



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

ESCOLA DE CIÊNCIAS E TECNOLOGIA

RELATÓRIO SOBRE TREINAMENTO COM MACHINE LEARNING: RECONHECIMENTO FACIAL

Discente 1: Maria Eduarda Fernandes Lago

Matrícula: 20180146466

Discente 2: Luciano Alexandre Matrícula:

Docente: Orivaldo Vieira de Santana Júnior

Disciplina: Tópicos de Informática 1

NATAL, 27 de junho de 2023

Resumo

Os recentes avanços na área da Inteligência Artificial possibilitaram o desenvolvimento de tecnologias de reconhecimento facial. Tal tecnologia já é utilizada há algum tempo, no entanto, sua adoção tem aumentado nos últimos anos dada a precisão no reconhecimento com recursos de Inteligência Artificial. Os sistemas de reconhecimento facial vêm evoluindo muito nos últimos anos. Sendo assim neste relatório será descrito como foi feito um sistema de reconhecimento facial introdutório com o objetivo de entender a teoria por trás desta tecnologia.

1. Introdução

O tópico da inteligência artificial tem sido muito discutido nos últimos tempos, mas neste ano de 2023 a temática teve um crescimento de atenção exponencial com a criação do famoso “chatGPT”, além dele surgiram vários outros recursos que utilizam a inteligência artificial para facilitar o trabalho humano, portanto com a divulgação massiva da utilização dessas plataformas, essas tecnologias fizeram com que se tornassem mais intensos os debates sobre a temática.

Os softwares de Reconhecimento Facial buscam identificar pessoas por meio de imagens ou vídeos. Há alguns anos os celulares e bancos com aplicativos mobile tem essa tecnologia para desbloqueio de tela ou transações. A realização do reconhecimento facial através de um software pode parecer muito complexa, mas partindo do princípio que todas as pessoas devem ter uma boca, dois olhos e um nariz. se torna mais fácil entender como o reconhecimento é realizado. Então o seguinte relatório tem o objetivo de explicitar o desenvolvimento de um sistema de reconhecimento facial partindo do pressuposto citado.

2. Metodologia

Levando em consideração o que foi dito que o ser humano tem olhos, boca e nariz foi iniciado o treinamento com Machine Learning a fim de realizar o reconhecimento facial. Dessa forma, o objetivo deste trabalho é não apenas a conclusão da disciplina de Tópicos de Informática I, mas o professor sugeriu como um passo inicial para aplicação no controle de entrada do Laboratório de Automação e Robótica(LAR) na Escola de Ciências e Tecnologia(ECT) da Universidade Federal do Rio Grande do Norte(UFRN). Para a implementação do algoritmo de reconhecimento facial foi escolhida a linguagem de programação Python, porque existem diversas bibliotecas de Inteligência Artificial com Aprendizado de Máquina. Os códigos e resultados desses treinamentos podem ser encontrados no link do anexo deste relatório.

O trabalho foi realizado como forma de estudo, então inicialmente foi estudada a etapa de detecção facial, como o algoritmo pode reconhecer que é um rosto e depois disso o próprio reconhecimento facial. Então foi utilizada a detecção de faces utilizando classificadores em cascata baseados em recursos Haar é um método eficaz de detecção de objetos proposto no artigo Rapid Object Detection using a Boosted Cascade of Simple Features (Detecção rápida de objetos usando

uma cascata de características simples), (VIOLA e JONES, 2001). Essa abordagem é baseada em aprendizado de máquina em que a função cascata é treinada a partir de muitas imagens positivas(faces) e negativas(imagens sem rostos). Estaremos usando um Haar Cascade Model treinado para detectar rostos para obter as coordenadas da caixa delimitadora de rostos em uma imagem. Assim sendo a realização do algoritmo de detecção facial, foi utilizada uma foto do arquivo do projeto Um Robô por Aluno (Figura 1).



Figura 1: Foto com membros do projeto Um Robô por Aluno.

A primeira etapa para realização da detecção facial foi fazer as importações de bibliotecas e arquivos, e em seguida começar o treinamento em si. Inicialmente, a Figura 1 foi transformada numa imagem preto e branco, e logo em seguida foi criado o modelo de detecção a partir do arquivo `haarcascade.xml`, esse arquivo tem várias partes de um rosto humano, assim como imagens positivas e negativas, como foi citado anteriormente. Na figura 2 é possível observar a etapa em que fizemos a primeira parte do treinamento. E, na Figura 3 conseguimos fazer o treinamento em si, nesse caso, cada vez em que ‘rodamos’ essa linha de código, conseguimos detectar melhor as faces em questão. Apesar disso, se fizer muitas tentativas acontece um *overfitting*, quando o algoritmo sobrepõe os dados. Um atributo importante de se observar é o *scale Factor*, a primeira tentativa o valor dele foi 1.3 e o resultado não foi tão eficiente, já na segunda tentamos com o valor 1.1 e isso fez toda a diferença, na próxima sessão serão mostrados os resultados desta etapa.

```
detector_face = cv2.CascadeClassifier('/content/haarcascade_frontalface_default.xml')
```

Figura 2: Código detecção Facial

```
detecoes = detector_face.detectMultiScale(imagem_cinza, scaleFactor=1.1, minSize=(30,30))
```

Figura 3: Código treinamento

Na segunda etapa em que foi feito o reconhecimento facial, foi utilizada uma base de dados de da Universidade de Yale. Ela é composta por 165 imagens em escala de cinza de 15 indivíduos. São 11 imagens por indivíduo com variações de iluminação, expressão facial e usando ou não óculos. A Figura 2 é uma amostra da base de Yale. Os criadores da base disponibilizaram alguns tratamentos das imagens, por exemplo, existe um tratamento em que todas as imagens tiveram as faces centralizadas em imagens de 195x231 pixels, essa foi a variação usada neste experimento.



Figura 4: Amostra de imagem da base de Yale. Fonte: Universidade de Yale, 1997.

O algoritmo utilizado para a realização dessa etapa foi o LBPH (Local Binary Patterns Histograms) (STEKAS; HEUVEL, 2016). Esse algoritmo é muito utilizado quando se tem o objetivo de desenvolver projetos nessa temática de reconhecimento facial. Pressupondo que existe uma matriz de pixels dentro de uma imagem, o algoritmo pega o valor do pixel central para fazer uma comparação com os 8 pixels ao redor, trazendo valores maiores transformados em “1” e valores menores em “0”. A partir disso, a matriz de pixels é transformada em vários números binários, depois é convertida para números decimais, feito isto é formado um histograma. Por fim, é realizada uma combinação do histograma de cada bloco da imagem gerando um histograma final, que será utilizado para representar essa imagem (Figura 5).

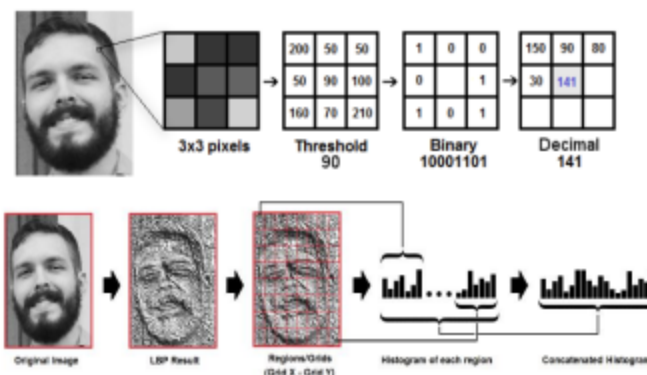


Figura 5: Exemplo de operação do LBPH. Fonte:

Assim sendo, o LBPH consegue reconhecer faces sob variadas condições de iluminação, tendo em vista que ao mudar as condições de iluminação os valores dos pixels são alterados e geram o mesmo resultado quando se comparam os pixels (SumaS, L.; Raga, S., 2018).

3. Resultados

a. Detecção Facial

Na Figura 6 é possível notar que o primeiro resultado da detecção facial não foi muito efetivo, pois o rosto da primeira pessoa ao lado esquerdo não foi detectado de forma completa e a pessoa de óculos escuro também não foi detectada como um rosto humano. Essa imagem foi gerada a partir do treinamento com o `scaleFactor` possuindo o valor 1.3. Levando em consideração que não foi efetivo, foram feitas algumas análises, mudanças e novas tentativas para que o algoritmo conseguisse aprender melhor.



Figura 6: Primeira tentativa de detecção facial.

Dessa forma, a mudança realizada que teve o melhor desempenho foi alteração do `scaleFactor` para o valor 1.1, isso trouxe o reconhecimento da pessoa com óculos escuro e um melhor enquadramento da pessoa do canto esquerdo - como pode ser observado na Figura 7. Neste caso, pode-se perceber que a imagem teve uma espécie de overfitting, porque a cada tentativa a marcação das faces vinham com sobreposições e, até mesmo reconhecendo objetos como rostos. Tendo isso em vista, o algoritmo pode ser aprimorado com mais detalhes para que não aconteça coisas desse tipo.

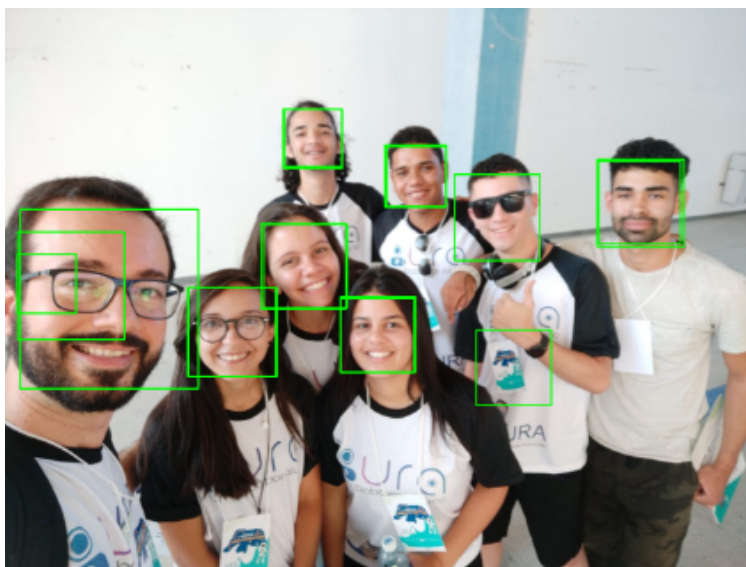


Figura 7: Resultado mais efetivo do treinamento de detecção.

b. Reconhecimento Facial

Por conseguinte, na segunda etapa de realização do trabalho que refere-se ao reconhecimento facial por imagem os resultados também foram bastante satisfatórios. Isso se dá tanto porque o algoritmo é mais eficaz e mais robusto que o haarcascade, como também porque a base de dados é mais completa. Como foi explicado anteriormente o algoritmo percorre os pixels das imagens fazendo transformações para conseguir realizar o reconhecimento, então foi solicitado ao algoritmo que trouxesse a imagem com o sombreamento na parte direita da foto da classe 1, então o algoritmo trouxe a Figura 8.



Figura 8: Resultado do algoritmo LBPH

Fazendo as devidas comparações e transformações o algoritmo conseguiu ter um resultado inicial de 100% e isso nos causou estranheza, então deletamos parte da base de dados para melhorar o cálculo e houve uma mudança para 66% de acerto no cálculo da acurácia do método LBPH.

4. Conclusão

Então como era esperado os dois métodos foram parcialmente efetivos, mas com resultados satisfatórios. O segundo método pode ser utilizado para reconhecimento em tempo real também, porém foi feito dessa forma, apenas com imagens, porque este trabalho teve o objetivo demonstrativo explicativo para a avaliação final da disciplina de Tópicos de Informática 1, ministrada pelo professor Orivaldo Santana na Universidade Federal do Rio Grande do Norte.

Sendo assim, o mais interessante da temática deste trabalho foi entender a teoria por trás do reconhecimento facial, porque atualmente com tantos avanços tecnológicos o uso dessa ferramenta tem se tornado cada vez mais comum, desde segurança de aplicativos até segurança de prédios e casas. E, no geral, muito se ouve sobre a Inteligência Artificial, mas existem dentro dessa temática muitas ramificações para se conhecer.

5. Referências bibliográficas

[1] P. Viola and M. Jones, "**Rapid object detection using a boosted cascade of simple features**," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[2] N. Stekas and D. Van Den Heuvel, "**Face Recognition Using Local Binary Patterns Histograms (LBPH) on an FPGA-Based System on Chip (SoC)**," 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Chicago, IL, USA, 2016, pp. 300-304, doi: 10.1109/IPDPSW.2016.67.

[3] SumaS, L., & Raga, S. (2018). "**Real Time Face Recognition of Human Faces by using LBPH and Viola Jones Algorithm**". International Journal of Scientific Research in Computer Science and Engineering.

6. Anexos

Lago, Maria. (2023). Reconhecimento facial. GitHub. Disponível em https://github.com/marialago/topicosInfo1/blob/main/Reconhecimento_facial.ipynb