Facial recognition from images (identification)

POVa - Počítačové vidění (v angličtině)

Vojtěch Vlach	xvlach22
Martin Kneslík	xknesl02
Zuzana Hrkľová	xhrklo00

1 Assignment

Prepare a demo application demonstrating facial recognition in good lighting conditions. Evaluate accuracy on you own data and on a existing dataset. Ideal approach is:

1. Detect faces using existing detector. Good choices are OpenCV, Dlib or DCGM/MTCNN.
2. Align the face based detected facial features (map to avarage face).
3. Extract face fingerprint using a convolutional neural network. You can start with some pretrained network or train or fine-tune your own - search Model Zoo for suitable network Model-Zoo.
4. Search database of faces.

2 Team responsibilities

- Zuzana Hrkľová - face detection and alignment
- Martin Kneslík - datasets, demo application
- Vojtěch Vlach - fine-tuning and evaluation

3 Proposed solution

Face detection. We compared several detectors including `OpenCV`, `Dlib` and `MTCNN` and decided to use `facenet_pytorch` in our future implementation due to its high accuracy in detecting faces across various poses and lighting conditions. It performs face detection through `MTCNN`.

Face embedding extraction. We've chosen the facenet_pytorch library so far as it offers two pre-trained models: `VGGFace2` [1] and `CASIA-WebFace` [3]. Both models implement `InceptionResnetV1` [2] architecture varying only in training datasets.

Dataset. We will be using Large-scale CelebFaces Attributes (CelebA) Dataset. It contains 202599 aligned and cropped face images of 10177 identities. Each image is additionally annotated with class (identity), bounding box, landmarks positions (coordinates of left eye, right eye, nose, left mouth, right mouth) and 40 binary attributes (such as eyeglasses, mustache, wearing hat, etc.). Dataset is also partitioned into training, validation and testing set (1-162770 train, 162771-182637 val, 182638-202599 test).

4 Planned experiments

The models available today have been trained on large datasets making them robust. We plan to find a more specific task/dataset, where these models can be improved, e.g. facial hair/make-up invariance or historical/s-canned pictures. After we find sufficient data (subset of CelebA dataset or our own custom dataset) for a task we will fine-tune and evaluate our model against the original one.

5 Evaluation

For this project, we choose to see face identification as **classification of faces** into classes/identities so we can use typical classification metrics like **accuracy** or **F1 score** (average precision over recall for individual classes). Our primary distance metric to choose a class will be **Cosine similarity**.

Public project repository: `https://github.com/vlachvojta/POVa_face_identification`

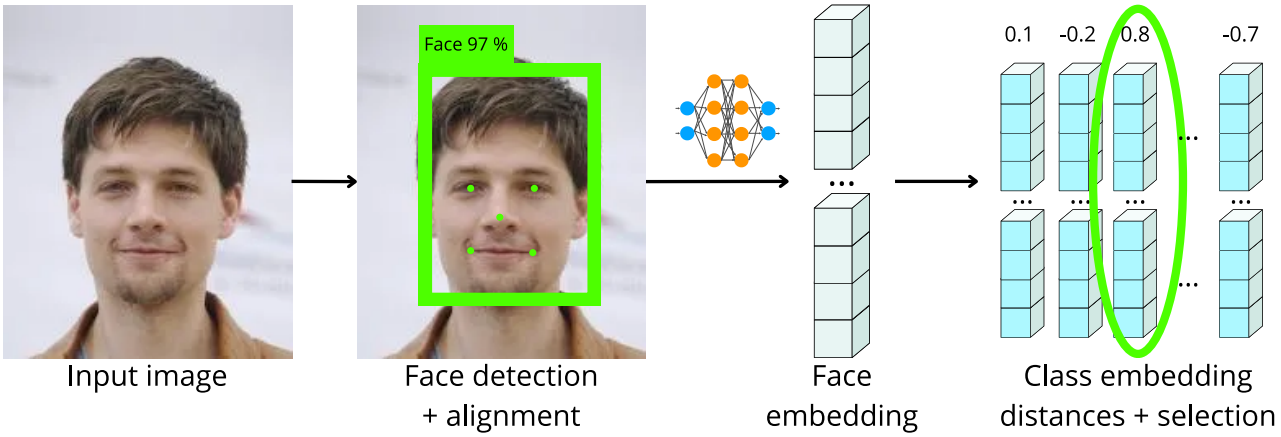

Figure 1: Face identification pipeline

References

[1] Cao, Q., Shen, L., Xie, W., Parkhi, O. M. and Zisserman, A. VGGFace2: A Dataset for Recognising Faces across Pose and Age. In: *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. 2018, p. 67–74. DOI: 10.1109/FG.2018.00020.

[2] Peng, S., Huang, H., Chen, W., Zhang, L. and Fang, W. More trainable inception-ResNet for face recognition. *Neurocomputing*. 2020, vol. 411, p. 9–19. DOI: https://doi.org/10.1016/j.neucom.2020.05.022. ISSN 0925-2312. Available at: https://www.sciencedirect.com/science/article/pii/S0925231220308572.

[3] Yi, D., Lei, Z., Liao, S. and Li, S. Z. Learning face representation from scratch. *ArXiv preprint arXiv:1411.7923*. 2014.