Presentation of coursework

in specialty “Computer science” 122

Implementation of the face recognition system using deep learning methods

Made by student
Shpir M. V.

Coursework supervisor
Yushchenko Y.O.

Face recognition relevance

Enables the identification and authentication of individuals based on their based on unique facial features.
Has numerous practical applications across various domains, including **security, customer service, and social media**.

Why it is important to improve the accuracy of existing facial recognition systems?

- Enhanced Security
- Crime Prevention
- Public Safety
- Efficient Identity Verification
- Personalization and User Experience
- Streamlined Operations
- Privacy Protection
- Mitigating Bias
- Improved Social Media Experience
- Advancing Technological Innovation

Face recognition model: FaceNet

FaceNet is a face recognition model developed by Google researchers with Siamese model architecture [1][2].

FaceNet aims to generate face embeddings where faces of the same identity are close together and faces of different identities are far apart.

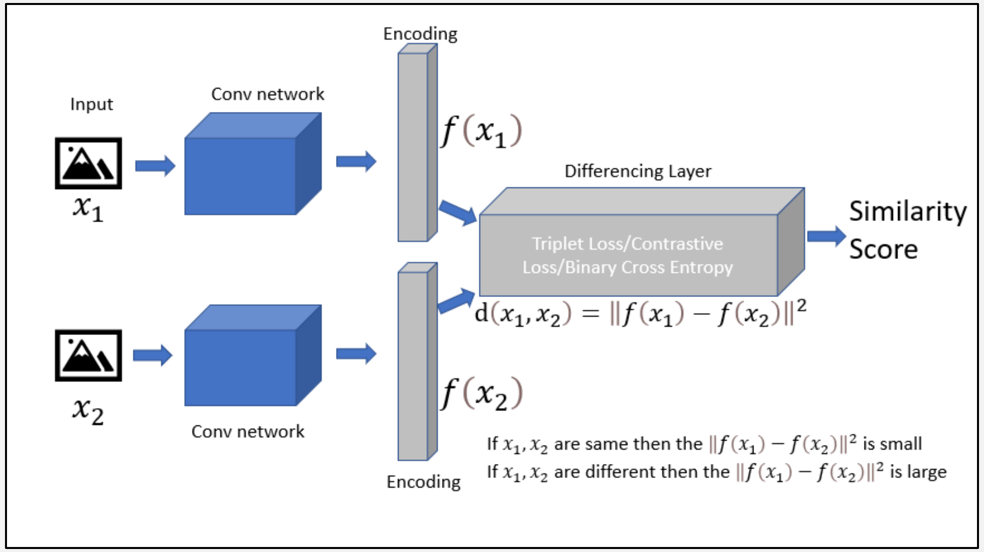

Figure 1. Siamese model architecture [5]

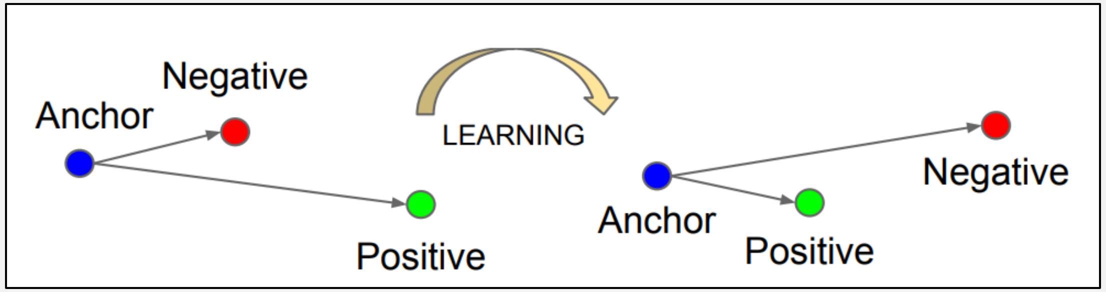

Figure 2. Learning with the Triplet Loss function [2]

FaceNet achieved state-of-the-art performance at the time of its release. On the Labeled Faces in the Wild (LFW) dataset [4], FaceNet achieved a verification accuracy of **over 99%**.

Datasets

Labeled Faces in the Wild (LFW)

Figure 3. Visualization of the Labeled faces in the Wild dataset [3]

VGGFace2

Figure 4. Visualization of the VGGFace2 dataset [6]

Method

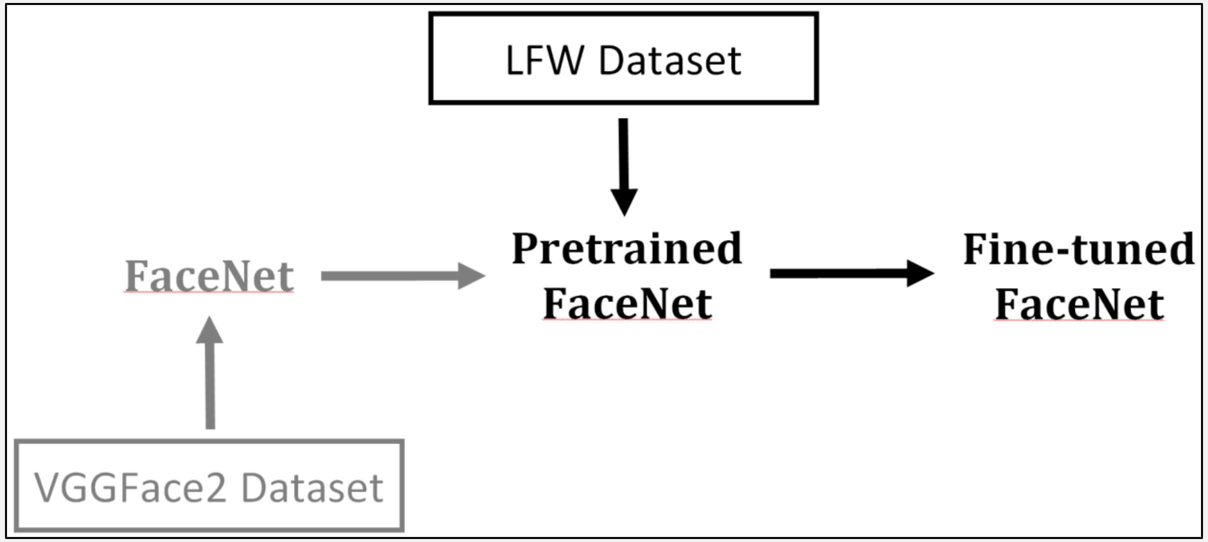

Figure 5. Pipeline of fine-tuning pretrained FaceNet model

The FaceNet model architecture and pre-trained weights on the VGGFace2 dataset developed by Tim Esler et al. [7]

	Pretrained model
Loss	0.02
Averaged positive distance	0.73
Averaged negative distance	1.26
VAL	0.89
FAR	0.05

Table 1. Calculated evaluation metrics of the pretrained model

Results

	Original model	Fine-tuned model
Loss	0.02	0.01
Averaged positive distance	0.73	**0.41**
Averaged negative distance	1.26	**1.38**
VAL	0.89	0.94
FAR	0.05	0.04

Table 2. Calculated evaluation metrics of the original and fine-tuned models

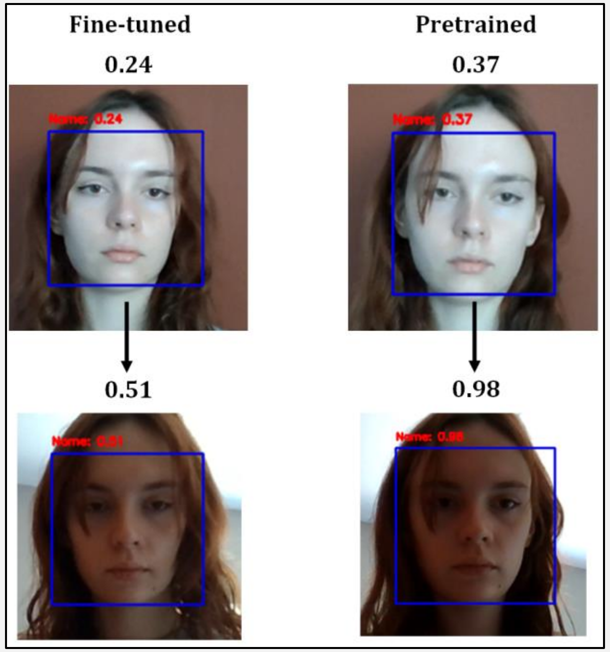

Figure 6. Distances under normal and poor lighting of pretrained and fine-tuned models

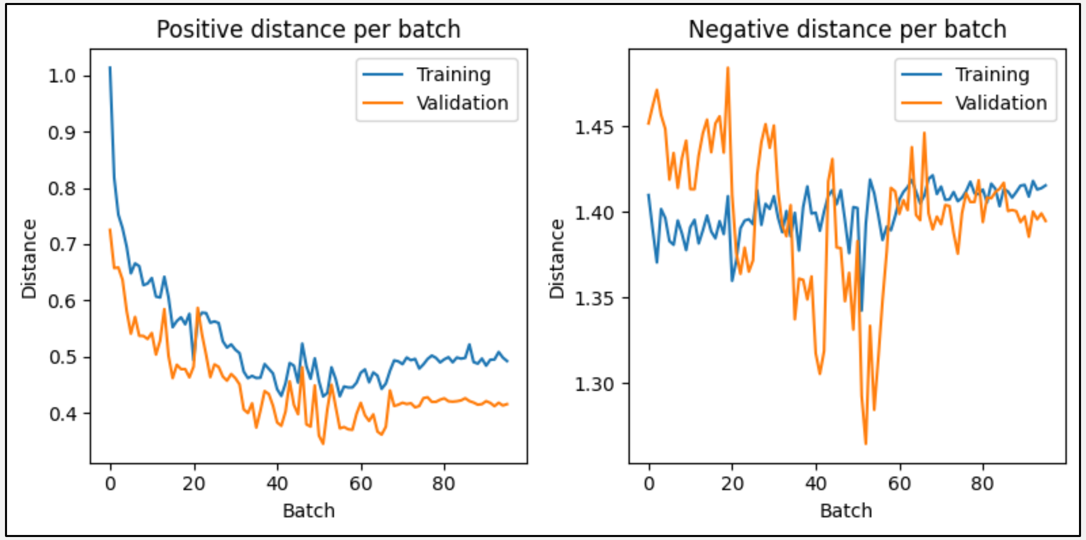

Figure 7. Averaged distances per batch

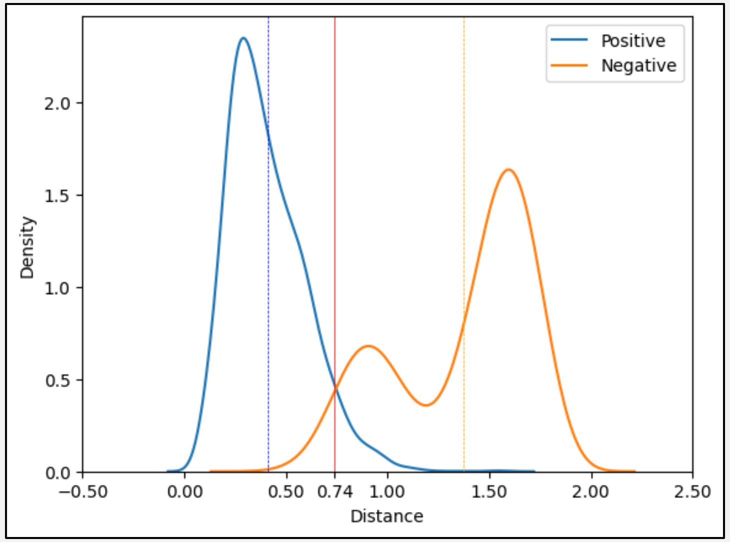

Figure 8. Distance distribution calculated between embeddings of fine-tuned model

Implementation

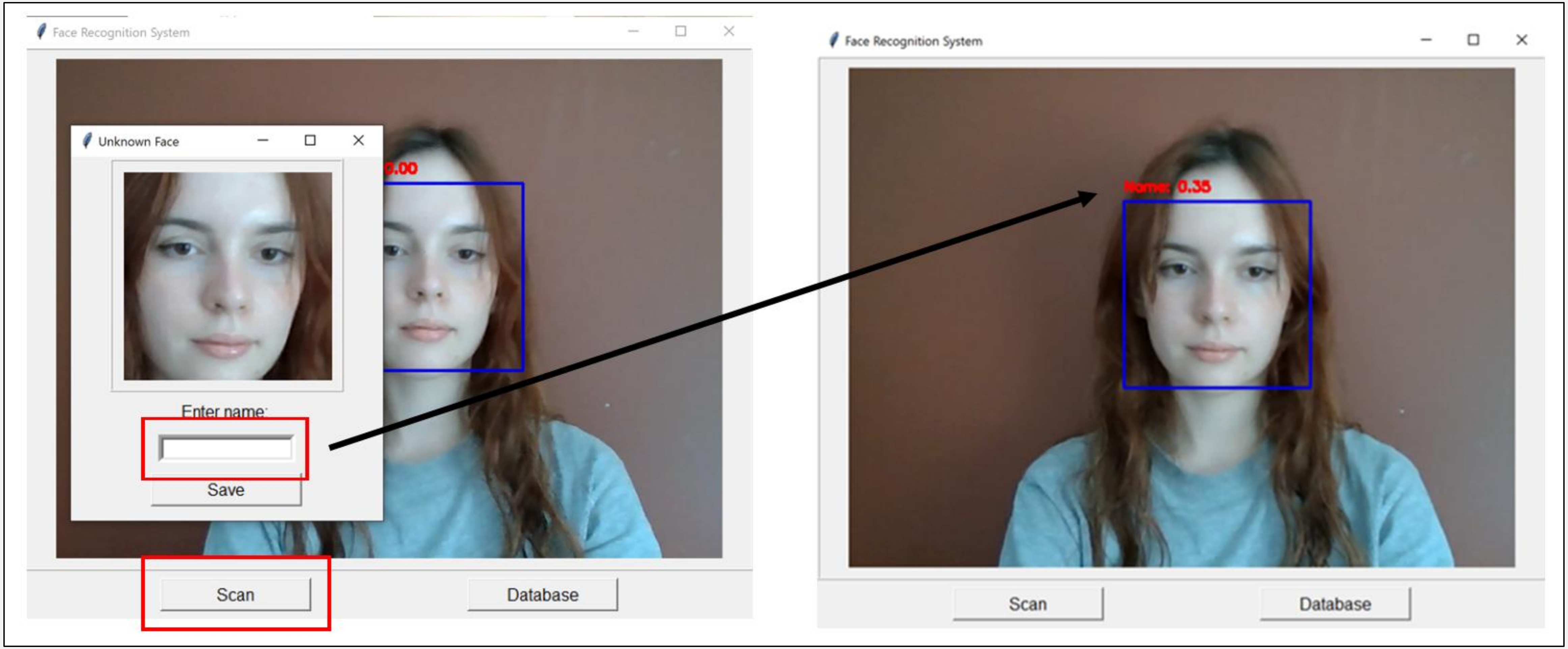

Figure 9. The window for saving detected unknown face

Implementation

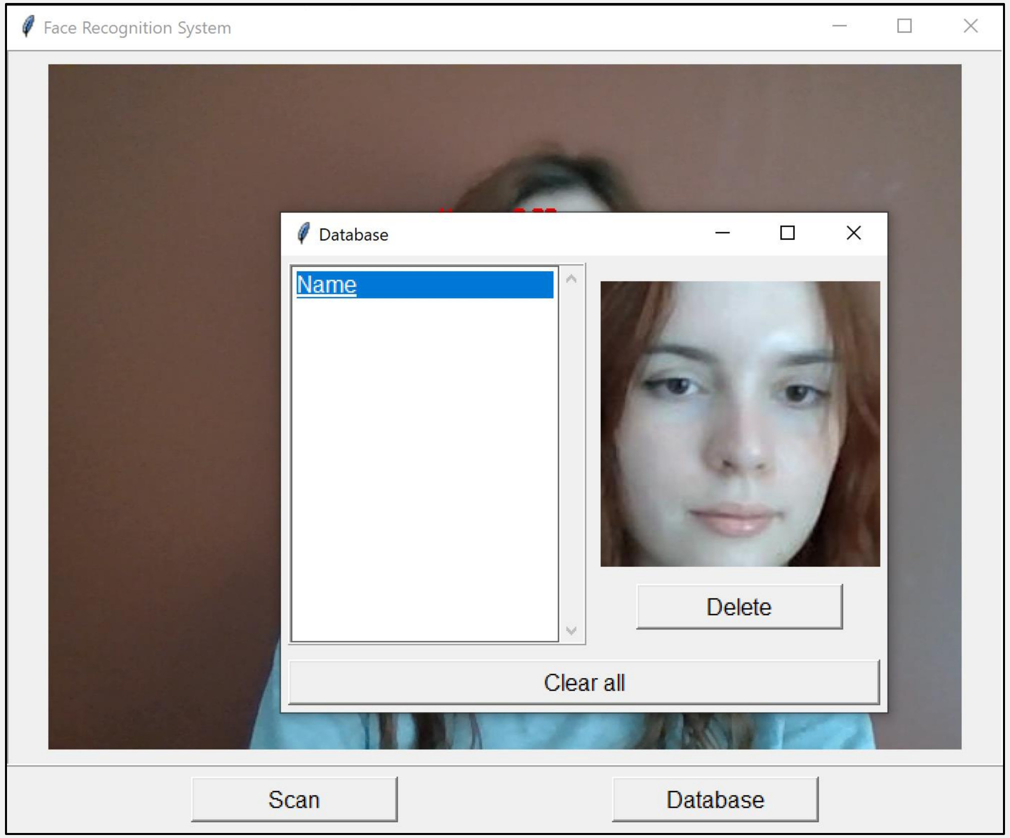

Figure 10. The user list interface

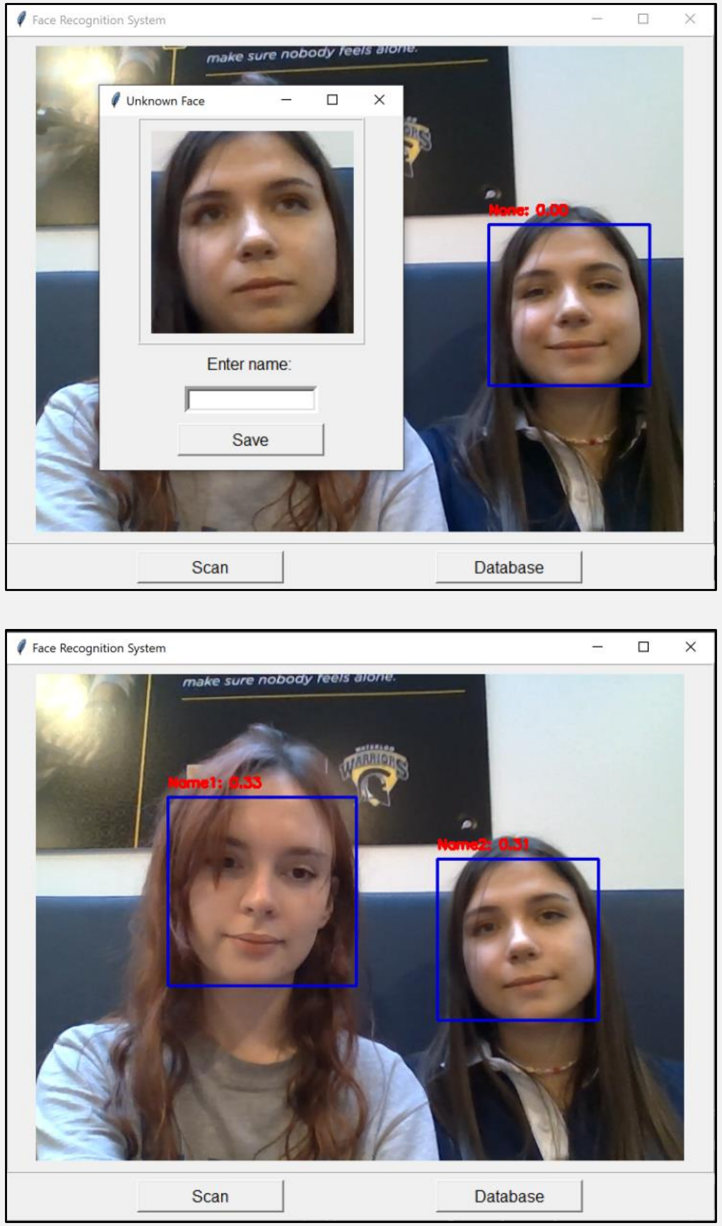

Figure 11. The system detects multiple faces

References

1. Koch, G., Zemel, R., and Salakhutdinov, R. (2015**). Siamese neural networks for one-shot image recognition.** Proceedings of the 32nd International Conference on Machine Learning, 1302-1310.
2. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). **FaceNet: A unified embedding for face recognition and clustering.** Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 815-823.
3. **LFW Dataset** (2023) Machine Learning Datasets. Available at: https://datasets.activeloop.ai/docs/ml/datasets/lfw-dataset/
4. Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008). **Labeled faces in the wild: A database for studying face recognition in unconstrained environments.** Technical Report, University of Massachusetts, Amherst.
5. Khandelwal, R. (2021) **One-shot learning with Siamese network**, Medium. The Startup. Available at: https://medium.com/swlh/one-shot-learning-with-siamese-network-1c7404c35fda
6. Cao, Q., Shen, L., Xie, W., Parkhi, O. M., & Zisserman, A. (2018). **VGGFace2: A dataset for recognising faces across pose and age**. In 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018) (pp. 67-74). IEEE.
7. Esler, T. et al. (2019) Timesler/facenet-pytorch: **Pretrained Pytorch Face Detection (MTCNN) and facial recognition (InceptionResnet) models**, GitHub. Available at: https://github.com/timesler/facenet-pytorch