FACE RECONSTRUCTION FROM DEEP FACIAL EMBEDDINGS USING A CONVOLUTIONAL NEURAL NETWORK

Hatef Otroshi Shahreza[*†], *Vedrana Krivokuća Hahn*[*], *and Sébastien Marcel*[*‡]

[*]Idiap Research Institute, Switzerland
[†]École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[‡]Université de Lausanne (UNIL), Switzerland

ABSTRACT

State-of-the-art (SOTA) face recognition systems generally use deep convolutional neural networks (CNNs) to extract deep features, called embeddings, from face images. The face embeddings are stored in the system's database and are used for recognition of the enrolled system users. Hence, these features convey important information about the user's identity, and therefore any attack using the face embeddings jeopardizes the user's security and privacy. In this paper, we propose a CNN-based structure to reconstruct face images from face embeddings and we train our network with a multi-term loss function. In our experiments, our network is trained to reconstruct face images from SOTA face recognition models (ArcFace and ElasticFace) and we evaluate our face reconstruction network on the MOBIO and LFW datasets. The source code of all the experiments presented in this paper is publicly available so our work can be fully reproduced.

Index Terms— embedding, face reconstruction, face recognition, template inversion

1. INTRODUCTION

Face recognition systems are widely used and have become a popular authentication tool in recent years. State-of-the-art (SOTA) face recognition systems are based on deep convolutional neural networks (CNNs) which extract features, called "embeddings", from face images. In the enrollment stage, these deep features are extracted and stored in the database of the face recognition system, and later, in the recognition stage, new features are extracted from the user's face and are compared with the reference embeddings which are stored in the system's database. Hence, the face embeddings contain information about a user's identity. While most attacks against face recognition systems threaten the security of these systems [1, 2, 3], a template inversion attack jeopardizes the privacy of the users as well. In a template inversion attack, the adversary gains access to the system database and tries to invert the templates (embeddings) stored within to reconstruct the underlying face images. Then, the adversary can enter the system by injecting the reconstructed face image as a query to the system. Moreover, the adversary may be able to obtain privacy-sensitive information about the users from the reconstructed face images. For instance, Fig. 1 shows sample face images from the FFHQ [4] dataset and their reconstructed versions from ArcFace [5] and ElasticFace [6] embeddings using our face reconstruction network. As this figure shows, the reconstructed face images reveal important information about the user's identity, such as race, age, and gender.

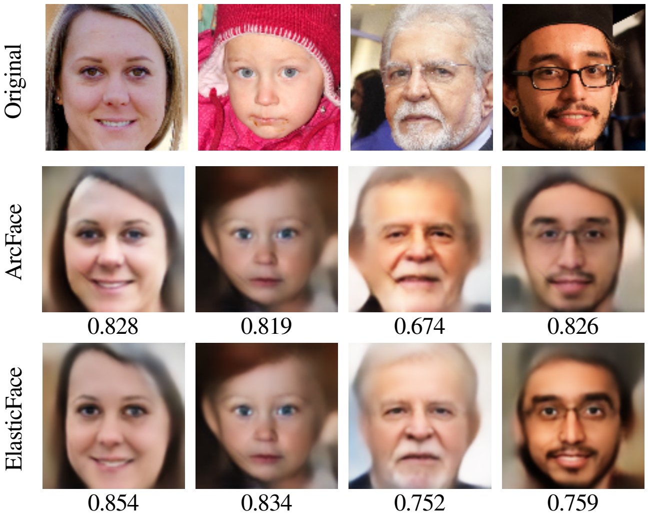


Fig. 1: Sample face images from the FFHQ dataset (first row) and their corresponding reconstructed face images from ArcFace (second row) and ElasticFace (third row) embeddings. The values indicate cosine similarity between the original and reconstructed image embeddings. The decision thresholds corresponding to FMR $= 10^{-3}$ are 0.37 and 0.41 for ArcFace and ElasticFace, respectively, on the MOBIO dataset.

Different methods have been proposed in the literature to reconstruct face images from deep templates [7, 8, 9, 10, 11, 12]. Considering the amount of knowledge about the face recognition model, these methods can be categorized into *whitebox* (where the model and its parameters are known) and *blackbox* (where there is no information on the internal functioning of the model). Zhmoginov and Sandler [7] considered whitebox face reconstruction and proposed a gradient-ascent-based approach to reconstruct face images using the face recognition model and regularization terms. They also trained a deconvolution neural network to generate face

This research is based upon work supported by the H2020 TReSPAsS-ETN Marie Skłodowska-Curie early training network (grant agreement 860813).

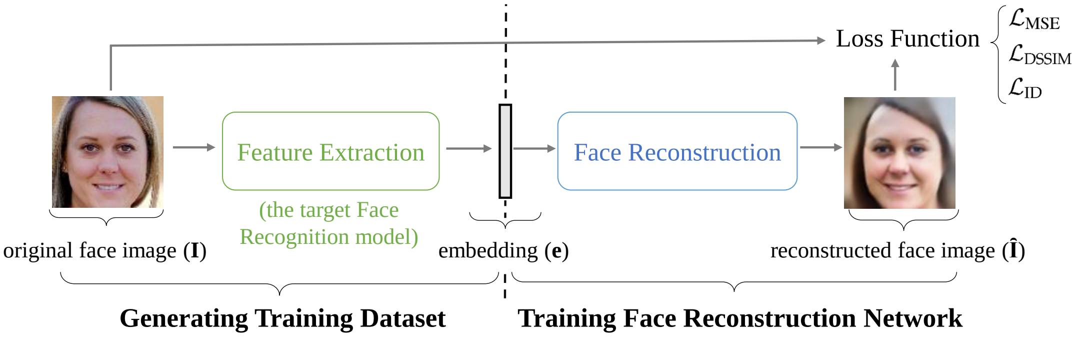

Loss Function
$\mathcal{L}_{\text{MSE}}$
$\mathcal{L}_{\text{DSSIM}}$
$\mathcal{L}_{\text{ID}}$
Feature Extraction
(the target Face Recognition model)
Face Reconstruction
original face image (I)
embedding (e)
reconstructed face image ($\hat{\mathbf{I}}$)
Generating Training Dataset
Training Face Reconstruction Network

Fig. 2: Block diagram of the proposed face reconstruction method

images. Cole *et al.* [8] used a multi-layer perceptron (MLP) to estimate landmark coordinates and a CNN to generate face textures. Then, they used differentiable warping to generate reconstructed faces. In the blackbox scenario, they trained their MLP and CNN separately, and then reconstructed the face image using the warping function. However, in the whitebox scenario, they used the warping function in the training and used an additional loss using the face recognition model to minimize the distance between embeddings extracted from the original and reconstructed face images. Mai *et al.* [9] considered the blackbox scenario and proposed two CNNs based on two new blocks, NBNet-A and NBNet-B, to reconstruct face images. Duong *et al.* [10] used bijection learning and proposed a generative adversarial network (GAN) to reconstruct face images. They proposed their method based on a whitebox scenario, though in the blackbox scenario, they used distillation of knowledge to train a student network from the face recognition model. However, they did not report details (and also did not publish source code) on the network structure and training process for their student network. Vendrow and Vendrow [11] considered the blackbox scenario and proposed a greedy random optimization over the latent space of StyleGAN [13]. Then, using a hill climbing approach, they find a latent vector which synthesizes an image that has embedding close to the target embedding.

In this paper, we consider a whitebox template inversion attack against SOTA face recognition systems. We propose a convolutional neural network to reconstruct face images from face embeddings and train our network with a multi-term loss function. We train our network for two SOTA face recognition models, ArcFace [5] and ElasticFace [6], and evaluate our trained face reconstruction networks on the MOBIO [14] and LFW [15] datasets. Our experiments show that the proposed network improves the face reconstruction performance in terms of an adversary's success attack rate.

The rest of the paper is organized as follows. In section 2, we describe our proposed face reconstruction method. Then, in section 3, we describe our experiments and discuss our results. Finally, the paper is concluded in section 4.

2. PROPOSED METHOD

In this section, we describe our proposed method for the reconstruction of face images from face embeddings. First, we explain how to generate our training data in section 2.1. Then, we describe our network structure in section 2.3, and finally we describe our loss function in section 2.2. Fig. 2 illustrates the block diagram of the proposed method.

2.1. Training Data

To train our face reconstruction network, we need a dataset of face images and their corresponding embeddings. To generate such a dataset, let us consider a dataset $\mathcal{I}=\{\mathbf{I}_i\}_{i=1}^N$ containing N face images. We can generate our training dataset $\mathcal{D}=\{(\mathbf{e}_i,\mathbf{I}_i)\}_{i=1}^N$ by extracting face embeddings from all face images in $\mathcal{I}$, where $\mathbf{e}_i=F(\mathbf{I}_i)$ indicates the face embedding extracted from image $\mathbf{I}_i$ using face recognition model $F(.)$.

2.2. Loss Function

Let $(\mathbf{e},\mathbf{I})\in\mathcal{D}$ denote a (face embedding, face image) pair in our training dataset $\mathcal{D}$, and $\hat{\mathbf{I}}$ the face image reconstructed from the face embedding $\mathbf{e}$ using our face reconstruction network. We train our network with a multi-term loss function including:

- *Mean Squared Error (MSE):* To reduce reconstruction error of the generated face, we use the Mean Squared Error (MSE) loss term using the square of ℓ_2-norm of the reconstruction error:

$$\mathcal{L}_{\text{MSE}}(\hat{\mathbf{I}},\mathbf{I})=||\hat{\mathbf{I}}-\mathbf{I}||_2^2 \quad (1)$$

- *Dissimilarity Structural Index Metric (DSSIM):* In addition to MSE of the reconstructed face, we maximize the Similarity Structural Index Metric (SSIM) [16] of the reconstructed image, to maximize the reconstruction quality. To this end, we optimize the DSSIM loss term [17] as follows:

$$\mathcal{L}_{\text{DSSIM}}(\hat{\mathbf{I}},\mathbf{I})=\frac{1-\text{SSIM}(\hat{\mathbf{I}},\mathbf{I})}{2} \quad (2)$$

- *ID loss:* In addition to the above loss terms, we minimize the distance between the embeddings extracted from the reconstructed face $\hat{\mathbf{I}}$ and original face $\mathbf{I}$. To this end, we minimize the square of the ℓ_2-norm of the difference between the extracted features:

$$\mathcal{L}_{\text{ID}}(\hat{\mathbf{I}},\mathbf{I})=||F(\hat{\mathbf{I}})-F(\mathbf{I})||_2^2 \quad (3)$$

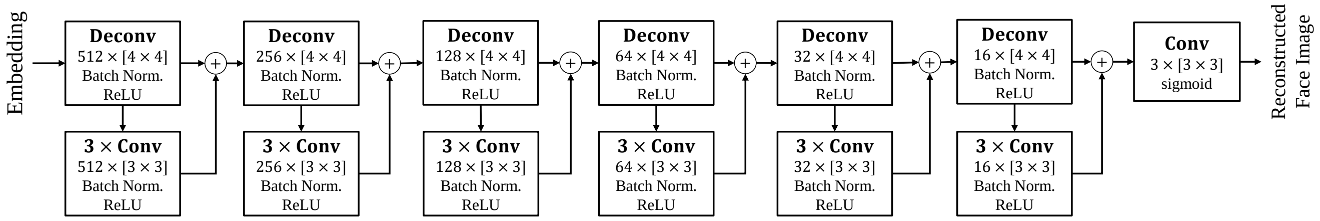


Fig. 3: Structure of the proposed face reconstruction network

We use a weighted summation of the aforementioned loss terms as our total loss:

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \alpha\mathcal{L}_{\text{DSSIM}} + \beta\mathcal{L}_{\text{ID}}, \tag{4}$$

where α and β are hyperparamters. We experimentally found that $\alpha = 0.1$ and $\beta = 0.005$ perform the best and we use these values for our final loss function. We train our face reconstruction network using the Adam [18] optimizer with the initial learning rate of 10^{-3}, and we decrease the learning rate by a factor of 0.5 every 10 epochs.

2.3. Network Structure

To reconstruct a face image from its corresponding embedding, we can use a deconvolutional neural network (e.g., [7, 9]). However, because deconvolution acts as upsampling it may generate a noisy result and insufficient details [9]. For this reason, we propose a new block using 3 cascaded convolutional layers with a skip connection, after each deconvolutional layer. In the proposed network, convolution layers are supposed to learn the residual and to enhance the deconvolution output.

We build our network using 6 of the proposed blocks with 512, 256, 128, 64, 32, 16 filters, respectively. For the deconvolution and convolution layers in our blocks, we use kernels of sizes 4 and 3, respectively. In addition, we use Batch Normalization [19] and a rectified linear unit (ReLU) after each deconvolution and convolution operation in our blocks. Finally, we use a convolutional layer with a kernel of size 3 and a sigmoid activation function, to generate the reconstructed face image. Fig. 3 depicts the general structure of our face reconstruction neural network.

3. EXPERIMENTS

In this section, we describe the experiments used to evaluate the performance of our face reconstruction network. First, in section 3.1 we describe our experimental setup. Next, we evaluate the performance of our reconstruction network in section 3.2 Finally, we provide an ablation study in section 3.3.

3.1. Experimental Setup

To evaluate the reconstruction performance of the proposed face reconstruction network, as stated in section 1, we train the network on two SOTA face recognition models, ArcFace [5] and ElasticFace [6]. For each model, we generate a training dataset as described in section 2.1, using the FFHQ [4] dataset and we train our face reconstruction network with the multi-term loss function proposed in Eq 4. Then, we evaluate the trained face reconstruction networks on the MOBIO [14] and Labeled Faced in the Wild (LFW) [15] databases. To evaluate the trained face reconstruction network, we first build a face recognition system. Next, we consider the scenario where the adversary gains access to the system's database and aims to reconstruct face images from the enrolled face embeddings, then enter the system by injecting the reconstructed image as a query to the system. Hence, we evaluate the performance of our face reconstruction network in terms of an adversary's Success Attack Rate (SAR) in entering the system using the reconstructed face images of the corresponding embeddings stored in the system's database.

The FFHQ [4] dataset consists of 70,000 high-quality face images and contains variations in terms of age, ethnicity, and gender. We use a random 90% portion of this dataset to generate the training dataset for our face reconstruction network, and we use the remaining 10% for validation. The MOBIO dataset is a bimodal dataset including face and audio data taken with mobile devices from 152 people. In our experiments, we use the *development* subset of the *mobio-all* protocol[1]. The LFW database includes 13,233 images of 5,749 people, where 1,680 people have two or more images. We use the *View 2* protocol[2] in our experiments. It is also worth mentioning that the studied face recognition models (ArcFace and ElasticFace) are trained using the MS-Celeb-1M [20] dataset.

We use the Bob[3] toolbox [21, 22] and PyTorch package in our implementations. The source code of our experiments is publicly available to help reproduce our results[4].

3.2. Performance Evaluation

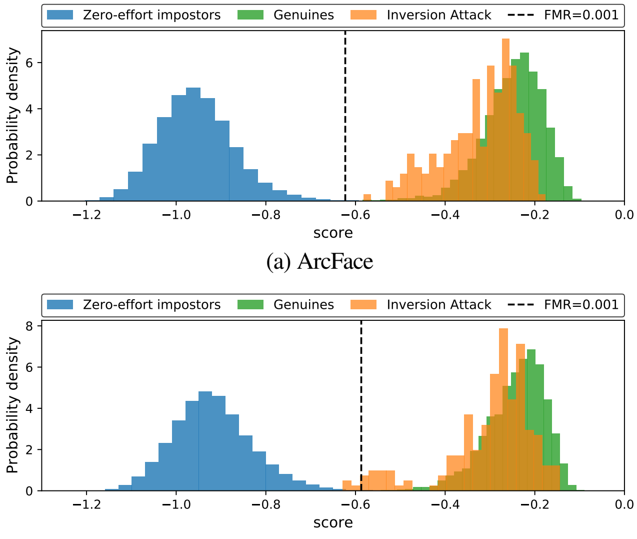
As described in section 3.1, we trained our network to reconstruct face images from ArcFace and ElasticFace embeddings. Fig. 1 illustrates sample face images from the validation set of the FFHQ dataset. Fig. 4 also shows the histogram of scores between embeddings extracted from the original and reconstructed face images, as

[1] The implementation of the *mobio-all* protocol for the MOBIO dataset is available at https://gitlab.idiap.ch/bob/bob.db.mobio

[2] The implementation of *View 2* protocol for the LFW dataset is available at https://gitlab.idiap.ch/bob/bob.db.lfw

[3] https://www.idiap.ch/software/bob/

[4] Source code: https://gitlab.idiap.ch/bob/bob.paper.icip2022_face_reconstruction



(a) ArcFace

(b) ElasticFace

Fig. 4: Histogram of scores (negative cosine distance) evaluated on the MOBIO dataset: a) ArcFace b)ElasticFace.

Table 1: Face reconstruction performance (SAR) of our network for embeddings extracted from the ArcFace and ElasticFace models, as well as the recognition performance (TMR) of each model, at FMR $= 10^{-2}$ and FMR $= 10^{-3}$ on the MOBIO and LFW datasets (the values are reported in percentage).

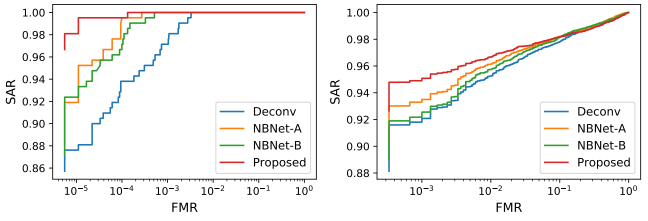
Model	**Dataset**	**FMR$=10^{-2}$**		**FMR$=10^{-3}$**	
		TMR	**SAR**	**TMR**	**SAR**
ArcFace	**MOBIO**	100.00	100.00	100.00	100.00
	LFW	99.70	96.67	96.63	94.90
ElasticFace	**MOBIO**	100.00	100.00	100.00	98.10
	LFW	96.96	95.21	94.60	91.78

well as scores of genuine and zero-effort impostor pairs, evaluated on the MOBIO dataset. Table 1 also reports the performance of our face reconstruction network in terms of SAR as well as the recognition performance of each model in terms of True Match Rate (TMR) at different False Match Rate (FMR) decision threshold configurations. As this table shows, the face images reconstructed using our network are highly likely to be recognized as face images coming from genuine (enrolled) users, when employing a face recognition system based on ArcFace or ElasticFace.

3.3. Ablation Study

In this section, we investigate the effect of our network structure and loss function on the reconstruction performance of our method. To this end, we use embeddings of the ArcFace model and report our ablation study on the MOBIO and LFW datasets.

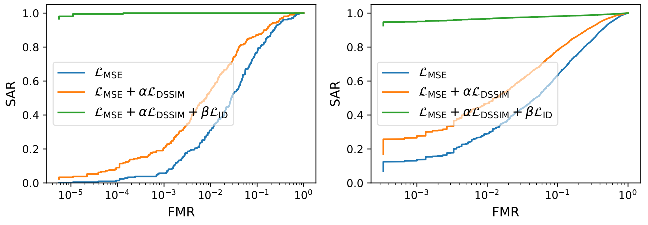
Network Structure To evaluate the effect of our network structure, we use our loss function and train similar networks based on NBNet-A [9], NBNet-B [9], and typical deconvolution blocks. Fig. 5 compares the reconstruction performance of these networks in terms of SAR for different values of the face recognition system's FMR. As this figure shows, the proposed network in general outperforms the other networks, particularly when the ArcFace-based face recognition system (which is being attacked) operates at a lower FMR.



(a) MOBIO

(b) LFW

Fig. 5: Effect of network structure on face reconstruction performance using ArcFace embeddings evaluated on a) MOBIO and b) LFW datasets.



(a) MOBIO

(b) LFW

Fig. 6: Effect of loss function on face reconstruction performance using ArcFace embeddings evaluated on a) MOBIO and b) LFW datasets.

Loss Function To evaluate the effect of our loss function, we train our network structure with the different loss terms from Eq. 4. Fig. 6 compares the reconstruction performance of networks trained with different loss functions in terms of SAR for different values of the system's FMR. As this figure shows, the DSSIM and ID loss terms enhance the reconstruction performance. In particular, the ID loss significantly improves the SAR.

4. CONCLUSION

In this paper, we proposed a CNN-based structure to reconstruct face images from face embeddings and trained it with a multi-term loss function. We used 3 convolution layers (with a skip connection) after each deconvolution layer to enhance the deconvolution output by learning the residual. We evaluated our proposed face reconstruction network for two SOTA face recognition models (ArcFace and ElasticFace) on the MOBIO and LFW datasets. Our experiments show that the reconstructed face images are highly likely to be recognized (as the face images of enrolled system users) by the face recognition system.

5. REFERENCES

[1] Battista Biggio, Paolo Russu, Luca Didaci, Fabio Roli, et al., “Adversarial biometric recognition: A review on biometric system security from the adversarial machine-learning perspective,” *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 31–41, 2015.

[2] Javier Galbally, Chris McCool, Julian Fierrez, Sebastien Marcel, and Javier Ortega-Garcia, “On the vulnerability of face verification systems to hill-climbing attacks,” *Pattern Recognition*, vol. 43, no. 3, pp. 1027–1038, 2010.

[3] Abdenour Hadid, Nicholas Evans, Sebastien Marcel, and Julian Fierrez, “Biometrics systems under spoofing attack: an evaluation methodology and lessons learned,” *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 20–30, 2015.

[4] Tero Karras, Samuli Laine, and Timo Aila, “A style-based generator architecture for generative adversarial networks,” *arXiv preprint arXiv:1812.04948*, 2018.

[5] Jiankang Deng, Jia Guo, Xue Niannan, and Stefanos Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper, “Elasticface: Elastic margin loss for deep face recognition,” pp. 1578–1587, 2022.

[7] Andrey Zhmoginov and Mark Sandler, “Inverting face embeddings with convolutional neural networks,” *arXiv preprint arXiv:1606.04189*, 2016.

[8] Forrester Cole, David Belanger, Dilip Krishnan, Aaron Sarna, Inbar Mosseri, and William T Freeman, “Synthesizing normalized faces from facial identity features,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3703–3712.

[9] Guangcan Mai, Kai Cao, Pong C Yuen, and Anil K Jain, “On the reconstruction of face images from deep face templates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 5, pp. 1188–1202, 2018.

[10] Chi Nhan Duong, Thanh-Dat Truong, Khoa Luu, Kha Gia Quach, Hung Bui, and Kaushik Roy, “Vec2face: Unveil human faces from their blackbox features in face recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 6132–6141.

[11] Edward Vendrow and Joshua Vendrow, “Realistic face reconstruction from deep embeddings,” in *Proceedings of NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021.

[12] Xingbo Dong, Zhe Jin, Zhenhua Guo, and Andrew Beng Jin Teoh, “Towards generating high definition face images from deep templates,” in *Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2021, pp. 1–11.

[13] Tero Karras, Samuli Laine, and Timo Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4401–4410.

[14] Chris McCool, Roy Wallace, Mitchell McLaren, Laurent El Shafey, and Sébastien Marcel, “Session variability modelling for face authentication,” *IET Biometrics*, vol. 2, no. 3, pp. 117–129, Sept. 2013.

[15] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[16] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[17] Sahar Sadrizadeh, Hatef Otroshi-Shahreza, and Farokh Marvasti, “Impulsive noise removal via a blind cnn enhanced by an iterative post-processing,” *Signal Processing*, vol. 192, pp. 108378, 2022.

[18] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, California., USA, May 2015.

[19] Sergey Ioffe and Christian Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proceedings of the International Conference on Machine Learning (ICML)*, Lille, France, Jul. 2015, pp. 448–456.

[20] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao, “Ms-celeb-1m: A dataset and benchmark for large-scale face recognition,” in *Proceedings of European Conference on Computer Vision (EECV)*. Springer, 2016, pp. 87–102.

[21] A. Anjos, L. El Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel, “Bob: a free signal processing and machine learning toolbox for researchers,” in *Proceedings of the 20th ACM Conference on Multimedia Systems (ACMMM)*, Oct. 2012.

[22] A. Anjos, M. Günther, T. de Freitas Pereira, P. Korshunov, A. Mohammadi, and S. Marcel, “Continuously reproducing toolchains in pattern recognition and machine learning experiments,” in *Proceedings of the International Conference on Machine Learning (ICML)*, Aug. 2017.