Generating Master Faces for Use in Performing Wolf Attacks on Face Recognition Systems

Generating Master Faces for Use in Performing Wolf Attacks on Face Recognition Systems

Huy H. Nguyen[1], Junichi Yamagishi[1,2,4], Isao Echizen[1,2,3], and Sébastien Marcel[5]
[1]The Graduate University for Advanced Studies, SOKENDAI, Kanagawa, Japan
[2]National Institute of Informatics, Tokyo, Japan; [3]The University of Tokyo, Japan
[4]The University of Edinburgh, Edinburgh, UK; [5]Idiap Research Institute, Martigny, Switzerland
Email: {nhhuy, jyamagis, iechizen}@nii.ac.jp, marcel@idiap.ch

Abstract

Due to its convenience, biometric authentication, especial face authentication, has become increasingly mainstream and thus is now a prime target for attackers. Presentation attacks and face morphing are typical types of attack. Previous research has shown that finger-vein- and fingerprint-based authentication methods are susceptible to wolf attacks, in which a wolf sample matches many enrolled user templates. In this work, we demonstrated that wolf (generic) faces, which we call "master faces," can also compromise face recognition systems and that the master face concept can be generalized in some cases. Motivated by recent similar work in the fingerprint domain, we generated high-quality master faces by using the state-of-the-art face generator StyleGAN in a process called latent variable evolution. Experiments demonstrated that even attackers with limited resources using only pre-trained models available on the Internet can initiate master face attacks. The results, in addition to demonstrating performance from the attacker's point of view, can also be used to clarify and improve the performance of face recognition systems and harden face authentication systems.

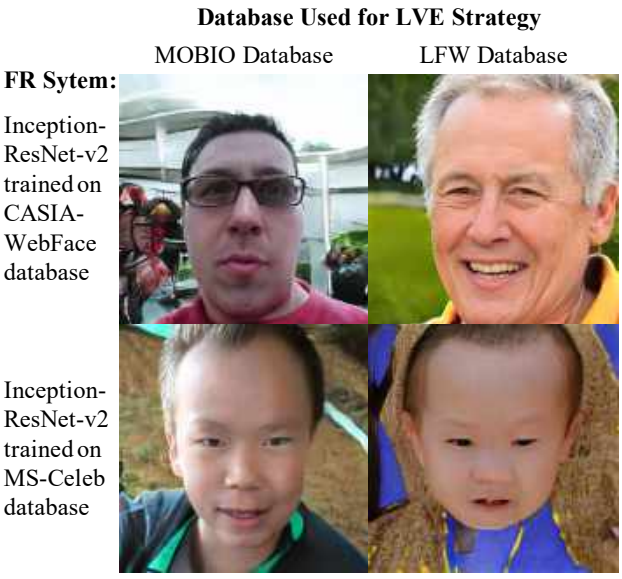

Figure 1. Example master faces generated with our method. Images in first and second column were generated using the MOBIO [21] and Labeled Faces in the Wild [19] databases, respectively. Training for images in first row used Inception-ResNet-v2 [33] based FR system trained on CASIA-WebFace database [40] while that for ones in second row used same FR system trained on MS-Celeb database [10].

1. Introduction

Recent advances in the development of biometric authentication, especially in its ease of use, have enabled face authentication (which uses face recognition) to be implemented in many portable and handheld devices, from laptop PCs to smartphones. Digital wallets, which are also called "e-wallets" and are popular in many countries, also utilize face authentication from the user's smartphone to process payments. As a result, face authentication systems have become a prime target for attackers. Even before this trend, interest in creating a face that matches multiple faces led researchers to come up with the idea of face morphing [30], which is a special case of image morphing [38]. Given two or more faces from different identities, a system creates a blended face that can match all component identities when using face recognition (FR) systems and possibly fool a human observer. Morphing attacks often target automated border control systems, possibly by criminals to avoid being detected [30].

Another kind of attack is called a "wolf attack," which targets biometric authentication systems by finding or crafting a generic sample, a "wolf sample," that has high sim-

ilarity to many of the enrolled templates [36]. The advantage of a wolf attack is that it does not require knowledge about the enrolled subjects. Wolf attacks are commonly aimed at finger-vein- and fingerprint-based recognition systems [36, 5]. Bontrager et al. introduced a generative adversarial network (GAN) [7] based method for generating realistic fingerprint images ("MasterPrints") using the latent variable evolution (LVE) strategy for use in attacking systems using partial fingerprint images [5]. Although a defense method was subsequently proposed [27], this kind of attack has much room for improvement.

In this work, we demonstrated that wolf attacks can also compromise FR systems. An LVE algorithm running on a pre-trained high-quality face generator StyleGAN [15] was able to match selected **master face** (wolf) samples with multiple user templates (in-domain and out-of-domain) for both known and unknown FR systems. A **known** FR system is the one used on the running LVE algorithm while an **unknown** FR system can be understood as one with the same architecture but trained on a different database or one with a completely different architecture. We modified the LVE algorithm [5] by changing the way it calculates the scores of the generated faces. The master faces can be easily and quickly generated by simply using pre-trained models available on the Internet. Examples are shown in Figure 1.

We used a combination of a StyleGAN model pre-trained on the Flickr-Faces-HQ (FFHQ) database [15] and an Inception-ResNet-v2 [33] based FR system pre-trained by de Freitas Pereira et al. [6] on the CASIA-WebFace database [40] (available in Bob toolbox [1]) to generate the master faces. With this combination and less than 24 hours of training on a conventional personal computer (PC) without a graphics processing unit (GPU), we generated master faces that achieved false acceptance rates (FARs) between 6 and 35% depending on the test database and targeted FR system. These high rates raise a major concern about the ability of FR systems to deal with a master face attack, which can be launched by someone without any special training. In addition to considering the attacker's point of view, we also consider countermeasures to mitigate this threat.

The rest of the paper is organized as follows. First, we present general information about facial image generation, FR systems, wolf attacks, and the LVE algorithm in section 2. We then introduce our proposed method in section 3 and describe our experimental design, present the results, and discuss them in section 4. Next, we discuss possible ways in the literature to defend against master face attacks in section 5. Finally, we summarize the key points and mention future work in section 6.

2. Related Work

2.1. Facial Image Generation

Facial image generation has recently been attracting the attention of the research community, especially since the introduction of variational autoencoders (VAEs) [18] and GANs [7]. Initially, only facial images with low resolution and sizes were generated. In addition, VAEs suffer a trade-off between disentangled representations and reconstruction errors while GANs are difficult to train. To solve the later problem, Arjovsky et al. proposed the Wasserstein GAN (WGAN), which improves the stability of learning and eliminates the mode collapse problem [2]. Subsequent work led to an improved WGAN called WGAN-GP in which the weight clipping is replaced with a gradient-based penalty function [9]. WGAN [2, 9] was used in the work of Bontrager et al. to generate master prints for use in attacking partial fingerprint authentication systems [5].

Despite these improvements, GANs still suffer the problem of generating high-resolution images. Karras et al. proposed a training methodology in which both the generator and discriminator are progressively trained [14]. It starts with a low-resolution image model and then repeatedly adds new layers to the model to incorporate fine details. Using this idea of progressive training and borrowing the idea of image generating from the style transfer field, Karras et al. introduced a novel face generator network called StyleGAN [15]. This network has the ability to automatically learn and separate high-level attributes (such as pose and identity) and stochastic variation in the generated images (such as freckles and hair). Unlike traditional GANs, StyleGAN includes two components: a mapping network that maps the input latent vector to intermediate style vectors and feeds them into the synthesis network. With these two components, StyleGAN handles disentanglement well and supports style mixing. Subsequent work focused on analyzing and improving the quality of StyleGAN generated imags [16]. In this work, we used StyleGAN [15] for facial image generation.

2.2. Face Recognition Systems

Recent advances in convolutional neural networks (CNNs) and the releases of large annotated databases substantially improved the performances of FR systems, enabling them to be applied to not only homogeneous but also heterogeneous domains [6]. Two examples of such large databases are the CASIA-WebFace database [40] and the MS-Celeb database [10], which are commonly used to create training data for state-of-the-art FR systems. Smaller well-known databases that had been previously released, like the MOBIO database [21] and the Labeled Faces in the Wild (LFW) database [19], are usually used for validation. Reusing an architecture that performed well in

the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [28] as a feature extractor for CNN-based FR systems, rather than designing a new architecture from scratch, is a commonly used approach. The two most commonly used architectures are the VGG (Visual Geometry Group) network [32] and the Inception network [33].

Parkhi et al. adopted the VGG-16 network [32] for an FR task ("VGG-Face") and trained it on a custom-built large-scale database [25]. Wu et al. introduced a light CNN that uses max-feature-map activation and has ten times fewer parameters than the VGG-Face network [39]. The Inception architecture [33] was used by Schroff et al. to build the FaceNet model, which maps facial images to a compact Euclidean space embedding [31]. Therefore, it can be used for face verification, recognition, and clustering. The closet open-source implementation of FaceNet was done by David Sandberg [29] using the Inception-ResNet v1 and v2 architectures [33]. Additionally, de Freitas Pereira et al. used the Inception-ResNet v2 architecture in their heterogeneous FR work [6]. Experiments demonstrated that Inception-based methods perform better than VGG-based ones. Using another approach, Tran et al. proposed a disentangled representation learning GAN method ("DR-GAN") that can deal with face variations, especially in pose [35]. In our experiments, we used three FR systems: (1) Inception-ResNet v1 based FaceNet by David Sandberg [29], (2) Inception-ResNet v2 network by de Freitas Pereira et al. [6], and (3) DR-GAN by Tran. et al. [35].

2.3. Wolf Attacks

For biometric authentication systems, Une et al. [36] defined a "wolf sample" as an input sample that can be falsely accepted as a match with multiple user templates (enrolled subjects). They also defined a measurement called wolf attack probability (WAP), which is the maximum probability of a successful attack with one wolf sample. Wolf attacks and defenses against them are the subject of much research in the finger-vein and fingerprint recognition fields. Wolf attacks has been evolved from generating forged minutiae [26] to generating real partial fingerprint images [5]. In the latest work [5], Bontrager et al. used a latent variable evolution algorithm to maximize the WAPs of partial fingerprint images generated by a GAN. This type of attack is applicable to systems using small-size sensors with limited resolution. In contrast, in this work, we focused on generating high-quality high-resolution facial images for use in attacking FR systems using full-face input.

2.4. Latent Variable Evolution Algorithm

Inspired by biological evolution, evolution algorithms have long been used in artificial intelligence applications without any assumption about the underlying fitness landscape. One such algorithm is the evolution strategies (ES) algorithm, which can be used for complex, multimodal, and non-differentiable functions. Proposed by Hansen and Ostermeier, the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is a powerful ES algorithm designed for non-linear and non-convex functions [12]. Bontrager et al. demonstrated that interactive evolutionary computation can be used in combination with a GAN [4]. After the GAN is trained, a latent vector used as input to the GAN can be put under evolutionary control, resulting in the generation of high-quality samples. Bontrager et al. proposed combining this method with the CMA-ES algorithm [12] to generate partial fingerprints. The resulting LVE algorithm maximizes the WAP of the generated partial fingerprint images against a fingerprint authentication system [5]. Following this success in the fingerprint domain, we modified the scoring method used for the LVE algorithm and applied the resulting algorithm to the facial domain, which is trickier since human vision is more sensitive to faces than fingerprints. With the help of the StyleGAN high-quality face generator [15] and the powerful CMA-ES algorithm [12], our proposed method can generate high-resolution master faces that are both natural looking and have a high WAP.

3. Proposed Method

3.1. Overview

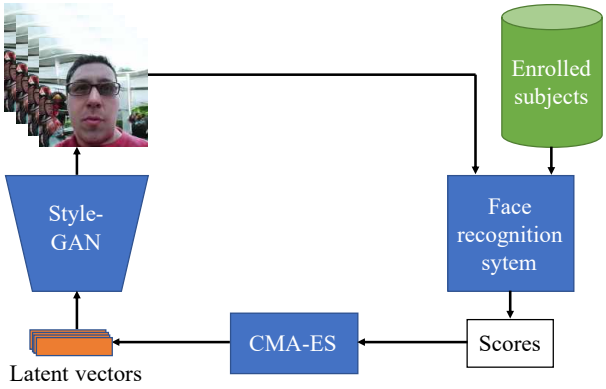

Figure 2. Overview of proposed method. With default setting, 22 latent vectors with a dimensionality of 512 are fed into StyleGAN [15] to generate 22 facial images. The surrogate FR system then calculates mean score for each image on basis of enrolled subjects. CMA-ES [12] algorithm then uses scores to estimate 22 new latent vectors.

An overview of the proposed method is shown in Figure 2. In addition to the LVE algorithm, we need a pre-trained face generator (in this case, StyleGAN [15]), a surrogate pre-trained FR system (used to approximate the target FR system), a surrogate face database, and an implementation of the CMA-ES algorithm [12]. If a pre-trained generator is not available, one must be trained from scratch:

- Prepare three or four face databases: one to train StyleGAN, one or two to train the FR system (in the case of two, one is used for validation), and one to run the LVE algorithm. Some or all of them could overlap; in our work, we used the hardest case, i.e., non-overlapping, to demonstrate the generalizability of our proposed method.
- Prepare and train FR system.
- Prepare and train StyleGAN.
- Implement or use open-source library for CMA-ES algorithm, e.g., pycma [11].
- Run LVE algorithm.

3.2. Latent Variable Evolution

Algorithm 1 Latent variable evolution.

$m \leftarrow 22$ ▷ Population size
procedure RUNLVE(m, n)
  MasterFaces = {} ▷ Master face set
  MasterScores = {} ▷ and the corresponding score set
  $\mathbf{z} \leftarrow 0$ ▷ Initialize latent vectors $\boldsymbol{z} \in \mathbb{R}^m$
  for n iterations **do** ▷ Run LVE algorithm n times
    $F \leftarrow$ StyleGAN($\boldsymbol{z}$) ▷ Generate m faces F
    $\boldsymbol{s} \leftarrow 0$ ▷ Initialize scores $\mathbf{s} \in \mathbb{R}^m$
    for face F_i in faces F **do**
      for face E_j in data E **do**
        $s_i \leftarrow s_i +$ FaceMatching(F_i, E_j)
      $\boldsymbol{s} \leftarrow \frac{\boldsymbol{s}}{|E|}$ ▷ Calculate the mean scores
      $F_b, s_b \leftarrow$ GetBestFace($F, \boldsymbol{s}$)
      MasterFaces.append(F_b)
      MasterScores.append(s_b)
      $\boldsymbol{z} \leftarrow$ CMA_ES($\boldsymbol{s}$) ▷ Evolve $\boldsymbol{z}$ based on $\boldsymbol{s}$
  return MasterFaces, MasterScores
$F_b, s_b \leftarrow$ GetBestFace(MasterFaces, MasterScores)

The LVE procedure is formalized in Algorithm 1. The FaceMatching(F_i, E_j) function calculate the similarity between two input faces F_i and E_j. As the default setting for the CMA-ES library [11], we set population size m to 22. Unlike the algorithm of Bontrager et al. [5], our algorithm does not require information about the pre-defined false matching rate (FMR). Moreover, the accumulated matching scores $\boldsymbol{s}$ are not added in a binary way (matched or unmatched: FaceMatching($\cdot, \cdot$) function only returns 1 or 0, respectively) but instead by using the actual similarity scores calculated by the FR system. As a result, the CMA-ES algorithm tries to maximize the $\boldsymbol{s}$ in each iteration. Therefore, the optimization curve is smoother than that of the Bontrager's LVE algorithm, especially when the training data for StyleGAN, the FR system, and this LVE algorithm (E) differ. One example is that, if there are no matches, the accumulated score s_i of each generated face F_i is 0 with the Bontrager's algorithm, whereas the CMA-ES algorithm has no clue to use in evolving $\boldsymbol{z}$. This problem is solved by using the actual scores for $\boldsymbol{s}$.

The local best master face F_b is selected among m master faces and collected after each iteration on the basis of its score s_b, which is also logged. After n iterations, the final (global) best master face is chosen among n best master faces on the basis of the logged scores. The reason we do this instead of selecting the best master face of all master faces in every iteration is that (1) it reduces the storage of master faces and (2), if the number of iterations is large enough, besides getting better, the m master faces generated in each iteration get closer to each other (by identity, appearance, background, and pose). Therefore, selecting the best one among them is sufficient. The running of the LVE algorithm on two databases based on the FMR and t-distributed stochastic neighbor embedding (t-SNE) visualization [20] of the master faces at certain iterations is described in section 4.2.

4. Evaluation

In this section, we first describe our experimental design, including the FR systems and databases we used (section 4.1). We then describe the running of the proposed LVE algorithm on the LFW - Fold 1 database (scenario 1) and MOBIO database (scenario 2) and the analysis of its behavior (section 4.2). Finally, we describe the calculation of the attack success probabilities of the obtained master faces for both scenario 1 (section 4.3) and 2 (section 4.4).

4.1. Experimental Design

4.1.1 Face Recognition Systems

We used four pre-trained FR systems supported by the Bob toolbox [1]:

- Inception-ResNet-v2 [33] based FR systems trained by de Freitas Pereira et al. [6]: one trained on the CASIA-WebFace database [40] and one trained on the MS-Celeb database [10]. **We used the one trained on the CASIA-WebFace database to run the LVE algorithm**.
- FaceNet (using the Inception-v1 architecture [34]) proposed by Schroff et al. [31], implemented and trained by David Sandberg [29] on the MS-Celeb database [10].
- DR-GAN proposed and implemented by Tran et al. [35], pre-trained on a combination of the Multi-PIE database [8] and the CASIA-WebFace database [40].

4.1.2 Databases

Beside the databases used to train the FR systems mentioned above (CASIA-WebFace [40], MS-Celeb [10], and Multi-PIE [8]), we used the MOBIO database [21] with both male and female components and the LFW database [19] aligned by funneling [13] to run the LVE algorithm and validate the master faces. The LFW database has several protocols; we used the fold 1 protocol. The MOBIO and LFW - Fold 1 databases were divided into two mutually exclusive sets (with non-overlapping identities): a world set used for training and a development (dev) set used for threshold selection for the FR systems (based on the equal error rate, EER), and an evaluation (eval) set. The dev and eval sets were both used to evaluate the master faces. In the dev and eval sets, the test pairs included both genuine and zero-effort imposter cases. To test the master faces, we replaced the test pairs by matching the master faces with all enrolled subjects and measured the false matching rates (FMRs).

The StyleGAN face generator was pre-trained on the Flickr-Faces-HQ (FFHQ) database [15]. For this training, there were no overlaps between the databases used for training the FR systems, training StyleGAN, and running the LVE algorithm. We wanted to demonstrate the generalizability of our proposed method since attackers often lack the knowledge and resources needed to perform this kind of attack and therefore tend to use resources widely available on the Internet.

4.2. Running Latent Variable Evolution

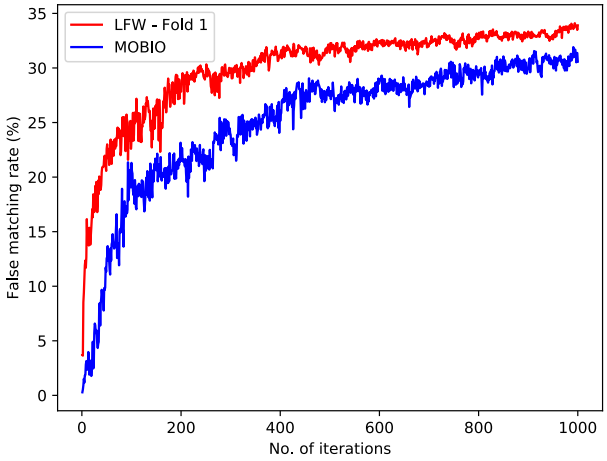

Figure 3. FMR for Inception-ResNet-v2 based FR system [33] when running LVE algorithm on LFW - Fold 1 database [19] and MOBIO database [21].

We ran the LVE algorithm on two databases (LFW - Fold 1 [19] - scenario 1 and MOBIO [21] - scenario 2) with the Inception-ResNet-v2 based FR system [33] trained on the CASIA-WebFace database [40]. We ran it on a PC without a GPU for only 1000 iterations, which took less than 24 hours per database. The FMRs for the two databases are plotted in Figure 3. Since the MOBIO database has high variability in the pose and illumination conditions compared with the LFW - Fold 1 database, which greatly affected the FR system (the selected threshold was trickier), the FMRs for the MOBIO database were lower than those for the LFW database. They were about 35% for the LFW - Fold 1 database and 30% for the MOBIO database at the 1000th iteration. Nevertheless, both curves still tend to increase. We limited the number of iterations to demonstrate that the attack can be done in a limited time. Comprehensive analysis will be done in follow-up work.

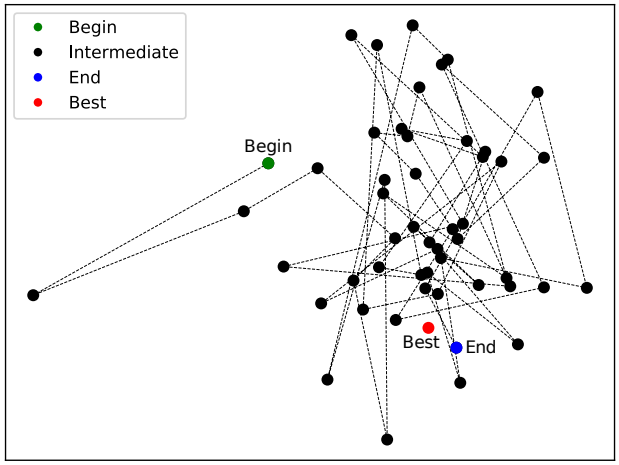

Figure 4. T-SNE visualization of master faces obtained every 20 iterations (1000 in total) on LFW - Fold 1 database [19]. Green dot represents master face at beginning; it is connected by dashed lines with intermediate master faces (black dots) that end at blue dot. Red dot represents best master face, created at 989 th iteration; therefore, it does not overlap any black dot.

T-SNE visualization [20] of the process of running the LVE algorithm on the LFW - Fold 1 database is shown in Figure 4. Initially, the CMA-ES algorithm was unsure about the optimal direction. After finding some clues, it began generating master faces that jumped around the best master face (the red dot) and came closer and closer to it.

4.3. Scenario 1: LFW - Fold 1 Database

In this scenario, we ran the LVE algorithm on the LFW - Fold 1 database [19] using the Inception-ResNet-v2 based FR system [33]. The best master face is shown in Figure 1 (top right). We then tested it using three FR systems with four configurations on the LFW - Fold 1 database [19] and the MOBIO database [21]. The results are shown in Figure 5 and summarized in Table 1.

We tried to match the obtained master face with all enrolled faces in the dev and eval sets of the LFW - Fold 1 database. A score histogram for the master face is plotted

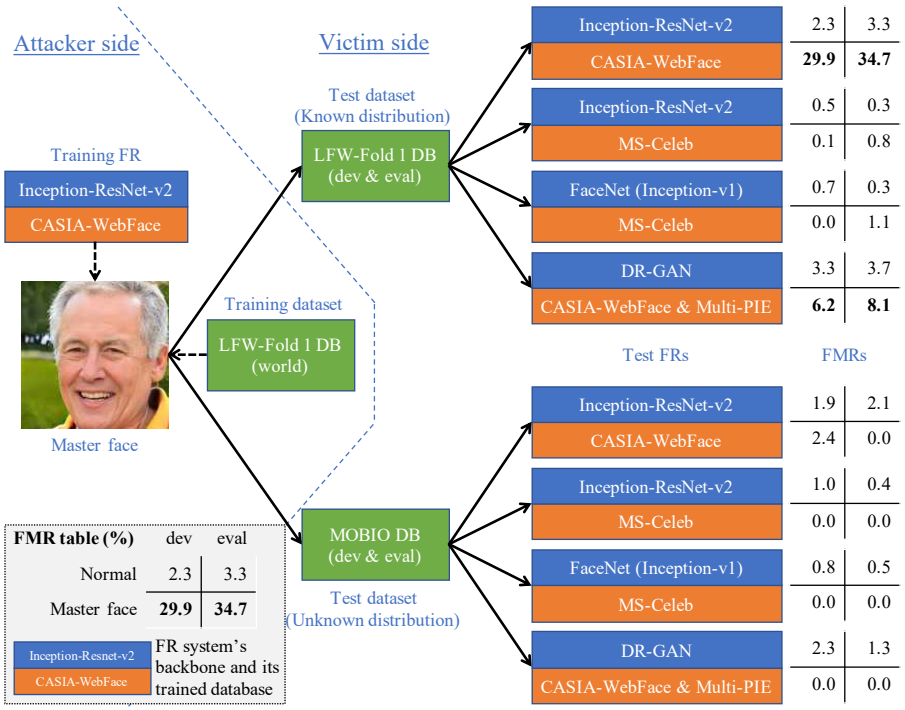

Figure 5. FMRs of original test designs and of master face generated using LFW - Fold 1 database [19] calculated using four configurations of three FR systems on LFW - Fold 1 database [19] and MOBIO database [21].

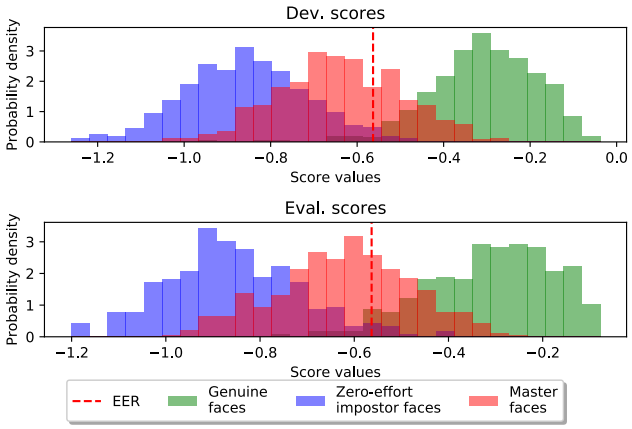

Figure 6. Histogram of scores for genuine faces, zero-effort imposter faces, and master face generated using LFW - Fold 1 database [19] calculated using Inception-ResNet-v2 based FR system [33].

in Figure 6 along with those for the genuine faces and the zero-effort imposter faces from the original test design of the database. The master face scores moved away from the zero-effort imposter scores in the direction of the genuine face scores with about 30–35% overlap. This means that the master face matched 30–35% of the enrolled faces, which is significant.

As shown in Figure 5, the wolf attack worked on two FR systems on the LFW - Fold 1 database: the Inception-ResNet-v2 based one trained on the CASIA-WebFace database (which was also used to train the master face) and the DR-GAN one trained on a combination of the CASIA-WebFace and Multi-PIE databases. The results for the DR-GAN FR system demonstrate that the proposed method is generalizable to other FR system architectures. The wolf attack did not work in two cases:

- LFW - Fold 1 database: The Inception-ResNet-v2 and Inception-v1 based FR systems were trained on the MS-Celeb database. Since the MS-Celeb database is larger than the CASIA-WebFace one, the FR systems trained on it were more robust than the others, which can be observed from the FMRs for the normal test cases without master faces. Since the master face was trained using the FR system trained on the CASIA-WebFace database, it was not strong enough to work on the normal cases.

- MOBIO database: As we mentioned above, the MOBIO database has high variability in the pose and illumination conditions compared with the LFW - Fold 1 database; therefore, the master face generated using the LFW - Fold 1 database was not sophisticated enough to work with the MOBIO database.

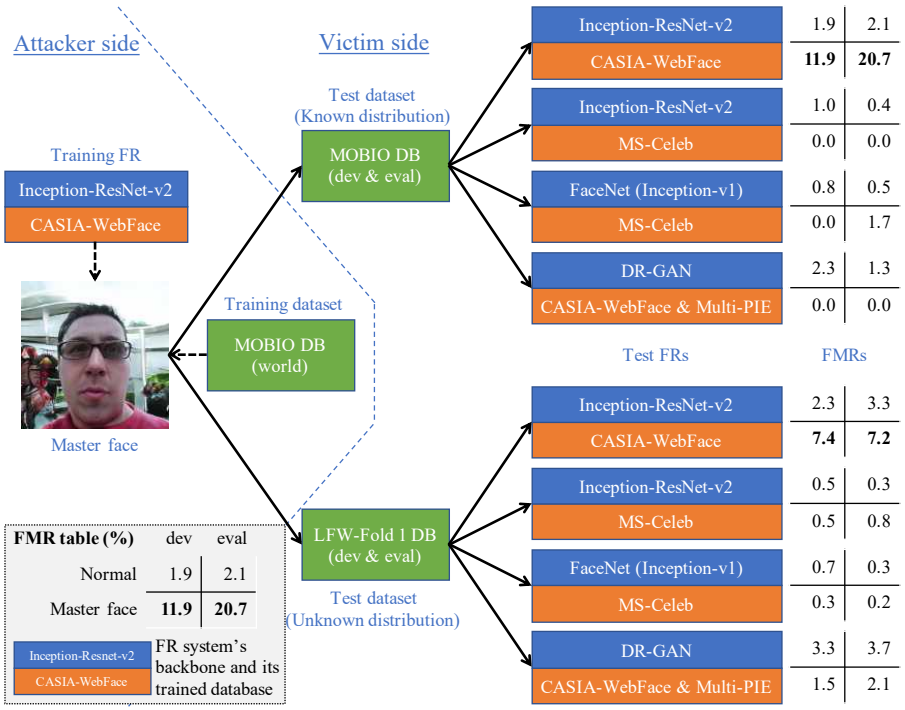

Figure 7. FMRs of original test designs and of master face generated using MOBIO database [21] calculated using four configurations of three FR systems on MOBIO database [21] and LFW - Fold 1 database [19].

FR Setting	Scenario 1		Scenario 2	
	Known DB	Unknown DB	Known DB	Unknown DB
Same Arch. - Same DB	**1**	0	**1**	**1**
Same Arch. - Different DB	0	0	0	0
Different Arch. - Same DB	**1**	0	0	0
Different Arch. - Different DB	0	0	0	0

Table 1. Summary of successful attacks for scenarios 1 and 2 with different FR system settings (including their architecture and database used to train them) and databases.

4.4. Scenario 2: MOBIO Database

In this scenario, we ran the LVE algorithm on the MOBIO database [21] using the Inception-ResNet-v2 based FR system [33]. The best master face is shown in Figure 1 (top left). As in the previous scenario, we tested it using three FR systems with four configuration on the MOBIO database [21] and the LFW - Fold 1 database [19]. The results are shown in Figure 7 and are summarized in Table 1.

The wolf attack worked on the Inception-ResNet-v2 based FR system trained on the CASIA-WebFace database (which was also used to train the master face) on both the MOBIO and LFW - Fold 1 databases. The FMRs were about 12–20% on the MOBIO database and 7%–12% on the LFW - Fold 1 database. These results demonstrate that the proposed method is generalizable to different databases. This is because the MOBIO database is more sophisticated than the LFW - Fold 1 database. Unfortunately, it did not work on the other FR systems. One possible explanation is that since the MOBIO database is sophisticated, the LVE algorithm was trying to "overfit" the used FR system with master faces that were uncommon to other FR systems to increase the FMRs.

Two examples of all faces matched are shown in Figure 8 (eval set for MOBIO database) and Figure 9 (dev set for LFW - Fold 1 database). The images were sorted from nearest to furthest. The master face matched both male and female subjects of different races with different skin tones, poses, and illumination, with or without beard, glasses, hat, and scarf. The master face was trained on the MOBIO database, which has many selfie-like photos; therefore, it also had an appearance of a selfie photo.

In addition to helping us understand the attacker's point of view, the obtained results can be used to identify the weaknesses of an FR system. For example, the Inception-ResNet-v2 based FR system trained on the CASIA-WebFace database has two weaknesses: it has trouble distinguishing between male and female and under-

standing racial differences. Another example comes from Fig 1 shown in section 1. The Inception-ResNet-v2 based FR system trained on the MS-Celeb database seems to be poor at recognizing images of children as two of the master faces are of children. One possible explanation is that the MS-Celeb database lacks photographs of children since most celebrities are teenagers or adults.

Figure 8. Master face (bordered in red) and all matching enrolled faces sorted from nearest to furthest for eval set of MOBIO database [21].

Figure 9. Master face (bordered in red) and all matching enrolled faces sorted from nearest to furthest for dev set of LFW - Fold 1 database [19].

5. Defense Against Master Face Attack

To prevent such kinds of attack, besides improving FR systems, we need to use an additional detector to filter out master faces. For camera-based FR and face authentication systems, using a presentation attack detector [3] is a good option. If the system takes a digital image as input, a computer-generated/manipulated image detector [37] is needed. However, the generalizability of such detectors is a major concern [17], especially when the master faces were generated using databases that have a different distribution from those covered by the detectors. Although recent work has addressed this problem, performance on cross-databases is still limited [24, 22, 23] and needs further improvement.

6. Summary and Future Work

Aimed at simplicity by using available resources easily obtained on the Internet, including a pre-trained StyleGAN model, a pre-trained face recognition system, a face database, and a conventional PC without a GPU, our proposed method can generate master faces in less than a day. Beside the ability to attack seen data and seen face recognition systems (white box attacks), the master faces, in some cases, can be generalized. This discovery raises concerns about the robustness of face recognition and face authentication systems. Moreover, the properties of the master faces can provide clues for understanding and improving FR systems. Although countermeasures can be used to mitigate this type of attack, further research is needed to make face authentication systems more secure, especially when they are used in applications related to finance and personal data.

Future work will mainly focus on deep analysis of the properties of the master faces, mostly about the correlation between the skin color, race, gender, age, and pose of the master face and their proportion in the data used for running LVE. Another important task is to perform more experiments on more face recognition systems and more databases and to improve the generalizability of the master faces on them. One possible solution for the generalizability is using multiple surrogate face recognition systems for LVE. The recently released StyleGAN 2 [16] will be used instead of the previous version to improve the quality of the generated images.

Acknowledgements

This research was supported by JSPS KAKENHI Grants JP16H06302, JP17H04687, JP18H04120, JP18H04112, JP18KT0051, and by JST CREST Grant JPMJCR18A6, Japan.

We would like to thank Dr. Tiago de Freitas Pereira and Dr. Amir Mohammadi from Biometrics Security and Privacy (BSP) group at Idiap for providing the pre-trained face recognition systems and for their supports on Bob toolkit.

References

[1] 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 *ICML*, Aug. 2017.

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223, 2017.

[3] S. Bhattacharjee, A. Mohammadi, A. Anjos, and S. Marcel. Recent advances in face presentation attack detection. In *Handbook of Biometric Anti-Spoofing*, pages 207–228. Springer, 2019.

[4] P. Bontrager, W. Lin, J. Togelius, and S. Risi. Deep interactive evolution. In *International Conference on Computational Intelligence in Music, Sound, Art and Design*, pages 267–282. Springer, 2018.

[5] P. Bontrager, A. Roy, J. Togelius, N. Memon, and A. Ross. Deepmasterprints: Generating masterprints for dictionary attacks via latent variable evolution. In *BTAS*, pages 1–9. IEEE, 2018.

[6] T. de Freitas Pereira, A. Anjos, and S. Marcel. Heterogeneous face recognition using domain specific units. *IEEE Transactions on Information Forensics and Security*, 14(7):1803–1816, 2018.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[8] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, 2010.

[9] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein GANs. In *NIPS*, pages 5767–5777, 2017.

[10] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *ECCV*, pages 87–102. Springer, 2016.

[11] N. Hansen, Y. Akimoto, and P. Baudis. CMA-ES/pycma on Github. Zenodo, DOI:10.5281/zenodo.2559634, Feb. 2019.

[12] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001.

[13] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*. IEEE, 2007.

[14] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018.

[15] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410. IEEE, 2019.

[16] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of stylegan. *arXiv preprint arXiv:1912.04958*, 2019.

[17] A. Khodabakhsh, R. Ramachandra, K. Raja, P. Wasnik, and C. Busch. Fake face detection methods: Can they be generalized? In *BIOSIG*, pages 1–6. IEEE, 2018.

[18] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[19] G. B. H. E. Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[20] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[21] C. McCool, S. Marcel, A. Hadid, M. Pietikäinen, P. Matejka, J. Cernockỳ, N. Poh, J. Kittler, A. Larcher, C. Levy, et al. Bimodal person recognition on a mobile phone: using mobile phone data. In *ICMEW*, pages 635–640. IEEE, 2012.

[22] A. Mohammadi, S. Bhattacharjee, and S. Marcel. Domain adaptation for generalization of face presentation attack detection in mobile settings with minimal information. In *ICASSP*. IEEE, 2020.

[23] A. Mohammadi, S. Bhattacharjee, and S. Marcel. Improving cross-dataset performance of face presentation attack detection systems using face recognition datasets. In *ICASSP*. IEEE, 2020.

[24] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. In *BTAS*. IEEE, 2019.

[25] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *BMVC*, pages 41.1–41.12. British Machine Vision Association, 2015.

[26] N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM systems Journal*, 40(3):614–634, 2001.

[27] A. Roy, N. Memon, and A. Ross. Masterprint attack resistance: A maximum cover based approach for automatic fingerprint template selection. In *BTAS*. IEEE, 2019.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 2015.

[29] D. Sandberg. Facenet: face recognition using tensorflow. https://github.com/davidsandberg/facenet, 2017.

[30] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch. Face recognition systems under morphing attacks: A survey. *IEEE Access*, 7:23012–23026, 2019.

[31] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823. IEEE, 2015.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[33] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[35] L. Tran, X. Yin, and X. Liu. Disentangled representation learning GAN for pose-invariant face recognition. In *CVPR*, pages 1415–1424. IEEE, 2017.

[36] M. Une, A. Otsuka, and H. Imai. Wolf attack probability: A new security measure in biometric authentication systems. In *ICB*, pages 396–406. Springer, 2007.

[37] L. Verdoliva. Media forensics and deepfakes: an overview. *arXiv preprint arXiv:2001.06564*, 2020.

[38] G. Wolberg. Image morphing: a survey. *The visual computer*, 14(8-9):360–372, 1998.

[39] X. Wu, R. He, Z. Sun, and T. Tan. A light CNN for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security*, 13(11):2884–2896, 2018.

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