP (falsos positivos) indica a quantidade de rostos detectados incorretamente (rostos detectados como não-rostos)
- FN (falsos negativos) indica a quantidade de rostos detectados incorretamente (não-rosto detectado como rostos).

$$Precisão = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

com este metodo deu-se uma precisão de 71,48% e nota-se um elevado índice de falso positivos(FP), com isso foi observado o valor preditivo positivo definido(PPV)

$$PPV = TP / (TP + FP) \times 100\% \quad (2)$$

A questão foi que este valor obtido com o PPV foi ainda menor que o anterior tem como resultado 64,65% de precisão. Os dados foram analisados em um computador *Desktop*

com processador *Intel (R) Celeron (R) CPU N3060 @ 1,60 GHz* CPU e uma memória RAM interna de 4 GB e uma conexão com a internet.

4. Metodologia

4.1. eXtreme Programmin (XP)

XP como e conhecido segundo [Beck et al. 2001] e [Nunes 2017] uma metodologia leve para times de tamanho pequeno a médio a equipe XP é composta de dois a 10 desenvolvedores chegando no máximo em 12 se este for mais pede-se que seja dividida em duas equipes, pois, a comunicação começa a ser reduzida XP desenvolve software na presença de requisitos vagos que se modificam rapidamente, e uma metodologia de software ágil utilizado na busca pela qualidade para que se atenda às necessidades do cliente a extreme programa busca o máximo de valor a cada dia de trabalho da equipe em um curto espaço de tempo, o cliente terá um produto para ser avaliado podendo aprender com o mesmo e reavaliar se o que foi desenvolvido é realmente o desejado. A empresa que deseja realmente utilizar *eXtreme Programmin* deve respeitar e utilizar um conjunto de valores e práticas que visam o sucesso em suas operações é justifiquem o investimento feito pelo cliente, visto na Figura 4.

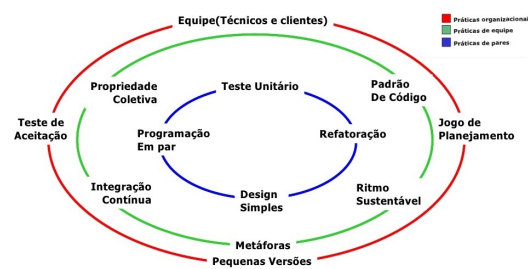


Figura 4. Práticas XP

4.1.1. Bases do XP

Os valores são as diretrizes utilizadas da extreme programa eles definem as atitudes das equipes e as principais prioridades da metodologia que são:

- *feedback*
- Comunicação
- Simplicidade
- coragem

No *feedback* o cliente reavalia o produto recebido frequentemente com isso pode realimentar a equipe de desenvolvimento com as suas reais necessidades com *feedback* do cliente com o desenvolvimento do seu produto estabelece prioridade me informa aquilo que é realmente importante, na comunicação para o *feedback* entre cliente desenvolvedor possa ser efetuado com sucesso e necessário ter uma boa comunicação entre eles Extreme programa prega que a comunicação ocorra da forma mais direta e eficaz possível oferecendo agilidade aos assuntos tratados, recomenda -se o contato direto face a face entre

cliente e desenvolvedor para evitar qualquer tipo de especulação ou mal entendido entre as partes, para que possíveis dúvidas possam ser resolvidas de imediato além de sanar as dúvidas no momento o cliente deverá estar disponível para a equipe ou mesmo presente no ambiente de trabalho da empresa, isto fará com que a equipe entenda o sistema e engrandecerá os relacionamentos pessoais criando um elo de parceria e confiança mútua, e simplicidade para que o cliente consiga dar *feedback* necessário a equipe não basta apenas uma boa comunicação é necessário que os desenvolvedores implementem da forma mais simples possível, o que o cliente deseja o desenvolvedor deve implementar apenas o necessário para que o cliente tenha o seu pedido atendido e dando *feedback* instantaneamente evitando pressuposições, ser simples é um ato de consciência absoluta precisão e por fim coragem por ser um processo de desenvolvimento baseado em diversas premissas de que contrariam o modelo tradicional a Extreme programa exige que os desenvolvedores tenham coragem para manter as premissas requeridas pela metodologia revisando do seu trabalho e mudando aquilo que for necessário.

5. Teste de validação

Nessa seção será citados os requisitos para validar se o produto final deste trabalho será um *software* que cumpre com os objetivos do projeto. E portanto para verificar isso, a validação se baseia nos acertos referente a classificação das imagens de rostos recebidos pela interface para validação iremos verificar imagens de indivíduos na posição frontal, perfil esquerdo e direito, com um fundo unicolor para imagens fixas. Ainda para verificar melhor precisão faremos a aquisição das imagens em formato de Vídeo deste iremos destacar 50 fotos diferentes para uma melhor confiança pois quanto mais informações destacamos de um indivíduo melhor será a performance do *software* exemplo na Figura 5.



Figura 5. Sistema de reconhecimento facial

6. Cronograma

Para execução das atividades a serem realizadas até a defesa do Trabalho Final da Graduação teremos as seguintes etapas, conforme Tabela 1 abaixo.

1. Revisão bibliográfica
2. Definição da metodologia de desenvolvimento
3. Definição dos testes de validação
4. Escrita de texto do TFG I
5. Apresentação TFG I
6. Implementação e testes Escrita do texto TFG II e Apresentação TFG II

A Tabela 1 Tabela abaixo mostrar o cronograma do TFGI e TFGII.

Tarefa	Jun	Ago	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun
1	X	X	X	X								
2		X	X	X								
3					X	X						
4				X	X	X						
5						X						
6				X	X	X	X	X	X	X		
7									X	X	X	
8												X

Tabela 1. Fonte: Própria (2019)

7. Conclusões e perspectivas

Este projeto apresentou a proposta de uma Interface para auxiliar o reconhecimento de faces humanas, abordamos assuntos como processamento de imagem digitais e redes neurais, reconhecimento de padrões. Dentro do contexto abordado proposto, é possível destacar algumas vantagens, tendo como principal objetivo disponibilizar para a comunidade junto a segurança publica no auxilio e agilidade no reconhecimento e buscas de indivíduos, inseridos no meio da população

Para realizar a implementação do sistema, foi decidido utilizar a linguagem de programação *Python* para construir todo o sistema juntamente com *OpenCV*, Classificador *HaarCascade* inicialmente, esse método foi dado por Paula Voila e Michael Jones [Viola et al. 2001].

Para uma visão mais técnica do referido sistema reunimos com o Delegado regional de Santa Maria/RS Sandro Meinerz, que nos apresentou como es feita hoje uma prisão e como são feitas as caracterização e atualização dos dados cadastrais de indivíduos que são detidos, Sandro também relatou a inoperância de um sistema como este proposto para a segurança, tendo em vista que que varias cidades estão aderindo a uma sistema de cercamento eletrônico, que não possui um sistema como o proposto.

Referências

- Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., et al. (2001). Manifesto para desenvolvimento ágil de software. *Retirado em*, 20. Acessado em 24 set 2019.
- Cândido, J., Marengoni, M., et al. (2009). Combining information in a bayesian network for face detection. *Brazilian Journal of Probability and Statistics*, 23(2):179 195.
- Carneti, K. (2015). Polícia chinesa usa óculos com reconhecimento facial para identificar suspeitos disponível=<https://exame.abril.com.br/tecnologia/china-desenvolve-caixa-eletronico-com-reconhecimento-facial>. Acesso 30 ago.2019,.
- Globo, G. (2010). Para evitar fraudes, ônibus de santa maria têm reconhecimento facial. <http://g1.globo.com/rs/rio-grande-do-sul/noticia/2016/03/para-evitar-fraudes-onibus-de-santa-maria-tem-reconhecimento-facial.html>. Acesso 27 ago.2019.
- Koh, L. H., Ranganath, S., Lee, M. W., and Venkatesh, Y. (1999). An integrated face detection and recognition system. In *Proceedings 10th International Conference on Image Analysis and Processing*, pages 532–537. IEEE.

- Lung, L. F., Barua, M. N., and Juarez, P. S. (2019). An image acquisition method for face recognition and implementation of an automatic attendance system for events. acessado em 20 Out 2019.
- Marengoni, M. and Stringhini, S. (2009). Tutorial: Introdução à visão computacional usando opencv. *Revista de Informática Teórica e Aplicada*, 16(1):125 160.
- Martins, J. and Mould, S. T. (2017). Introdução ao python para alunos de engenharia de polímeros. Acessado em 02 out 2019.
- Nagpal, A. and Gabrani, G. (2019). Python for data analytics, scientific and technical applications. In *2019 Amity International Conference on Artificial Intelligence (AICAI)*, pages 140–145. IEEE.
- Nunes, R. D. (2017). A implantação das metodologias ágeis de desenvolvimento de software scrum e extreme programming (xp): uma alternativa para pequenas empresas do setor de tecnologia da informação. *ForScience*, 4(2).
- POTIGUAR, T. (2018). Com técnica de reconhecimento facial, acusado de matar publicitária em natal é preso 21 anos após o crime. url<https://terrapotiguar.com.br/com-tecnica-de-reconhecimento-facial-acusado-de-matar-publicitaria-em-natal-e-preso-21-anos-apos-o-crime>. Acessado em 14 set 2019.
- Queiroz de Santana, L. M., Rocha, F., and Santos, T. (2014). Uma análise do processo reconhecimento facial. *Cadernos de Graduação: ciências exatas e tecnológicas*, 2:49 58.
- Songini, M. L. (2005). Put in plain language: The high portable, object oriented python language moves into enterprise application development. *Computerworld*, 12.
- Viola, P., Jones, M., et al. (2001). Rapid object detection using a boosted cascade of simple features. *CVPR (1)*, 1(511-518):3.