https://eventos.utfpr.edu.br//sicite/sicite2020

Visão computacional para contagem e classificação de gênero e idade em vídeos

Computer vision for counting and classifying gender and age in videos

Rafael Hora Ramos
rafram@alunos.utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

Heitor Silvério Lopes
hslopes@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil

RESUMO

O projeto tem como objetivo a reidentificação de pessoas em vídeo, ou seja, dado um vídeo, verificar se possui pessoas no mesmo e, caso possua, fazer a contagem e a predição do gênero e da idade dessas pessoas, uma vez que ambas são características comuns na identificação de pessoas desconhecidas ou desaparecidas. Para tal são usados redes neurais e métodos de rastreamento e reconhecimento de faces. Com a finalização do projeto foi possível chegar a um acerto significativo na contagem e na predição, além de ter sido montado um *dataset* de vídeos com anotações para fazer testes, medir e comparar resultados.

PALAVRAS-CHAVE: Redes neurais. Aprendizado do computador. Processamento de imagens.

ABSTRACT

The project aim to re-identify people on video, that is, given a video, check if it has people in it and, if it does, count and predict the gender and age of these people, since both are common characteristics to identify unknown or missing persons. For this purpose, neural networks and methods of tracking and face recognition are used. With the completion of the project, it was possible to reach a significant success in counting and prediction, in addition to having assembled a video dataset with annotations for testing, measuring and comparing results.

KEYWORDS: Neural networks. Machine learning. Image processing.

Recebido: 19 ago. 2020.
Aprovado: 01 out. 2020.

INTRODUÇÃO

Na atualidade é inquestionável o grande número de dados gerados pelos mais diversos meios tecnológicos, especialmente imagens e vídeos. Concomitantemente ao grande número de dados gerados, houve também o aumento da necessidade de automatizar a análise desses dados seja ela para reconhecer padrões, para classificação ou para qualquer outro fim útil por meio da Inteligência Computacional.

Quando se trata de imagens e vídeos, a Visão Computacional entra em cena e seus problemas têm como objetivo usar os dados de imagens para inferir algo sobre o mundo (PRINCE, 2012). O aprendizado de máquina, que é uma das vertentes da Inteligência Artificial, segue a mesma linha de pensamento do aprendizado humano e aprende por exemplos.

Vários são os problemas abordados pela Visão Computacional entre eles problemas de descrição humana e biometrias, como detecção facial e reconhecimento facial, ambos utilizados nesse projeto.

MATERIAL E MÉTODOS

Para o projeto de estudo foram utilizadas Redes Neurais (NN) com o Keras (HOLLET, 2015) e Tensorflow (MARTIN ABADI, 2015), além de bibliotecas Python como Numpy (VAN DER WALT; COLBERT; VAROQUAUX, 2011), OpenCV (BRADSKI, 2000) e Scikit-learn (PEDREGOSA, 2011). Para iniciar o trabalho, foi pesquisado por modelos e *datasets* para poder estudar e compreender o problema, então foi encontrado o YoloKerasFaceDetection (ABARS, 2019), que trata do problema e tem disponível experimentos com *datasets* tais como o Adience Dataset (HASSNER, 2015) e o IMDB-Wiki (ROTHE; TIMOFTE; GOOL, 2018).

Para todo o processo foi necessária uma infraestrutura computacional robusta e para isso o Laboratório de Bioinformática e Inteligência Computacional (LABIC) dispõe de um cluster com a finalidade de atender tal necessidade.

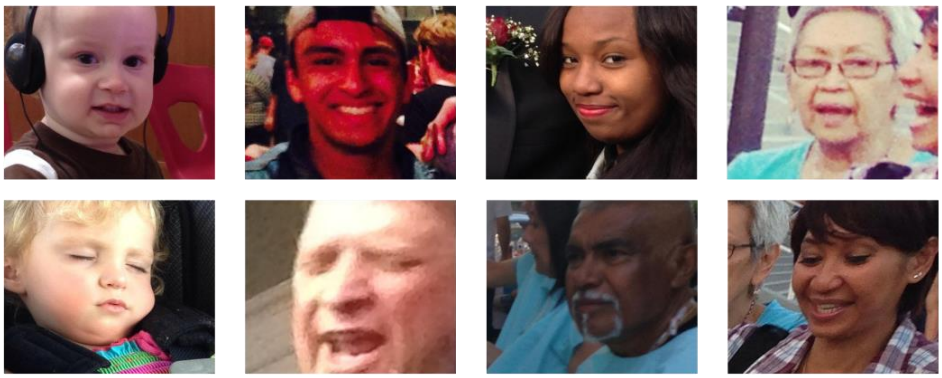
O *dataset* nada mais é que um conjunto de dados no qual são treinados e/ou testados os modelos de aprendizado de máquina. Neste projeto foram testados dois *datasets* para o treinamento e validação dos modelos de detecção de gênero e idade.

Os *datasets* utilizados foram o Adience Dataset e o IMDB-Wiki. O Adience possui 26,580 imagens de faces de pessoas e foi criado a partir de álbuns do Flickr com a finalidade de auxiliar estudos de reconhecimento de gênero e idade. Este *dataset* possui um leve desbalanceamento, ou seja, há levemente uma maior concentração de imagens em determinado grupo que em outros.

Já o IMDB-Wiki possui mais de meio milhão de imagens de faces e foi criado com rostos das 100.000 celebridades mais famosas do site IMDb juntamente com rostos das páginas de pessoas do site Wikipédia, também com a finalidade de auxiliar em estudos de reconhecimento de gênero e idade. Por sua vez, o IMDB-Wiki também não dispõe de um balanceamento muito bom, inclusive é consideravelmente mais desbalanceado que o Adience.

Vale ressaltar que, embora tenham sido feitos testes em ambos os *datasets*, foi escolhido como principal o Adience Dataset, o qual teve uma certa ênfase durante este projeto. Além disso, foi notado que o *dataset* possui imagens de pessoas pintadas, com chapéus, óculos e etc, o que são considerados ruídos e dificultam o treinamento, entretanto aproximam as imagens da realidade, onde muitas vezes pessoas estão dispostas dessa forma. Uma amostra das imagens do Adience Dataset pode ser vista na Figura 1.

Figura 1 – Amostra de imagens do Adience Dataset



Fonte: Autoria própria (2020).

Após a definição das classes, normalização das imagens e outros ajustes, foi iniciado o treinamento das redes necessárias para o projeto. Tanto para a classificação do gênero quanto da idade são utilizadas imagens de faces de pessoas e assim, a partir da características gerais como distância dos olhos, formato da face, formato da boca e expressões faciais, as pessoas são classificadas.

Para o treinamento da classificação do gênero foram comparados alguns modelos de detecção de imagens treinados em ambos os *datasets* e dois algoritmos otimizadores, que são usados no lugar do clássico gradiente descendente estocástico.

Três experimentos foram feitos para decidir qual a melhor arquitetura de rede neural, o melhor otimizador e o melhor *dataset* para o problema em questão. No primeiro experimento, para decidir a arquitetura, foram realizados treinamentos mantendo-se o otimizador e o *dataset* e substituindo apenas a arquitetura de rede neural, como pode-se observar na Tabela 1. No segundo experimento, representado na Tabela 2, para decidir o melhor otimizador, foi selecionada a melhor arquitetura de rede neural e mantido o *dataset*, substituindo apenas o otimizador. Já no terceiro experimento, para definir o melhor *dataset*, foram mantidos a melhor arquitetura e o melhor otimizador substituindo o apenas *dataset*, como mostrado na Tabela 3.

Em geral o modelo EfficientNet (TAN; LE, 2019), que é uma arquitetura de rede neural convolucional, se mostrou melhor, inclusive com um F1-Score, que é uma boa métrica para avaliar a qualidade do modelo e usa como base a quantidade de erros e acertos nas predições, de aproximadamente 0.91 na verificação do *dataset* IMDB, que possui mais de meio milhão de imagens para classificação, portanto, em um primeiro momento, foi escolhido o modelo gerado pelo treino do IMDB para classificação do gênero. Além disso, como pode-se perceber no experimento

2, nas comparações, o otimizador Adam se mostrou mais eficiente que o otimizador RMSprop nos treinos, inclusive com tempo de processamento bem inferior, o que também levou à escolha do mesmo.

Para o treinamento da classificação de idade foi usada uma estratégia diferente na escolha do *dataset*, embora tenha sido mantida a mesma escolha de arquitetura e otimizador. Ao verificar o balanceamento dos *datasets* para idade, foi observado que o *dataset* do Adience tem um balanceamento muito melhor que o IMDB para isso, ou seja, tem uma disposição melhor de pessoas em cada classe de idade.

Tabela 1 – Resultados do primeiro experimento.

Modelo	Otimizador	Duração	F1_score	Dataset
EfficientNet	Adam	3h58m	0. 9729	Adience
InceptionV3	Adam	1h40m	0.9726	Adience
VGG16	Adam	0h25m	0.9379	Adience
SqueezeNet	Adam	0h25m	0.9437	Adience
SqueezeNet2	Adam	0h21m	0.9050	Adience

Fonte: Autoria própria (2020).

Tabela 2 – Resultados do segundo experimento.

Modelo	Otimizador	Duração	F1_score	Dataset
EfficientNet	Adam	3h58m	0. 9729	Adience
EfficientNet	RMSprop	6h21m	0.9638	Adience

Fonte: Autoria própria (2020).

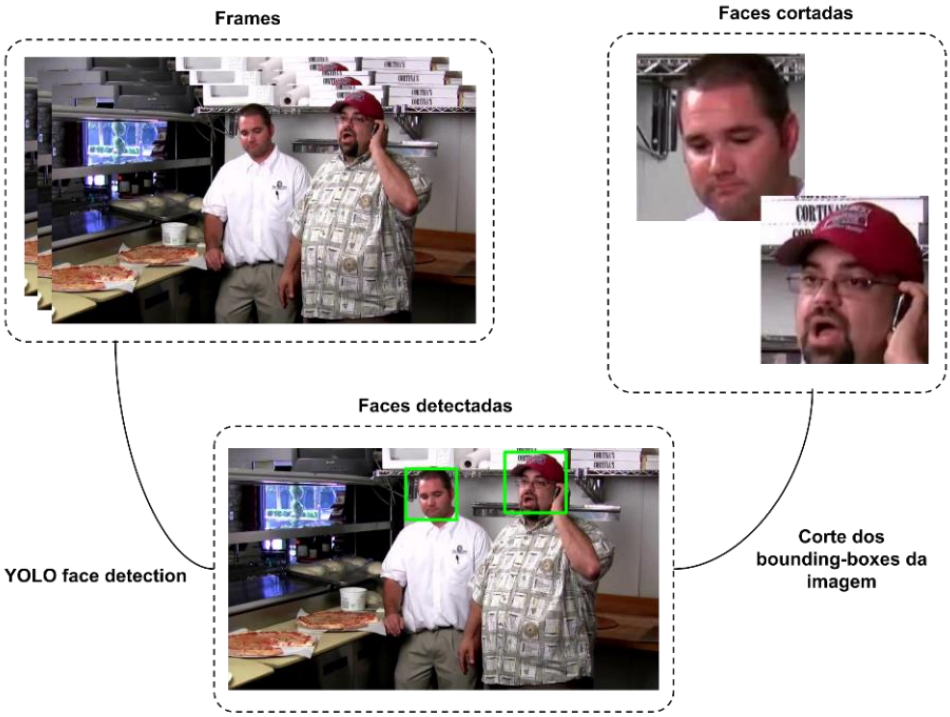
Tabela 3 – Resultados do terceiro experimento.

Modelo	Otimizador	Duração	F1_score	Dataset
EfficientNet	Adam	3h58m	0. 9729	Adience
EfficientNet	Adam	15h25m	0.9086	IMDB

Fonte: Autoria própria (2020).

Após o treinamento dos modelos de gênero e idade, a atenção foi voltada para a detecção da face e o corte da mesma para então ser passada pela rede neural e fazer as predições. Para a detecção facial foram usados três métodos: Utilizando o cvlib (PONNUSAMY, 2018), o OpenFace (AMOS; LUDWICZUK; SATYANARAYANAN, 2016) e o YOLOv2 (REDMON; FARHADI, 2016). Os três tiveram bons resultados, entretanto ainda assim houve erros de detectar faces onde não existia de fato uma face, o que foi possível melhorar ajustando pelo YOLO, tal processo de detecção e corte pode ser visualizado na Figura 2.

Figura 2 – Processo de detecção e corte das faces



Fonte: Autoria própria (2020).

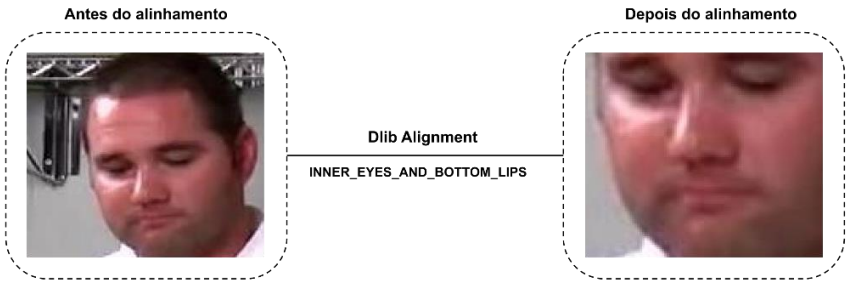
Como as faces são detectadas e cortadas *frame* a *frame*, surgiu um novo desafio para a contagem e reidentificação das pessoas no vídeo: Como saber se uma pessoa já foi detectada em *frames* anteriores ou se é uma nova pessoa no *frame* atual? A primeira solução foi de certa forma registrar e "rastrear" os *bounding-boxes* encontrados de forma que, a cada *frame*, quando uma pessoa é encontrada, esse *bounding-box* é comparado com os *bounding-boxes* já salvos, caso esse esteja a uma distância menor que um dado limite (*threshold*), significa que a pessoa já foi detectada e atualiza o *bounding-box* atual para comparar com *frames* futuros, porém caso esteja a uma distância maior que o *threshold*, significa que não foi detectada ainda e salva o *bounding-box* como nova pessoa, como é feito na Eq. (1).

$$\sum_{i=0}^{len(detected)} |actual\_face - detected\_faces[i]| < threshold, \quad (1)$$

onde *actual_face* é o *bounding-box* da face atual, *detected_faces* é a lista com os *bounding-boxes* das faces já detectadas e *threshold* é o limite de variação do *bounding-box* nos dados *frames*.

Essa abordagem gerou bons resultados, entretanto ainda aconteciam erros, principalmente em casos de transições nos vídeos. Para melhorar os resultados a solução foi usar reconhecimento facial para esse problema. Então novamente, as faces são salvas, mas além disso elas são alinhadas e vetorizadas utilizando a extração de características com a ferramenta do OpenFace. Então, a partir de uma função que calcula distância, é medida a similaridade de uma face com a outra. O alinhamento é utilizado para remover o máximo possível de tudo que não faz parte da face propriamente dita, assim, aumentando a eficiência do cálculo da distância entre as faces. Este resultado pode ser visto na Figura 3.

Figura 3 – Resultado do alinhamento com Dlib



Fonte: Autoria própria (2020).

Outro ponto importante é que foram utilizados dois modos diferentes para calcular a similaridade das faces. O primeiro foi a distância Euclidiana ao quadrado, ilustrada na Eq. (2), e a segunda foi usando a similaridade ilustrada na Eq. (3), sendo que essa última teve um resultado significativamente melhor para o problema abordado.

$$dist = \|(vet1 - vet2)\| = \sum(vet1 - vet2)^2 \quad (2)$$

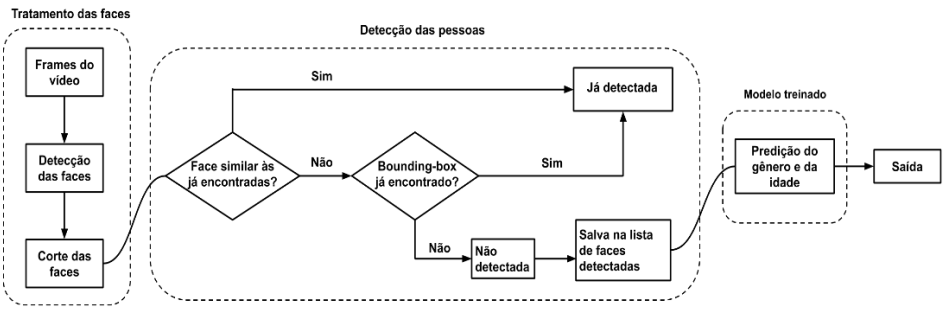
$$cos(\theta) = \frac{vet1 \cdot vet2}{\sqrt{vet1 \cdot vet1}\sqrt{vet2 \cdot vet2}} = \frac{vet1 \cdot vet2}{\|vet1\|\|vet2\|}, \quad (3)$$

onde vet1 é a imagem de uma face e vet2 é a imagem de outra face, ambas já alinhadas e vetorizadas.

Usar o reconhecimento facial gerou melhores resultados que a abordagem dos *bounding-boxes*, porém ainda não satisfatórios. Então foi decidido juntar ambas as abordagens de forma que, após o cálculo da similaridade, se a pessoa tiver uma similaridade menor que o *threshold*, ou seja, se tiver uma probabilidade de ser uma nova pessoa, ainda assim é verificado o *bounding-box*, uma vez que um ângulo diferente pode fazer com que a similaridade varie bastante. Com isso, os resultados melhoraram significativamente.

Após todo o tratamento e identificação das diferentes pessoas no vídeo, as duas abordagens são utilizadas novamente para cada face encontrada, mas dessa vez com a finalidade de obter todas as predições de gênero e idade para uma determinada pessoa e pegar a moda dessas predições, aumentando a probabilidade de acerto na predição, o processo final do projeto pode ser visto na Figura 4.

Figura 4 – Processo final do projeto



Fonte: Autoria própria (2020).

Com o avanço dos testes foi feito um *dataset* de vídeos com anotações de gênero e grupo de idade para cada pessoa presente com a finalidade de fazer um

benchmark do algoritmo. Tal *dataset* foi criado com a seleção de vídeos de três *datasets* já existentes, o MSVD (CHEN; DOLAN, 2011), o MSR-vtt (XU; Mei; Yao; Rui, 2016) e o Charades (SIGURDSSON, 2016). Então em um dado momento foi feita a substituição do modelo treinado com o *dataset* IMDB-Wiki pelo dataset Adience e foi verificado que, apesar do bom resultado na validação do IMDB-Wiki, em situações de vídeos reais deste *dataset* de *benchmark*, o Adience também acerta mais na predição de gênero, portanto esse modelo substituiu o do IMDB-Wiki, assim, deixando toda a predição com modelos treinados no Adience Dataset.

RESULTADOS E DISCUSSÃO

Com a finalização do projeto, além de um algoritmo de detecção, contagem de pessoas e predição de gênero e idade em vídeos, foi gerado um *dataset* com 70 vídeos variados e anotados com o gênero e grupo de idade das pessoas a cada vídeo.

O acerto do algoritmo foi considerado bom, uma vez que quando testado no *dataset* criado com a finalidade de *benchmark* foi obtido um acerto de 70% na contagem, acertando o número exato de pessoas em 49 dos 70 vídeos, um acerto de 77.55% na predição de gênero e grupo de idade sem excesso, ou seja, desconsiderando possíveis pessoas contabilizadas a mais ou a menos e também um acerto de 67.86% considerando excessos, ou seja, considerando os erros das pessoas contabilizadas a mais ou a menos, o que é um resultado interessante dada a variedade dos vídeos, a baixa qualidade de alguns dos vídeos, buscando mais proximidade com a realidade, e dadas as posições difíceis que algumas pessoas estão dispostas nos mesmos, que são possíveis causas para não ter sido obtido um acerto ainda maior.

CONCLUSÃO

Levando em consideração todos os estudos realizados durante o projeto, foi possível observar a grande relevância da Visão Computacional para tarefas tão importantes como reconhecimento de padrões e classificações de imagens. Durante esse período de pesquisa também foi notada a complexidade que essa tarefa pode ter e o quão necessária a identificação de padrões pode ser.

Analisando isso, pode-se afirmar que o projeto foi um sucesso ao conseguir detectar, contar e predizer gênero e grupo de idade de pessoas em vídeos com um acerto significativo, o que pode auxiliar em futuros projetos de pesquisa ou de segurança, como identificação de pessoas desaparecidas, e com certeza irá agregar à área da Visão Computacional.

AGRADECIMENTOS

Agradeço ao Conselho Nacional de Desenvolvimento Científico e Tecnológico pela bolsa de Iniciação Científica concedida a mim. Também agradeço o meu orientador Heitor Silvério Lopes e aos colegas de laboratório que ajudaram muito no projeto. Além disso agradeço a Universidade Tecnológica Federal do Paraná pelo ensino superior de qualidade.

REFERÊNCIAS

ABARS. **Yolo Keras Face Detection**. [S.l.: s.n.], 24 fev. 2019. Disponível em: https://github.com/abars/YoloKerasFaceDetection. Acesso em: 20 out. 2019.

AMOS, B.; LUDWICZUK, B.; SATYANARAYANAN, M. **OpenFace**: A general-purpose face recognition library with mobile applications. [S.l.], 2016.

BRADSKI, G. **The OpenCV Library**. Dr. Dobb's Journal of Software Tools, 120; 122-125, 2000.

CHEN, D.; DOLAN, W. **Collecting Highly Parallel Data for Paraphrase Evaluation**. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA: Association for Computational Linguistics, jun. 2011. P. 190–200.

HASSNER, T. et al. **Effective Face Frontalization in Unconstrained Images**. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). [S.l.: s.n.], jun. 2015.

HOLLET, F. et al. **Keras**. [S.l.: s.n.], 2015. Disponível em: https://keras.io. Acesso em: 21 out. 2019.

J. Xu, T. Mei, T. Yao and Y. Rui, "**MSR-VTT**: A Large Video Description Dataset for Bridging Video and Language," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 5288-5296.

MARTIN ABADI et al. **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. [S.l.: s.n.], arXiv preprint arXiv:1603.04467, 2015.

PEDREGOSA, F. et al. **Scikit-learn**: Machine learning in Python. Journal of machine learning research, v. 12, Oct, p. 2825–2830, 2011.

PONNUSAMY, A. **cvlib** - high level Computer Vision library for Python. [S.l.: s.n.], 2018. Disponível em: https://github.com/arunponnusamy/cvlib. Acesso em: 16 fev. 2020.

PRINCE, S. J. D. **Computer Vision**: Models, Learning, and Inference. 1. ed. New York, NY, USA: Cambridge University, 2012.

REDMON, J.; FARHADI, A. **YOLO9000**: Better, Faster, Stronger. arXiv preprint arXiv:1612.08242, 2016.

ROTHE, R.; TIMOFTE, R.; GOOL, L. V. **Deep expectation of real and apparent age from a single image without facial landmarks**. International Journal of Computer Vision, Springer, v. 126, n. 2-4, p. 144–157, 2018.

TAN, M.; LE, Q. V. **EfficientNet**: Rethinking Model Scaling for Convolutional Neural Networks. [S.l.: s.n.], 2019.

VAN DER WALT, S.; COLBERT, S. C.; VAROQUAUX, G. **The NumPy Array**: A Structure for Efficient Numerical Computation. Computing in Science Engineering, v. 13, n. 2, p. 22–30, 2011.