

# SISTEMA DE RECONHECIMENTO FACIAL COM BASE EM TÉCNICAS DE APRENDIZADO DE MÁQUINA

Vagner Vieira dos Santos<sup>1</sup>, Daniel da Conceição Pinheiro<sup>2</sup>

Universidade Federal do Pará – UFPA – Campus Tucuruí / FECOMP – Faculdade de Engenharia de Computação

vagner.santos@tucurui.ufpa.br<sup>1</sup>, dpinheiro@ufpa.br<sup>2</sup>

**Abstract:** *The work proposes studies in the field of machine learning for facial recognition. The implementation of a personal identification system is sought to compare information flows contained in datasets obtained from the processing of videos and photos of individuals as a mechanism for fraud prevention. Neural Networks technology is used to create embeddings (vector representation of facial features) of faces by extracting faces from photos. Thus, models previously trained in massive databases, such as OpenFace and Facenet, are used, as they are validated and demonstrate scientific results with a high rate of effectiveness. It is worth mentioning that the procedures were applied and compared with different classifiers such as Support Vector Machine (SVM), k-Nearest Neighbors (KNN) and Convolutional Neural Network (ConvNet). The acquired results were based on the comparison process between Python programming via Jupyter Notebook and Orange Software version 3.29.3. The tests carried out showed positive and promising results.*

**Resumo:** O trabalho propõe estudos na área de aprendizado de máquina para reconhecimento facial. A implementação de um sistema de identificação pessoal visa comparar fluxos de informação contidos em conjuntos de dados obtidos a partir do processamento de vídeos e fotos de indivíduos como mecanismo de prevenção de fraudes. A tecnologia de redes neurais é usada para criar *embeddings* (representação vetorial de características faciais) de rostos, extraíndo rostos de fotos. Assim, são utilizados modelos previamente treinados em bancos de dados massivos, como OpenFace e Facenet, pois são validados e demonstram resultados científicos com alto índice de eficácia. Vale ressaltar que os procedimentos foram aplicados e comparados com diferentes classificadores como *Support Vector Machine* (SVM), *k-Nearest Neighbors* (KNN) e *Convolutional Neural Network* (ConvNet). Os resultados adquiridos foram baseados no processo de comparação entre programação Python via *Jupyter Notebook* e *Orange Software* versão 3.29.3. Os testes realizados mostraram resultados positivos e promissores.

## 1. Introdução

Reconhecimento facial é uma forma de identificar ou confirmar a identidade de uma pessoa usando seu rosto. Os sistemas de reconhecimento facial podem ser usados para identificar pessoas em fotos, vídeos ou em tempo real. Segundo J. B. Alam et al. (2019), o reconhecimento facial é primordial em diversas aplicações como, por exemplo,

sistemas de segurança em monitoramento de pessoas em tempo real e sistemas de cibersegurança.

Na área de segurança, é muito comum a aplicação de conceitos relacionados à análise biométrica, em que características físicas e comportamentais de indivíduos são examinadas e identificadas unicamente. A área de biometria possui um vasto estudo sobre diversificados métodos de biometria, tais como, reconhecimento de faces, íris, assinaturas, impressões digitais, etc. Atualmente as técnicas de reconhecimento facial mais utilizadas e de melhor performance são as técnicas baseadas em redes neurais convolucionais. As redes neurais convolucionais (CNNs / ConvNets) FaceNet (Google), OpenFace (CMU), DeepFace (Facebook), são exemplos de sistemas que possuem alta precisão para a tarefa de reconhecimento de faces.

O modelo de rede neural artificial fundamental é conhecido como rede neural convolucional (CNN), na qual é uma combinação entre Processamento Digital de Imagens (PDI) e Redes Neurais Artificiais (RNA). O objetivo de uma rede CNN é aprender características de objetos para que se possa classificá-los em cenas diversas representadas através de fotos, vídeos ou até mesmo em tempo real. O processo de aprendizagem de uma CNN se resume em treinar a rede com imagens que contenham os objetos que ela precisa reconhecer. Para que a rede seja capaz de extrair as características que julga necessária para o reconhecimento do objeto, é importante que as imagens de treinamento contenham o objeto em diferentes situações, como ângulo, escala, foco, ambiente etc (NIELSEN, 2015).

Este trabalho apresenta os resultados do desenvolvimento de algoritmos para identificar, em tempo real, rosto de indivíduos através de redes neurais convolucionais (*Convolutional Neural Network*) tendo em vista elementos faciais, especificamente, formatos e dimensões de rosto, boca, nariz e olhos.

Diante deste contexto, este projeto visa responder a seguinte questão: - **É possível desenvolver com baixo custo e alta qualidade um sistema de algoritmos que juntos são capazes de possibilitar o reconhecimento com perfeição das faces apresentadas por indivíduos, através de parâmetros faciais característicos de cada pessoa?**

O objetivo geral do artigo consiste no desenvolvimento dentro do contexto de visão computacional, aprendizagem de máquina (*machine learning*), um sistema de reconhecimento facial a partir da utilização de redes neurais convolucionais.

Destaca-se os seguintes objetivos específicos: realizar o reconhecimento que utiliza características a partir de imagens redimensionadas em resoluções padrões 160x160 pixels; gerar *embeddings* de imagens; validar o método para geração de resultados por meio do uso de *Deep Learning* profundo com o TensorFlow<sup>3</sup>; implementar estratégias e métodos comparativos de desempenho entre o *Orange DataMining* e programação Python via *Jupyter*<sup>4</sup> *Notebook* com os classificadores *Support Vector Machine* (SVM), *k-Nearest Neighbors* (KNN) e *Rede Neural Convolucional* (CNN).

3 <<https://www.tensorflow.org/>>

4 <<https://jupyter.org/>>

## 2. Trabalhos Correlatos

São relatados a seguir alguns trabalhos com abordagem similar à proposta neste artigo, e os respectivos resultados obtidos para efeitos de comparação.

Nunes (2016) propõe um trabalho que realiza um processo de reconhecimento facial de posição frontal de 31 pessoas, em que cada sujeito apresenta 3 imagens de profundidade e 3 imagens de cor (RGB). As imagens de cor são utilizadas para detecção facial por uso de um *HaarCascade*, que possibilita a extração dos pontos da face da imagem de profundidade os quais formam uma nuvem de pontos tridimensional. A partir desta nuvem de pontos são extraídas a intensidade normal e a intensidade do índice de curvatura de cada ponto que permite a formação de uma imagem bidimensional, nomeado como mapa de curvatura. O algoritmo apresentou resultados com um valor de *F-Measure* de até 94%, o qual contém uma baixa taxa de falsos reconhecimentos.

Pires et al. (2020) apresentam um estudo no campo de *Machine Learning* (ML) para reconhecimento facial e implementação de um sistema de autenticação pessoal que compara documentos com vídeos e fotos de cada sujeito. São utilizadas as tecnologias de Redes Neurais para gerar representações vetoriais (*Embeddings*) com detalhes geométricos faciais. Modelos já treinados em bases de dados massivas são empregadas nesse trabalho. Com a análise detalhada da rede FaceNet foi possível desenvolver um sistema com uma acurácia de 99,37%. O projeto gera valor ao passo que executa uma aplicabilidade para evitar fraudes de forma barata e capaz, em detrimento de sistemas caros e complexos, hospedados em produtos de custo elevado e de código fechado.

Diniz (2013) relatou a utilização de técnicas de Análise dos Componentes Principais (*Principal Component Analysis* - PCA), para reconhecimento de identificação facial e um Sistema de Codificação de Ação Facial (FACS), para rastreamento dos pontos característicos faciais, e um método de classificação da emoção baseado em regras para reconhecimento de expressões faciais. O sistema, por meio de uma câmera, permite identificar o aluno durante toda a realização das avaliações pedagógicas, além de reconhecer a emoção (alegria, tristeza, raiva e aversão) do aluno pela sua expressão facial. A abordagem atingiu uma taxa total de acurácia de reconhecimento de emoção de 85,5 %.

## 3. Metodologia

Nesta seção são apresentados os procedimentos metodológicos utilizados para desenvolvimento do trabalho. Estes foram divididos em cinco etapas que serão descritas a seguir: obtenção e seleção de fotos; recorte e extração de faces; vetorização das imagens e geração de *embeddings*; análise preditiva com base em algoritmos de *machine learning*; detecção e reconhecimento facial em tempo real. A metodologia deste trabalho se baseia em detectar tanto faces nos quadros de um vídeo executado quanto em tempo real através de uma *webcam*, enquanto relaciona as características faciais detectadas com conjuntos de informações extraídas pelo processo de treinamento. A Figura 1 resume a metodologia utilizada.

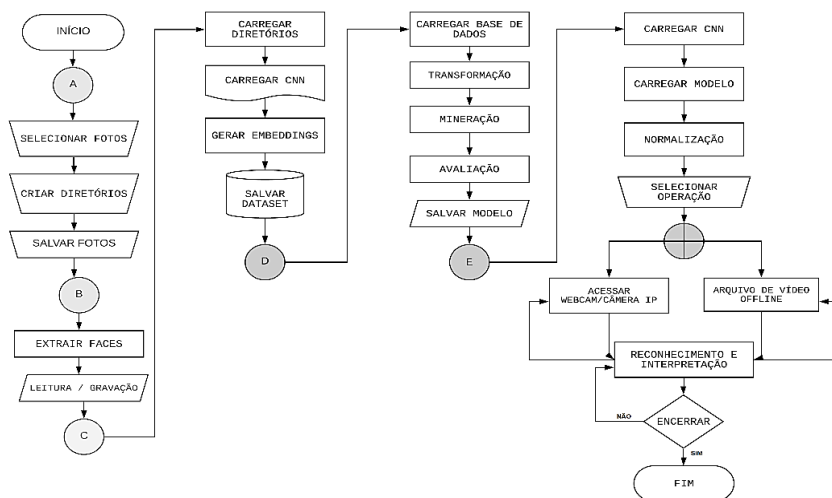


Figura 1. Diagrama de fluxo da metodologia do sistema

### A. Obtenção e seleção de fotos

A etapa constituiu na obtenção total de 25.000 imagens que incluem o rosto de 5 voluntários, que participaram deste estudo, e imagens de pessoas aleatórias ou desconhecidas obtidas no repositório de imagens da Google e no site *df11faces*<sup>1</sup>. As imagens utilizadas nesse processo são majoritariamente imagens de alta resolução, aproximadamente 70% são de imagens de tamanho de  $\approx 1\text{MB}$  e resolução vertical e horizontal de 72 DPI. As imagens correspondentes aos voluntários e pessoas aleatórias foram armazenadas localmente em diretórios. Esta divisão foi realizada para possibilitar o desenvolvimento de testes e a geração de *datasets* com informações relativas às imagens selecionadas. Na tabela a seguir são apresentadas essas distribuições.

Tabela 1. Fluxograma da distribuição das bases de imagens

| Quant. de fotos dos voluntários | Quant. de fotos de pessoas aleatórias | Total  |
|---------------------------------|---------------------------------------|--------|
| 1055                            | 1445                                  | 2500   |
| 1055                            | 3945                                  | 5000   |
| 1055                            | 6445                                  | 7500   |
| 1055                            | 8945                                  | 10.000 |
| 1055                            | 11.445                                | 12.500 |
| 1055                            | 13.945                                | 15.000 |
| 1055                            | 16.445                                | 17.500 |
| 1055                            | 18.945                                | 20.000 |
| 1055                            | 21.445                                | 22.500 |
| 1055                            | 23.945                                | 25.000 |

### B. Recorte e extração de faces

Elaborou-se o algoritmo para detecção de rosto baseado na biblioteca MTCNN (*Multi-task Cascaded Convolutional Neural Networks*) e capaz de reconhecer, redimensionar e padronizar as imagens, além de colaborar significativamente na minimização do tempo despendido para realização da extração e recortes faciais.

A extração de características tem tanto o objetivo de extrair os padrões mais relevantes da imagem/quadro quanto o objetivo de reduzir a dimensionalidade. Ressalta que neste passo as imagens originais são recortadas e redimensionadas em resoluções padrões 160x160 pixels.

A utilização da técnica de aumento de dados consistiu em gerar novas imagens para a base a partir de transformações das imagens presentes na base. Então uma mesma imagem é capaz de gerar várias imagens diferentes para a rede com aplicação de transformações como espelhamento. O aumento de dados de imagem é usado para expandir o conjunto de dados destinado para a etapa em que se realiza o processo de treinamento a fim de melhorar o desempenho e a capacidade de generalização do modelo.

### C. Vetorização das imagens e geração de *embeddings*

Neste processo para detectar os elementos faciais, especificamente, rosto, boca, nariz e olhos, utilizou-se os algoritmos baseados nas redes neurais convolucionais OpenFace e FaceNet para processar imagens e apresentar altas taxas de detecção dos recortes faciais. Busca-se otimizar a criação de *embeddings* de rostos humanos, ou seja, a transformação de rostos em um espaço vetorial Euclidiano de  $d$  dimensões que compara o desempenho de várias abordagens na criação desses *embeddings*, com o tamanho de vetor o qual chega ao número de 128 dimensões,  $d = 128$ . No aprendizado de máquina, esse vetor é chamado de incorporação, pois todas as informações importantes de uma imagem são incorporadas a esse vetor. Basicamente, o processo se baseia em pegar o rosto de uma pessoa e comprimir em um vetor de 128 números. O treinamento do modelo foi realizado através de rostos considerados conhecidos dividido em pastas separadas e identificadas com os nomes de cada indivíduo a ser identificado ou reconhecido.

Optou-se por utilizar 2 CNNs, com o intuito de tornar o reconhecimento eficiente, porque essas redes neurais convolucionais são capazes de detectar pontos de referência faciais, estimativa de pose da cabeça, reconhecimento de unidades de ação facial e estimativa de olhares. Aplicou-se a rede convolucional OpenFace no *Orange DataMining* e a rede convolucional FaceNet no Jupyter Notebook. As ConvNets selecionadas têm a capacidade de apresentarem resultados de última geração em todas as tarefas mencionadas. Para cada elemento facial, as 25000 imagens foram utilizadas como pares entrada/saída para treinar as correspondentes CNNs.

Após o tratamento dos dados de entrada, é necessário dividir os dados da base de dados para a realização do treinamento, validação e teste da CNN. Para isso, é utilizado o método *train\_test\_split()* do módulo de *model\_selection* da biblioteca *Sklearn*<sup>5</sup>, que se trata de uma biblioteca de *machine learning* de código aberto, com suporte para agrupamento de dados e classificação, dentre outras manipulações.

No treinamento do modelo, mais precisamente no que tange a rede neural convolucional, a imagem de entrada é lida em forma de uma matriz de dados, os quais são reduzidos na camada convolutiva mediante a operações de convolução. É importante salientar que se faz necessário a realização desses procedimentos no método

5 <<https://scikit-learn.org/>>

por programação via *Jupyter Notebook* por intermédio da utilização do framework *TensorFlow* como estrutura backend em conjunto com as bibliotecas *Keras*, *Sklearn* e *Pillow*. Essa estrutura altamente flexível e versátil tem a capacidade de auxiliar na criação e treinamento de redes neurais para detectar e decifrar padrões e correlações.

#### D. Análise preditiva com base em algoritmos de *machine learning*

A decisão final do reconhecimento é baseada nas saídas adquiridas a partir de comparações entre o desempenho do software Orange<sup>6</sup> e programação python no *Jupyter Notebook* com o emprego dos algoritmos de classificação *Convolutional Neural Network* (CNN), *Support Vector Machine* (SVM) e *K-Nearest Neighbors* (KNN) para avaliar a capacidade de generalização do conjunto desenvolvido e estimar o quão preciso é o modelo na prática, ou seja, o seu desempenho em um novo conjunto de dados. A avaliação de desempenho possibilita a tomada de decisão em criar o sistema de reconhecimento facial na prática a partir dos resultados comparativos obtidos.

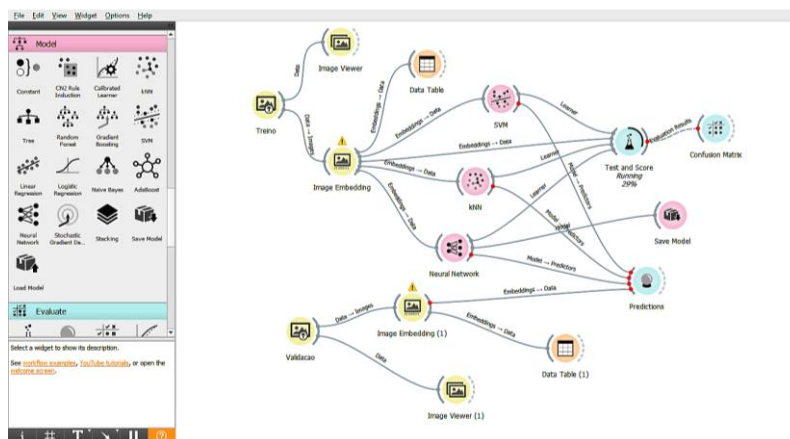


Figura 2. Implementação no Orange: Data Mining Toolbox

Como o problema de classificação de imagens se comporta de forma não linear, é necessário a utilização de um componente matemático após a convolução para introduzir a não-linearidade para rede, chamado de função de ativação. A principal finalidade das funções de ativação é converter um sinal de entrada de um neurônio, em uma rede neural artificial, para um sinal de saída. Essa saída será usada na próxima camada. Essas funções possibilitam que pequenas mudanças e atualizações nos pesos e biases causem apenas uma pequena mudança nas saídas. Quando não temos a função de ativação, os pesos e bias simplesmente fazem uma transformação linear. As funções de ativação tornam possível a propagação posterior desde que os gradientes sejam fornecidos juntamente com o erro para atualizar os pesos e bias. Sem a função não linear diferenciável, isso não seria possível. Dentre as funções de ativação existentes, utilizou-se a Unidade Linear Retificada (ReLU). A principal vantagem de usá-la sobre outras funções de ativação é que ela não ativa todos os neurônios ao mesmo tempo. Se a entrada for negativa, ela será convertida em zero e o neurônio não será ativado. Isso significa que, ao mesmo tempo, apenas alguns neurônios são ativados, os quais tornam a rede eficiente e fácil para a computação. Dessa forma, a função ReLU irá zerar entradas negativas e preservar os valores positivos.

6 <<https://orangedatamining.com/>>

Além da ReLU, a função Softmax é essencial para problemas de classificação e tem como objetivo representar a probabilidade de cada classe da rede para determinado parâmetro de entrada. Por ser uma função de probabilidade, a Softmax força a saída de uma rede neural a representar a probabilidade dos dados serem correspondentes a uma das classes definidas. Sem essa função de ativação as saídas dos neurônios são simplesmente valores numéricos onde o maior indica a classe vencedora.

Após todo o processo de treinamento, a rede neural estará pronta para fazer suas classificações e predições. Vale evidenciar que para utilizar o modelo treinado é necessário salvá-lo. Para isso, deve definir um local para salvar esse arquivo, o arquivo no formato '.h5' que será utilizado para realizar as predições da rede neural no algoritmo elaborado para detecção e reconhecimento facial.

#### E. Detecção e reconhecimento facial em tempo real

Finalmente, uma das etapas mais importantes se baseia em detecção e reconhecimento faces contidas na base de dados. Durante o processo de captura por meio do acesso a webcam ou arquivos de vídeo do (s) indivíduo (s) localmente armazenados, cada imagem é verificada sequencialmente, os rostos são detectados e reconhecidos de acordo com as imagens das pessoas treinadas para serem reconhecidas. Caso nenhum dos rostos identificados condizem com os das pessoas conhecidas pelo sistema, estas são denominadas como *Unknown*, do contrário o nome da pessoa identificada é apresentado. O algoritmo utiliza a biblioteca multiplataforma com diversas funções e metodologias próprias para Visão Computacional, que é o OpenCV (*Open Source Computer Vision Library*).

## 4. Resultados e Discussão

Neste capítulo serão apresentados os resultados obtidos no Capítulo 3, os testes realizados, comportamento do treinamento e outras formas de comprovação dos resultados.

Ressalta-se que posterior ao processo de normalização, os dados de entrada são reduzidos para um vetor de dimensão única pela camada *flatten* para serem treinados por camadas totalmente conectadas. Após cada camada totalmente conectada (*dense*) com função de ativação ReLU foi inserido uma camada de *dropout*. A primeira camada *dense* foi definida com 128 neurônios e *dropout* com taxa 0.5. O *Dropout* tem como vantagem possibilitar a regularização da rede e contribuir para minimização de problemas de sobre-ajuste (*Overfitting*). Os resultados de treinamento relativo ao desempenho quanto a acurácia do processo de classificação de rostos pela Rede Neural é exibida na Figura 1, ou seja, é indicada a performance geral do classificador.

Os valores correspondentes a seguir estão no intervalo entre 2.500 e 25.000 e são baseados no número de dez testes sequenciais de desempenho no intervalo de 2.500 dados. Por se tratar de um projeto de *deep learning* e *machine learning*, especialmente por lidar com reconhecimento facial, foi necessário a diversificação de testes durante todo o desenvolvimento, ou seja, alterna-se os *embeddings* correspondentes a faixa de dados supracitadas. As avaliações foram feitas com o objetivo de explorar o campo de estudo, ajustar parâmetros e orientar tomadas de decisão. Os testes apresentados nas

figuras 3 e 4 foram realizados de modo consecutivo com diferentes classificadores por intermédio de aprendizagem supervisionada.

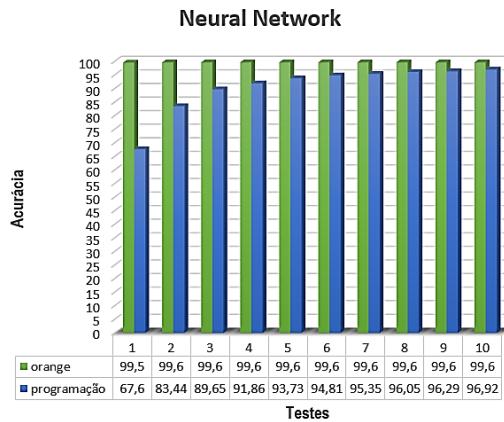


Figura 3. Desempenho da Rede Neural

Conforme pode ser visualizado na imagem acima, o software Orange apresentou uma taxa de reconhecimento de  $\approx 99\%$  numa base de dados ( $25.000 \text{ rows} \times 128 \text{ columns}$ ) a partir da rede neural convolucional OpenFace que combina amostragem de imagens locais e um mapa de redes neurais ajustáveis.

Resultados similares foram obtidos pelos classificadores *Support Vector Machine* (SVM) e *K-Nearest Neighbors* (KNN) que podem ser vistas na Figura 4. Verifica-se que após computar todos os testes, a acurácia atinge valores próximos de 100%, a qual indica que os valores se ajustam conforme o aumento do volume de dados, portanto o algoritmo começa a apresentar resultados mais confiáveis ao longo desse cenário. Como pode ser observado, fica evidente a dependência das redes neurais convolucionais no aumento das quantidades de dados.

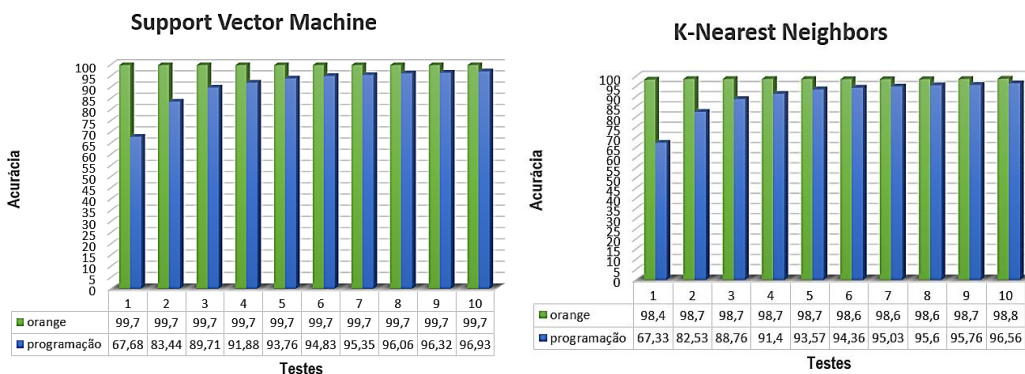


Figura 4. Desempenho dos classificadores SVM e KNN

Na Figura 5 são apresentadas as matrizes de confusão para as bases com imagens reais, em que a ordem dos números no eixo da abscissa e ordenadas correspondem ao número de classes avaliadas. É possível perceber a complexidade do problema de classificação de imagens, mas fica evidenciado que o modelo acertou a grande maioria classificações. O grande problema é que uma determinada classe poder apresentar pequenas semelhanças nas imagens fornecidas com outras 6 classes e esse erro ser distribuído em pequenas porções ao longo da matriz, dessa forma o número de acertos é

consideravelmente. Vale evidenciar que do montante de 25.000 dados, a porcentagem de imagens que foi atribuída para teste ou validação corresponde a 30%, ou seja 7.500 dados.

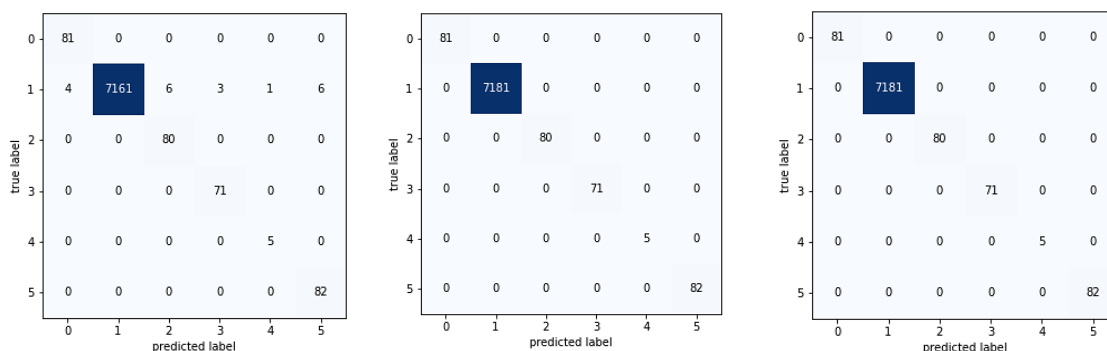


Figura 5. Matriz de confusão dos classificadores KNN, SVM e KNN respectivamente

No que se refere aos elementos faciais, as imagens de cada indivíduo foram utilizadas como pares de entrada/saída para treinar as CNNs. Como deseja-se treinar uma grande quantidade de dados e classes, o treinamento foi dividido em lotes  $batch\_size = 8$  e  $epochs = 54$  em relação aos dados de treino alocados para validação. Verificou-se que as redes neurais convolucionais reduzem o número de parâmetros ajustáveis pela rede, além de agilizar e melhorar o processo de treinamento. Uma outra característica relevante das CNN's é a invariância à escalas, translações e transformações dos sinais de entrada do sistema, que possibilita a realização de reconhecimento de padrões de forma automática e com mais robustez. Ao analisar a arquitetura da FaceNet e OpenFace é possível perceber que quanto mais profunda são as redes, mais parâmetros específicos são passados para a convolução. Ao optar por retreinar todas as camadas ou as primeiras camadas, a rede terá que aprender diversas características globais limitados a base de dados utilizada.

A criação de cada um dos *embeddings* necessitou da execução de alguns procedimentos custosos de processamento de imagem, como a leitura local de cada um dos arquivos, a detecção dos rostos pela MTCNN, o redimensionamento e a criação do *embedding* pela FaceNet. O tempo total para a criação dos 25.000 vetores das imagens selecionadas fica por volta de 59 minutos. Com intuito de avaliar os custos de geração de um modelo para redes profundas em equipamentos do cotidiano optou-se por treinar este modelo em um notebook com CPU Intel Core i7-7500U 2.70GHz (Tecnologia Intel Turbo Boost 3.50GHz), 4MB Cache, Memória RAM de 8 gb e HD 1 TB. Com a análise de dados feita a partir do banco de faces, foi possível entender a aplicabilidade e os limites da rede neural do modelo FaceNet e como ele se comporta em certas situações. Podemos concluir que a rede é poderosa na criação de *embeddings* faciais, uma vez que a foto do rosto tenha sido tirada de maneira adequada

Além de ter estrutura, modelos e dados carregados, também é necessário compilar a rede neural, e passar um algoritmo de otimização. Neste projeto, foi definida uma função para cálculo de perda, chamada *Categorical Crossentropy*, que irá informar o quanto ainda há para aprender sobre a base de dados. Para otimização do modelo foi

utilizado o RMSprop, um algoritmo de otimização baseado em gradiente. Também é nesta etapa que são definidas as métricas para avaliar o desempenho da rede neural. Essa avaliação define se é necessário continuar o treinamento ou se a rede já aprendeu o máximo possível com os dados processados. As métricas utilizadas são a “Accuracy” e “Loss”, como objetivo de validar o quanto a rede neural está sendo efetiva nos resultados. Os resultados deste processo de treinamento podem ser visualizadas na figura a seguir.

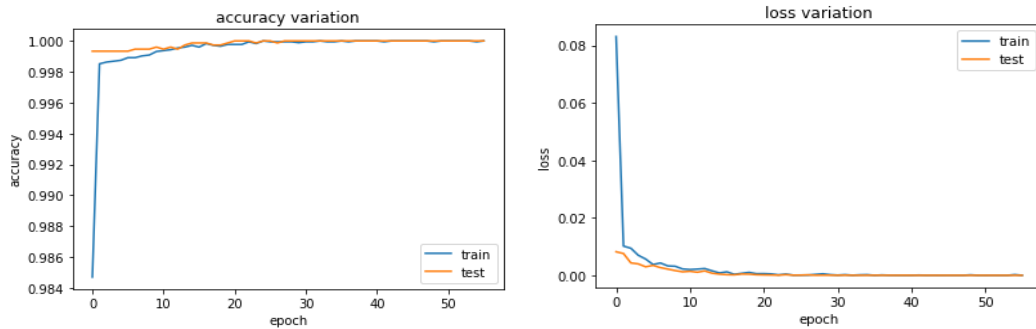


Fig. 6. Treinamento da CNN via Jupyter Notebook

Uma desvantagem em relação ao aumento do volume de dados se dá no desempenho do algoritmo, dado que o tamanho de cada vizinhança (A vizinhança é geralmente uma matriz quadrada) tende a ser maior, e apresenta um maior custo computacional para desenvolver o cálculo de similaridade. Isso acarreta em um tempo maior necessário para geração dos *embeddings* e o posterior processo de treinamento com os valores característicos das faces. Todas as implementações realizadas estão disponíveis em um repositório no Google Drive<sup>7</sup>.

Testes do reconhecimento em tempo real foram realizados. Para realizar o reconhecimento facial em tempo real, os frames funcionam como imagens para modelo de classificação. As figuras 7 e 8 ilustram o reconhecimento correto. Note que os nomes dos indivíduos são exibidos no retângulo verde na parte superior do rosto. O algoritmo de detecção e reconhecimento faces foi construído para capturar imagens frontais, sem óculos e com expressão neutra, e comparar com imagens frontais e invertidas usadas na base de dados. O tempo despendido para compilação do algoritmo para identificação leva aproximadamente 17 segundos. O desempenho varia conforme a configuração do sistema.



Figura 7. Detecção de rosto de 1 pessoa

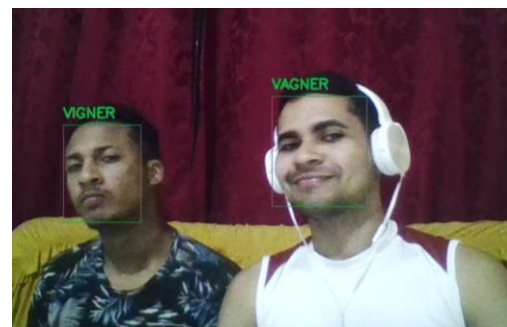


Figura 8. Detecção de rosto de 2 pessoas

7 <[https://drive.google.com/drive/folders/1\\_F1bL1kPI\\_0Nzwnvut1yBSzvRsPY3wKL?usp=sharing/](https://drive.google.com/drive/folders/1_F1bL1kPI_0Nzwnvut1yBSzvRsPY3wKL?usp=sharing/)>

Algumas limitações do algoritmo desenvolvido foram encontradas no decorrer dos testes. É necessário a proximidade entre a câmera e o sujeito para melhor identificação. Movimentos bruscos, utilização de óculos ou qualquer tipo de material que venha cobrir olhos ou bocas dos indivíduos provocam erros durante a identificação. Com estes testes, em vídeos fica evidente que apesar do modelo ser capaz de classificar a maioria dos casos, ele também erra principalmente em mudanças de angulação ou variações na iluminação.

## 5. Considerações Finais

Este artigo apresentou os resultados do desenvolvimento de um algoritmo de reconhecimento facial em tempo real. Elementos importantes da face, tais como: rosto, boca, nariz e olhos, são obtidos pelos algoritmos desenvolvidos. O algoritmo de detecção de faces, portanto, mostrou-se eficiente na taxa de acertos com uma acurácia de  $\approx 97\%$  na comparação de rostos na base de testes, no entanto, esta abordagem mostrou-se ser sensível em imagens com condições de iluminação não controladas. As taxas de acerto de classificação do sistema de reconhecimento facial se encontram entre as taxas de acerto para as implementações de sistemas de reconhecimento facial descritas na literatura.

Independentemente dos resultados, é necessário aumentar a base de imagens para realizar um minucioso procedimento de treinamento e teste do algoritmo. Além disso, foi evidenciado a dependência de uma grande quantidade de dados para que as redes neurais convolucionais sejam capazes de diferenciar o número de classes e generalizar para cenários do mundo real. Por isso, a técnica de aumento de dados foi essencial. A proposta apresentada pode ser aperfeiçoada por meio da implementação de aprimoramentos que possibilitem ampliar a taxa de reconhecimento e reduzir falsas identificações, através da exploração de outras características geométricas do rosto, e por modificações ao processo de reconhecimento.

Mediante o exposto, é possível afirmar que as redes neurais abriram um leque de possibilidades para solucionar problemas que antes não eram possíveis. Cada vez mais aplicações e sistemas são desenvolvidos com o propósito de melhorar a interação homem máquina, o que contribui para uma qualidade de vida cada vez melhor. Busca-se automatizar cada vez mais serviços com o auxílio de algoritmos capazes de realizar predições e adaptações, que antes só eram possíveis com a intervenção humana.

## 6. Referências

ALAM, Jannat Binta et al. **System development using face recognition**. In: 2019 International Conference on Automation, Computational and Technology Management (ICACTM). IEEE, 2019. p. 408-411.

ALPAYDIN, E. (2020). **Introduction to Machine Learning**. Adaptive Computation and Machine Learning series. MIT Press.

DINIZ, Fábio Abrantes. **RedFace – um sistema de reconhecimento de expressões faciais para apoiar um ambiente virtual de aprendizagem.** / Fábio Abrantes Diniz. -- Mossoró, 2013.

NIELSEN, Michael A. **Neural Networks and Deep Learning.** Determination Press, 2015. Disponível em: <http://neuralnetworksanddeeplearning.com>. Acesso em: 07 fev. 2022.

NUNES, L.F.M., (2016). **Reconhecimento Facial Biométrico em Nuvens de Pontos Tridimensionais.** Trabalho de Graduação em Engenharia de Controle e Automação, Publicação FT.TG-n36/2016, Faculdade de Tecnologia, Universidade de Brasília, Brasília, DF, 111p.

PIRES, Caio Augusto Silva. **Reconhecimento facial aplicado a prevenção de fraudes** / C. A. S. Pires, F.F. Lima, M. A. O. Silva -- São Paulo, 2020.

ROLIM, A. L.; BEZERRA, E. P. **Um Sistema De Identificação Automática De Faces Para Um Ambiente Virtual De Ensino E Aprendizagem.** In: XIV Simpósio Brasileiro de Sistemas Multimídia e Web, Vila Velha. 2008.

VIOLA, Paul; JONES, Michael J. **Robusta detecção facial em tempo real.** *Jornal internacional de visão computacional* , v. 57, n. 2, pág. 137-154, 2004.