UniID: Spoofing Face Authentication System by Universal Identity

Zhihao Wu
USSLAB, Zhejiang University
zhihaowu@zju.edu.cn

Yushi Cheng†
ZJU-UIUC Institute, Zhejiang University
yushicheng@zju.edu.cn

Shibo Zhang
USSLAB, Zhejiang University
zhsb@zju.edu.cn

Xiaoyu Ji†
USSLAB, Zhejiang University
xji@zju.edu.cn

Wenyuan Xu
USSLAB, Zhejiang University
wyxu@zju.edu.cn

***Abstract*—Face authentication systems are widely employed in access control systems to ensure the security of confidential facilities. Recent works have demonstrated their vulnerabilities to adversarial attacks. However, such attacks typically require adversaries to wear disguises such as glasses or hats during every authentication, which may raise suspicion and reduce their attack impacts. In this paper, we propose the UniID attack, which allows multiple adversaries to perform face spoofing attacks without any additional disguise by enabling an insider to register a universal identity into the face authentication database by wearing an adversarial patch. To achieve it, we first select appropriate adversaries through feature engineering, then generate the desired adversarial patch with a multi-target joint-optimization approach, and finally overcome practical challenges such as improving the transferability of the adversarial patch towards black-box systems and enhancing its robustness in the physical world. We implement UniID in laboratory setups and evaluate its effectiveness with six face recognition models (FaceNet, Mobile-FaceNet, ArcFace-18/50, and MagFace-18/50) and two commercial face authentication systems (ArcSoft and Face++). Simulation and real-world experimental results demonstrate that UniID can achieve a max attack success rate of 100% and 79% in 3-user scenarios under the white-box setting and black-box setting respectively, and it can be extended to more than 8 users.**

I. Introduction

Face authentication systems verify the legitimacy of a user by matching its face image with the corresponding user ID stored within the database. Such systems are extensively applied in various security-sensitive scenarios, including online payments and control access in confidential facilities. However, mainstream face authentication systems heavily rely on deep-learning algorithms and have exhibited inherent vulnerabilities to adversarial attacks [9, 12, 43], e.g., adversaries can spoof the face authentication systems by wearing a carefully-designed dress-up such as a glasses [32], hats [21] or masks [45]. However, such attacks require adversaries to wear disguises during every authentication, which may raise suspicion and reduce their attack impacts.

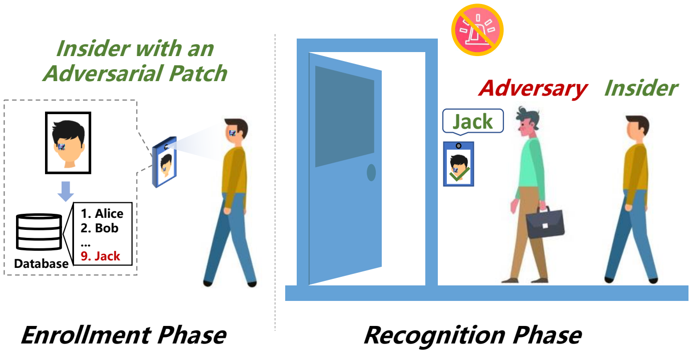


Fig. 1: UniID attack. During the enrollment phase, a legitimate insider wearing an adversarial patch registers his identity into the database to be the universal identity “Jack”. During the recognition phase, both the insider and adversaries can pass the authentication system as “Jack” without carrying any suspicious objects.

In this paper, we ask that *“Can we spoof the face authentication without any camouflage?”* Theoretically, if an insider can enroll the identities of all the given adversaries into the database, they can pass the target face authentication system without any disguise. It would grant them the ability to engage in organized criminal activities such as espionage or terrorist attacks. However, in practice, it is not feasible since an ordinary user typically lacks permission to access and modify the database. Hence, we propose the UniID attack, which injects a universal identity through the necessary but often-overlooked enrollment phase of the face authentication workflow. The basic idea of UniID is to register a universal identity with an insider wearing a carefully-crafted adversarial patch, which manipulates the face feature representation of the insider and thereby allows him to be an “average face” that incorporates the common feature of multiple adversaries. Specifically, we envision the following attack scenario as shown in Fig. 1: During the enrollment phase, an insider strategically wears an adversarial patch to register his identity in the database, effectively becoming the universal identity. In

† Yushi Cheng and Xiaoyu Ji are the corresponding authors

Network and Distributed System Security (NDSS) Symposium 2024
26 February - 1 March 2024, San Diego, CA, USA
ISBN 1-891562-93-2
https://dx.doi.org/10.14722/ndss.2024.241036
www.ndss-symposium.org

the subsequent recognition phase, the target system can not only authenticate insiders but also enable authentication of all given adversaries without requiring any kind of disguise.

Deploying UniID in scenarios involving multiple adversaries against black-box commercial face authentication systems poses challenges since UniID must ensure that multiple adversaries can successfully pass face authentication simultaneously with high confidence scores. Therefore, the key to successfully launching UniID attack is to answer the following questions: *a*) How to ensure its effectiveness with the insider and multiple adversaries simultaneously? *b*) How to make it available under the black-box setting? *c*) How to increase the physical robustness so that it can be deployed in the real world?

To overcome the above challenges, we propose a series of algorithms to optimize the adversarial patch. First, to maximize the number of enabled adversaries, we propose an attacker selection method based on feature engineering to carefully choose the appropriate adversaries. Next, we use the multi-target joint-optimization approach to generate an adversarial patch that enables both insiders and multiple adversaries to successfully bypass the face authentication system simultaneously. To address the black-box setting, we use agent model balance and connect dropout schemes to enhance the transferability of the adversarial patch. Finally, we implement color shift calibration and utilize Expectation over Transformation (EoT) to enhance the physical robustness of UniID, ensuring its reliability under real world conditions.

To validate our attacks, we conduct both simulation and real-world evaluations with six face recognition models, i.e., FaceNet, Mobile-FaceNet, ArcFace-18/50, MagFace-18/50, and two commercial face authentication systems, i.e., ArcSoft and Face++. In summary, our contributions include the points below:

- We identify the vulnerability in the face authentication system that enables multiple adversaries to be successfully authenticated by registering a universal identity in the enrollment phase.
- We design UniID that can build an adversarial patch with the common feature of adversaries, making the legitimate user register a universal identity into the database, thus achieving a stealthily spoofing attack against face authentication systems.
- We validate UniID on six face recognition models (FaceNet, Mobile-FaceNet, ArcFace-18/50, and MagFace-18/50), and two commercial face authentication systems (ArcSoft and Face++) in the laboratory setup. We achieve a max attack success rate of 100% in 3-user scenarios (1 insider and 2 adversaries) under the white-box setting and can extend to more than 8-user scenarios. We can also achieve a max attack success rate of 79% in 3-user scenarios when transferring to the black-box setting.

The goal of this paper is to offer a more comprehensive and systematic analysis of face authentication security, urging service providers to focus on security issues across all phases of the workflow to make face authentication systems more secure. To enhance the security of existing systems, we recommend three defense methods: *a*) Enhancing the ability of face authentication systems to distinguish different identities. *b*) Detecting adversarial examples at both the enrollment and recognition phases. *c*) Using assembled models to increase the attack difficulty.

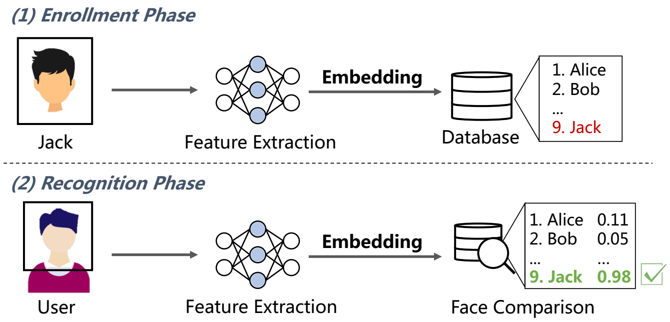


Fig. 2: The workflow of a face authentication system consists two phase: (1) Enrollment phase, and (2) Recognition phase.

II. Background

In this section, we first present the workflow and face extraction scheme of the face authentication system, and then introduce the adversarial attacks against the face authentication system.

A. Face authentication system

Face authentication is a widely-used biometric method for verifying the identity of a user. Typically, a standard face authentication system involves two phases: (1) The enrollment phase for registering a new ID, and (2) the recognition phase for verifying the identity, as shown in Fig. 2.

Enrollment Phase. A face authentication system requires a new user to enroll to be a legitimate user. The enrollment phase usually includes first requiring a face image from the user, then extracting the embedding feature vector of the provided image using the feature extraction model, and finally storing it in the database as the User ID.

Recognition Phase. When a user requests identity verification, the face authentication system first captures the user's face image and then determines whether the user is legitimate by rejecting strangers whose identity is not enrolled in the database. If the user is regarded as a legitimate one, the system will further identify the user as the one with the highest similarity score in the database.

For instance, when visitor A is trying to verify his identity, the face authentication system first extracts his embedding feature vector $f(A)$ with the feature extraction model $f(\cdot)$. Then, the feature vector $f(A)$ will be compared with each legitimate user's ID by calculating the similarity score $Simi(A, X_i)$, where X_i represents the ith user in the database. If all similarity scores are below the threshold θ, visitor A will be rejected as illegal access. Otherwise, visitor A will be successfully accessed as the legitimate user in the database with the highest similarity score.

B. Face feature extraction

Compared to image classification, face authentication presents an open-set task as the user's face images are seldom represented in the training dataset. Consequently, the face authentication system cannot directly determine the identity categories of input face images. Instead, it relies on feature comparison to distinguish between different identities, which renders feature extraction the most critical aspect of the process. Currently, convolutional neural networks (CNNs) are typically used for feature extraction, mapping the input face image into a high-dimensional feature vector. To accurately capture facial features, the feature extraction models must be trained on a large-scale face image dataset before deployment in the face authentication system.

In this context, we introduce two stages of the face extraction model: training and deployment.

Training: A face feature extraction model is designed to represent a face image in a high-dimensional feature space where the feature vectors of the same identities are clustered, and those of different identities are separated. To achieve this, the model is trained as a closed-set classification task on vast datasets like VGGFace2 [6], MS1M-ArcFace [10], among others, to learn the distribution of face images of different identities. Specifically, through loss functions such as TripletLoss [31], ArcFaceLoss [10], etc., the model learns to minimize the inter-class distance while keeping the features of different classes separable. After training, the model can map the input face image into a specific region in the high-dimensional space where the face images of the same identity are close by.

Deployment: When deployed in a face authentication system, the feature extraction model serves as a feature extractor rather than a classifier. Therefore, the fully connected layer located at the end of the CNN model is removed, and the output of the last layer is used as a high-dimensional feature vector. By comparing the distances between the feature vectors of different input face images, the face authentication system distinguishes various identities.

C. Adversarial attacks against face recognition

The face authentication system relies on feature vectors to recognize identities, which makes it susceptible to spoofing through the forging of feature vectors of legitimate users. Since the face feature vectors are typically extracted by CNN models, which have been shown to be vulnerable to adversarial attacks. By adding small perturbations, adversarial examples can spoof face recognition by identifying illegal attackers as legitimate users as follows:

$$\begin{cases} f(x) = c_{gt} \\ f(x+\delta) = c_t \neq c_{gt} \end{cases} \tag{1}$$

where $f(\cdot)$ is the face feature extraction model, c_{gt} is the correct identity of the input image x, c_t is the target identity that the adversary attempts to be, and $x+\delta$ is the adversarial example. Moreover, the perturbation δ can be calculated by the model, the input image, and the target identity as follows:

$$\arg\max_{\delta} \ J(f(x+\delta), c_t) \tag{2}$$

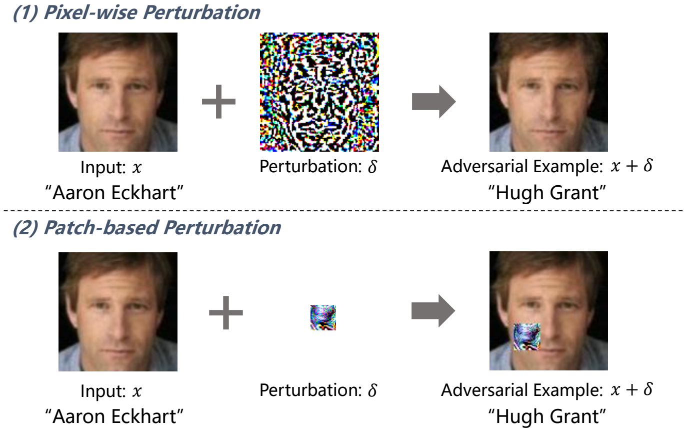


Fig. 3: Two types of adversarial perturbations: (1) Pixel-wised perturbation, (2) Patch-based perturbation.

where $J(\cdot,\cdot)$ is the loss function for solving δ, which can be resorted to numerous gradient-based approaches [7, 14] or evolutionary optimization methods [15].

In fact, as shown in Fig 3, there are two types of perturbations: ① **Pixel-wised perturbation**, a.k.a., global adversarial perturbation, which refers to the perturbation added over all pixels of the entire image. It is widely used in the digital domain [9, 12] since it can achieve a good attack performance while slightly changing the pixel values. However, this kind of perturbation may not be practical in the physical domain since adversaries cannot manipulate every pixel in the real world, and moreover, the perturbations are easily affected by environmental noises. ② **Patch-based perturbation** refers to the perturbation added in a specific area of the input image. Such perturbations are not constrained by the limited pixel value changes and appear in the form of patches, and thus are easier to deploy in the real world. In addition, with the help of transformation algorithms such as Expectation Over Transformation (EOT) [4], the patch-based perturbations can resist environmental noise and enhance their robustness in practice. Moreover, for the stealthiness of the patches, adversaries usually utilize real-existing objects as the carries for adversarial patches, such as Adv-hat [21], Adv-glasses [32], Adv-makeup [42], etc.

However, these adversarial attacks usually focus on the recognition phase and require adversary-dependent disguises every time when an adversary tries to authenticate, which could increase the likelihood of being detected by security guards.

In this paper, we aim to investigate a spoofing attack that injects a universal identity into the database during the often-overlooked enrollment phase, enabling multiple adversaries to be successfully authenticated for unlimited times.

III. Threat Model

In this paper, we investigate the feasibility of granting several adversaries legitimate identities by enrolling an insider with a universal identity. Specifically, we consider the following attack scenario: An insider tries to allow several adversaries to bypass the face authentication systems for malicious purposes such as stealing confidential information.

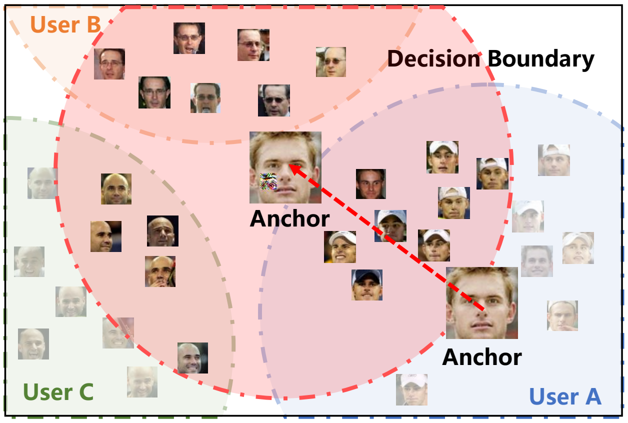


Fig. 4: Face presentation in the high-dimensional space. We manipulate the position of the anchor to launch our attack.

However, only the insider has permission to enroll himself as a legitimate user. To achieve the attack goal, the insider may try to disguise himself by such as wearing an adversarial patch to enroll a face with biometric information of multiple people, i.e., the universal identity. After enrollment, the insider can still be correctly identified in the daily routine (without carrying the adversarial patch) and more than one adversary can be identified as the identity of the insider by the target face authentication system.

To achieve the aforementioned attack, we assume the insider and adversaries are cooperators. The capability of the attackers includes providing face images of themselves to the insider. The capabilities of the insider are as follow:

Model Knowledge. We assume the insider may have either white-box or black-box access to the face authentication system. In the white-box setting, the adversary has full knowledge of the target face authentication system, including the architecture of the recognition model and its gradient information. In the black-box setting, the adversary has no knowledge of the target face authentication system and can only get the decision made by the recognition model.

Enroll Capability. We assume the insider can enroll an identity through (1) uploading photos via web or smartphone apps, or (2) taking photos on the spot. For the first case, we assume the insider can upload a face image with disguises and successfully enroll since to the best of our knowledge, the uploaded photo will not be examined by machines or humans yet. For instance, access control devices such as HikVision [17] and Invixium [20] allow users to upload face photos to complete the enrollment process. For the second case, we assume the insider can wear a disguise and complete enrollment on the spot without raising suspicions, e.g., when security guards are not present or not paying attention, as discussed in the user study in Sec. VII.

IV. Preliminary Analysis

The key to fulfilling the attack goal is to allow the insider wearing a disguise can be recognized as all adversaries by face authentication systems. To achieve this, we investigate the possibility of constructing an "average face" of adversaries at the feature level using adversarial attacks.

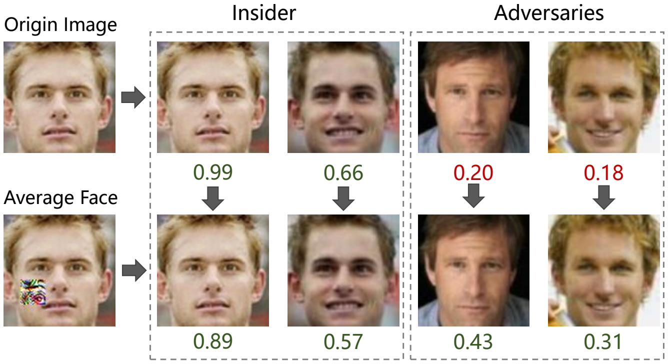


Fig. 5: An illustration of preliminary analysis results. The images from left to right are: the enrolled face images, the images of the insider and the images of the adversaries. The numbers under the images are the confidence scores.

TABLE I: Attack Success Rates under Different Number of Attackers

Number of Attackers	Attack Success Rate
1	90.0% (9/10)
2	22.2% (10/45)
3	3.3% (4/120)
4-10	0% (0/848)

A. Basic Idea

As described in Sec. II, the face authentication systems determine the identity by comparing the distance between feature vectors. Therefore, the "average face" at the feature level refers to the feature vector that is positioned in close proximity to all adversaries within the high-dimensional feature space. To achieve this, our basic idea is to manipulate the position of the insider's feature vector towards the central region of the adversaries' feature vectors through adversarial attacks, as shown in Fig. 4. Since the face authentication system utilizes a fixed threshold to establish the decision boundary radius for determining the legitimacy of access, both the insider and adversaries can be identified as the insider as long as the decision boundary encompasses the feature vectors of the adversaries.

B. Preliminary Results

In this paper, we employ a patch-based adversarial attack to generate the "average face" due to its portability and ease of deployment in real-world scenarios. To investigate its feasibility, we conduct experiments under a white-box setting against a commonly used face recognition model, ArcFace-18. [10]. Specifically, we fist randomly select one insider and ten adversaries from the face dataset LFW [19]. Then, we optimize the adversarial patch by a gradient-based optimization algorithm Fast Gradient Sign Attack (FGSM) [14]. During the experiments, we permute the ten adversaries to form 1,023 different combinations, ranging from one to ten adversaries. For each combination, we optimize an adversarial patch with a fixed size 25×25 pixels and place it on the right cheek of the insider to create the "average face". To validate the effectiveness of each "average face", we calculated the similarity score

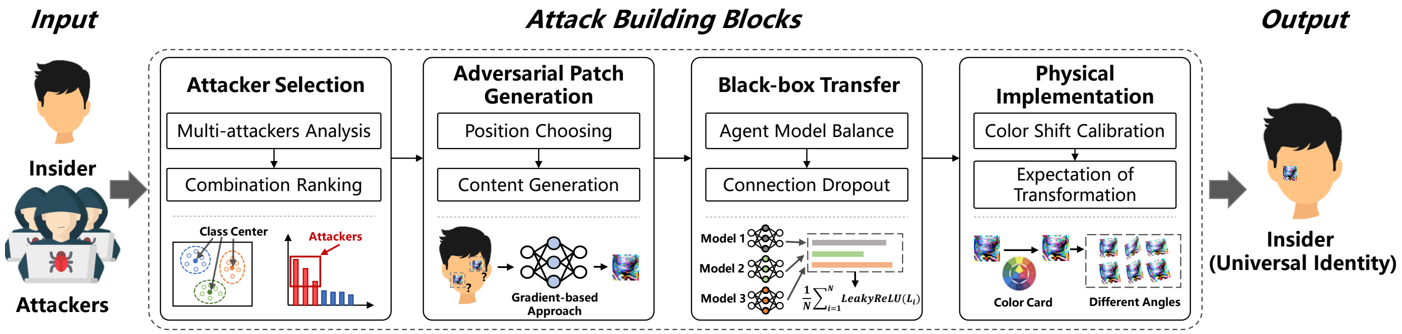


Fig. 6: Overview of UniID: The insider first selects a proper attacker combination from the candidates. Then, he generates the adversarial patch using the gradient-based optimization method, enhances the transferability of the adversarial patch to make it effective under black-box systems, and finally calibrates the color and shape shifts to make the patch practical in the real world.

between the "average face", the insider and the adversaries.

From the illustration shown in the Fig. 5, we find: (1) For the legitimacy of the insider, we compare the similarity scores between the "average face" and the various images of the insider. As a result, the similarity scores only decrease by ~ 0.1, indicating the adversarial patch has minimal impact on the identification of the insider. Since the similarity score is much higher than the default threshold (0.24), the insider can pass the face recognition model in his daily routine without any restrictions. (2) For the multiple adversaries, compared to the similarity scores with the origin image of the insider, their similarity scores with the "average face" all surpass the default threshold, which indicates that our notion of multiple adversaries simultaneously using one identity (i.e., the universal identity) to bypass face authentication is feasible.

However, as illustrated in Tab. I, the attack success rate declines sharply with an increase in the number of adversaries. This decline can be attributed to the random selection of adversaries, which may result in interactions between multiple adversaries and make it challenging for adversarial patches to converge during the optimization process.

Based on these observations, two challenges remain to be addressed: ① When dealing with a specific insider, it is crucial to carefully choose the appropriate adversaries to ensure the successful execution of our attack. ② Although our attack has been demonstrated to be feasible in the digital world, further efforts are required to maintain its effectiveness in real-world scenarios.

V. Attack Design

To achieve the attack goal, it is important to answer the following questions:

- **Q 1:** How to select the appropriate adversaries to ensure the effectiveness of our attack?
- **Q 2:** How to enhance the transferability of adversarial patches to address the black-box setting?
- **Q 3:** How to increase the physical robustness of adversarial patches such that they can be deployed in real life?

To address these questions, our attack incorporates three modules, as shown in Fig. 6. The **Attacker Selection** module chooses appropriate adversaries by feature engineering to enable a maximum number of adversaries. The **Adversarial Patch Generation** module finds the most suitable location to place the adversarial patch and generates its content that enables multiple attackers and the insider to pass the face authentication simultaneously. The **Black-box Transfer** module enhances the transferability of the adversarial patch using agent model balance and connect dropout to address the black-box setting. The **Physical Implementation** module calibrates the color shift caused by the printing-capturing process and uses the EoT strategy to improve its robustness in the physical world. In the following sections, we present these attack building blocks in detail.

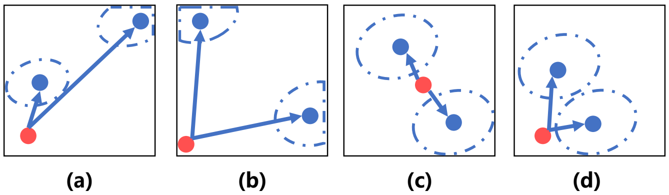


Fig. 7: Cases for different attackers: (a) One attacker is close to the insider while the others are far away. (b) Both attackers are far away from the insider. (c) Both attackers are close to the insider but stand on opposite sides. (d) Both attackers are close to the insider and situated together.

A. Attacker Selection

Based on the preliminary analysis, we find that given a specific insider, different adversaries may exhibit different attack success rates. With appropriate adversary candidates, we can maximize the number of enabled adversaries. In the following, we propose a ranking method to select the most appropriate combination of attackers.

1) Multi-attackers Analysis: To explore the potential impact of different attacker combinations, we analyze the distribution of face-embedding features in high-dimensional space. We identify two scenarios where generating adversarial patches may be difficult or even impossible: ① When the attackers are too far away from the insider: Our attack aims to manipulate the insider's anchor to the center of the attackers by adding an adversarial patch. However, if the distance between the optimization target (attacker) and the initial sample (insider) is too large, the anchor may fail to optimize to a point where its decision boundary can cover both of them, as shown in Fig. 7 (a), (b). Additionally, for portability reasons, the adversarial patches must be limited in size and wearing position, which further restricts the distance between the optimized anchor and the original one of the insider, making it difficult to cover attackers that are too far away. ② When the attackers are located on either side of the insider: In this scenario, as shown in Fig. 7 (c), the optimization process of an adversarial attack oscillates between different attackers and eventually fails to generate an effective adversarial patch.

To avoid the aforementioned issues, an optimal combination must ensure that the attackers and the insider are as close to each other as possible, while the attackers themselves are tightly clustered, as shown in Fig. 7 (d). In light of this, we propose two metrics to determine whether an attacker combination is appropriate. For ①, we use the similarity metric which is the average cosine similarity to indicate the distance between an attacker combination and the insider. Specifically, if we select a N-attackers combination $\mathbb{A} : \{A_1, ..., A_n\}$, the distance shall be as follows:

$$Sim(V, \mathbb{A}) = \frac{1}{N} \sum_{i=1}^{N} \frac{f(V) \cdot f(A_i)}{|f(V)| \times |f(A_i)|} \tag{3}$$

where the N is the number of attackers, $f(V)$ is the embedding feature of insider, and $f(A_i)$ is the embedding feature of i th attacker. For ②, we utilize the average cosine similarity between attackers as an aggregation metric to indicate whether the attackers are clustered together as follows:

$$Agg(\mathbb{A}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{f(A_i) \cdot f(A_j)}{|f(A_i)| \times |f(A_j)|} \tag{4}$$

2) Combination Choosing: In the attack scenario defined in Sec.III, we can select the most appropriate N-attackers from a set of attacker candidates to launch our attack. In general, we can get the appropriate combination by maximizing $Sim(V, \mathbb{A}) + Agg(\mathbb{A})$. However, we have observed that the combinations of attackers selected from different face recognition models can vary significantly. A combination selected from one model may not be feasible in other models. This is because different face recognition models typically employ various backbones and are trained from different datasets, leading to inconsistencies in feature representation.

To address this challenge, we propose assembling multiple models to select the appropriate combination, as follows:

$$\arg\max_{\mathbb{A}} \{\frac{1}{M} \sum_{m=1}^{M} (Sim_m(V, \mathbb{A}) + Agg_m(\mathbb{A}))\} \tag{5}$$

where M is the number of assembled models, $Sim_m(V, \mathbb{A})$ and $Agg_m(\mathbb{A})$ are the similarity metric and aggregation metric of m-th face recognition models. In this paper, we evaluate a variety of face recognition models with backbones consisting of VGG-16 [34], MobileNet [18], IRSE-18, and IRSE-50 [10], which are trained on the CASIA [41] and MS1MV2 [10] datasets. We find that a combination selected from such diverse models can successfully attack multiple models simultaneously and can even transfer to the black-box setting.

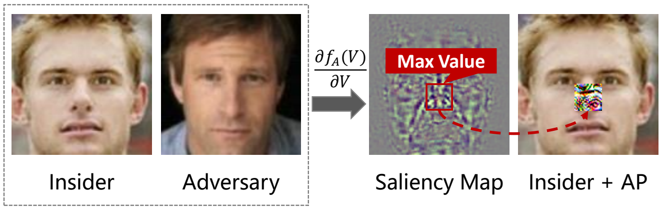


Fig. 8: Patch position choosing. The location is determined by applying the coordinate of maximum value in the saliency map from the insider and adversaries.

B. Adversarial Patch Generation

Our attack relies on the use of an adversarial patch, which is carried by the insider to enroll his identity. In this subsection, we determine the most appropriate position for placing the adversarial patch, striking a balance between attack effectiveness and the convenience of deployment. Then, we generate the contents of the adversarial patch through a gradient-based optimization algorithm in a white-box setting.

1) Position Choosing: In a physical adversarial attack, it is crucial to balance patch portability and attack effectiveness. Typically, the eye region plays a significant role in distinguishing between different identities. However, adding patches to this region may cause the insider to fail the face detection and liveness detection phases during enrollment. Conversely, adding a patch to the chin or cheek regions has limited impact on detection but results in reduced attack effectiveness.

To address this issue, we construct a saliency map based on the forward derivative [26], and utilize a mask to avoid covering the eye region. The saliency map identifies the contribution of each pixel when it is classified as an attacker. As shown in Fig. 8, we determine where to add the patch by maximizing the saliency map, using the following formula:

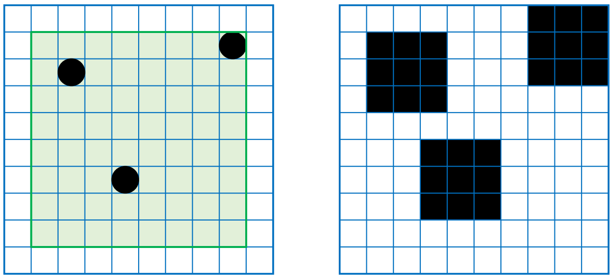
$$\mathbf{S}(V, \mathbb{A})[p, q] = \sum_{A \in \mathbb{A}} \frac{\partial f_A(V)}{\partial V_{p,q}} \times Mask \tag{6}$$

where (p, q) is the coordinate of a pixel in the image of the insider, $f_A(V)$ denotes the similarity score when the insider V is classified as an attacker A, and the $Mask$ is a binary image where the eye region is set to 0 and others are set to 1.

2) Content Generation: Once we have determined the most appropriate position, our next step is to generate an adversarial patch that can be used to carry out the attack. To achieve this, we utilize a gradient-based optimization method MI-FGSM [11]. Additionally, we have formulated a loss function that considers two significant factors: (1) multi-target similarity loss that aims to enhance the similarity between the attackers' faces and the perturbed face in scenarios where multiple attackers are involved. (2) content smooth loss that minimizes the noise effect resulting from the printing-capturing process. We strive to make the pixels of the optimized patch as smooth as possible, thereby reducing the impact of any distortions or anomalies that may arise during the generation of the adversarial patch. In short, the loss function is as follows:

$$\mathcal{L} = \alpha \mathcal{L}_{sim} + \beta \mathcal{L}_{tv} \tag{7}$$

where α and β are the weights of the multi-target similarity loss and the content smooth loss, respectively.



(a)

(b)

Multi-target Similarity Loss. In contrast to other adversarial attacks, our approach involves the collaboration of multiple attackers working in tandem. As explained earlier, our aim is to shift the insider's anchor by introducing an adversarial patch that repositions it at the center of the attackers. To achieve this, we first compute the average embedding features of the attackers and use this as the central point in the high-dimensional space where the attack is executed:

$$f(A)_{avg} = \frac{1}{N} \sum_{n=1}^{N} f(A_n) \tag{8}$$

where $f(\cdot)$ is the target model, A_n is the n-th attacker, N is the number of attackers. Then, we make the target model to identify the attackers as the insider. Thus, the similarity loss is as follows:

$$\mathcal{L}_{sim} = 1 - \frac{f(V,\delta) \cdot f(A)_{avg}}{|f(V,\delta)| \times f(A)_{avg}|} \tag{9}$$

where V is the insider and δ is the adversarial perturbation.

Content Smooth Loss. Since our attack is intended for deployment in the physical world, the adversarial patch will need to be printed and captured by a camera. However, if the internal pixel values of the patch are not continuous, variations in the patch's color distribution may arise during the printing-and-capturing process, which could reduce the effectiveness of the attack in the real world. To overcome this challenge, we utilize a total variation (TV) loss function to smooth out the color distribution of the patch pixels as follows:

$$\mathcal{L}_{tv} = \sum_{i,j} \sqrt{(x_{i,j} - x_{i+1,j})^2 + (x_{i,j} - x_{x,j+1})^2} \tag{10}$$

where $x_{i,j}$ is the pixel value of the adversarial patch x at the coordinate (i,j).

C. Black-box Transfer

Face authentication systems used in real-world scenarios often operate as black-box systems. As a result, we try to improve the transferability of our attack to make it effective against various unseen models. To accomplish this, we optimize the adversarial patch using a mount of agent models with random connection dropout.

1) Agent Model Balance: A straightforward black-box transfer approach is to calculate the average similarity of outputs from multiple models as the optimization objective. However, different models may have varying gradients, and simply averaging their outputs can cause over-optimization of models with larger absolute gradient values while neglecting those with smaller ones. This can potentially affect the effectiveness of the attack. To balance the relative relationship between the gradients of each model, we use the Leaky ReLU function. By doing so, we make the contribution of each agent model closer in the process of patch generation. The Leaky ReLU function is based on a ReLU and is commonly used as an activation function in model training [25]. It is defined as follows:

$$\mathrm{LeakyReLU}(x) = \begin{cases} x, \mathrm{IF}\ x \geq 0 \\ slope * x, \mathrm{IF}\ x < 0 \end{cases} \tag{11}$$

where *slope* is the angle of the negative slope. The balanced loss can be obtained from the following function:

$$\mathcal{L}_{balance} = \frac{1}{M} \sum_{m}^{M} Leaky\mathrm{ReLU}\left(L_{sim}^{m} - \varepsilon\right) \tag{12}$$

where L_{sim}^{m} denote the similarity loss of the m-th agent model, respectively, and the ε denote the balance threshold which is set as 0.4 in this paper. Based on this, the loss function now becomes as follows:

$$\mathcal{L} = \alpha \mathcal{L}_{balance} + \beta \mathcal{L}_{tv} \tag{13}$$

where α and β are the weights of the agent model balance loss and the content smooth loss, respectively.

2) Connection Dropout: Dropout is an essential technique for effectively training deep neural networks. By randomly dropping units and their connections, this technique helps to prevent overfitting, which is a common problem in complex models. While dropout is typically applied at the fully connected layer, it can have a significant impact on overall training performance.

To generate the adversarial patch in our application, we employ dropout between convolutional layers. This approach allows us to increase the potential configurations of sub-networks and combine them iteratively, resulting in a more effective overall approach [44]. However, due to the correlation between close units in convolutional layers, information from a dropped unit can still be passed to the next layer through adjacent units. To address this issue and further optimize dropout, we have devised square dropout, which drops units in a square region of a feature map together. This approach ensures that all units within the same region are dropped, thereby preventing correlated information from being passed to the next layer and making dropout even more effective in preventing overfitting.

In particular, a face model comprises multiple convolutional layers and an excessive amount of dropout can hinder the recognition capabilities of the model. Therefore, we have opted to incorporate square dropout solely between blocks, as depicted in Fig. **??**. For a face model, given the feature V_i from the i-th block, we randomly generate a mask M_i as follows:

$$\begin{aligned} Points_i &= \text{Random\_Sample}\left(p/size^2, W, H\right) \\ M_i &= \text{Square\_Apply}\left(Points_i, size, W, H\right) \end{aligned} \tag{14}$$

where p is the possibility of dropout, $size$ is the side length of the dropped square region, W and H are the width and height of V_i, Random_Sample$(\cdot)$ denotes generating scattered points as the center of square dropout, and Square_Apply$(\cdot)$

represents generating square mask centered on the points. Then, we apply the mask and normalize the features as follows:

$$V_i' = V_i \times M_i \times \text{Count}(M_i)/\text{Count\_Ones}(M_i) \tag{15}$$

where V_i' is the dropped feature, and $\text{Count}(\cdot)$ and $\text{Count\_Ones}(\cdot)$ calculate the size of the input vector M_i and the number of 1 in it, respectively.

D. Physical Implementation

To make our attack practical in the real world, we try to improve the robustness of the generated adversarial patch in this subsection. Specifically, to mitigate the negative impact of color shift and shape deformation caused by the printing-capturing process, we have adopted two methods: (1) We have developed an MLP model that maps the color transformation from the digital world to the physical world, which is used for pre-calibrating the color shift. (2) We have incorporated the Expectation of Transformation (EoT) technique to enhance the patch's resilience against deformation.

1) Color Shift Calibration: When executing a physical adversarial attack, the color distortion caused by the printing-capturing process can potentially lessen the effectiveness of our attack. Since the color transformation from the digital world to the physical world is a non-linear function, we have employed a Multi-Layer Perceptron (MLP) model to simulate it. Prior to training the model, we extracted 4096 colors (each consisting of 16 shades of red, green, and blue) to construct a color board denoted as B_d. This board represents most of the colors in RGB mode. We then captured the printed color board using a camera, which we refer to as B_p. By combining these two color boards, we created a training sample $\{B_d, B_p\}$ to train the MLP model, allowing it to learn the color transformation function.

During each step of the adversarial patch optimization, we utilize the MLP model to transform the color space. Once the patch has been fully optimized, we print the pre-calibrated patch to initiate our attack. This helps eliminate any potential color shift during the printing-capturing process.

2) Expectation of Transformation: In addition to the color shift issue, the user-camera distance and angle can also result in shape deformation of the adversarial patch, ultimately leading to a decrease in attack performance. To tackle this problem, we utilize the Expectation of Transformation (EoT) technique to enhance the shape robustness of the adversarial patch in the physical world during the optimization phase. The EoT method involves constructing a transform distribution T that encompasses position, scaling, rotation, affine, and brightness adjustments. Then, by randomly choosing a transformation function $t \in T$ at each step of the optimization process, the EoT can convert the origin adversarial patch δ into a *"physical"* patch δ^* as follows:

$$\delta^* = \arg\min_{\delta} \mathbb{E}_{t\sim T}[\mathcal{L}[f(V, \mathbb{A}, t(\delta)), f(\mathbb{A})]] \tag{16}$$

where t is the transformation function that is randomly sampled in contribution T. δ^* is the transformed patch and δ is the original patch. V and $\mathbb{A}$ are the insider and attackers, respectively.

TABLE II: The Backbone Networks and Training Datasets of Target Models

Target Models	Backbone	Training Dataset	Threshold
FaceNet	VGG-16	VGGFace	0.409
Mobile-FaceNet	MobileNetV2	MS-Celeb-1M	0.305
ArcFace-18	IRSE-18	CASIA	0.24
ArcFace-50	IRSE-50	MS1MV2	0.24
MagFace-18	IRSE-18	CASIA	0.24
MagFace-50	IRSE-50	MS1MV2	0.24

VI. Evaluation

In this section, we evaluate our attack in two aspects: (1) simulated attack evaluation, where the adversarial patch is generated against face recognition models with public face datasets, and (2) real-world evaluation, where the adversarial patch is deployed in the physical world to attack the commercial face authentication systems/devices. We use the attack success rate (ASR) as the metric to evaluate our attack both in simulation evaluation and real-world evaluation.

A. Simulation Evaluation

1) Experimental Setup: For the simulation evaluation, we generate adversarial patches in the white-box setting. The models and datasets used as targets for this evaluation are as follows:

Target models. We evaluate our attack using six face recognition models, i.e., FaceNet, Mobile-FaceNet, ArcFace-18/50 and MagFace-18/50. The former two are the representative face recognition model while the last four are the state-of-the-art models. The backbone networks, training datasets and default threshold of the models are list in the Tab. II.

Datasets. We use two face datasets LFW [19] and CelebA [24] in the simulation evaluation. LFW contains 13,233 face images for 5,749 identities, and CelebA contains 202,599 face images for 10,177 identities. For both datasets, we randomly choose 100 identities as the insiders and select the attacker combinations using the proposed attacker selecting method.

Attack Metric. We define our attack to be successful when all attackers in the combination $\mathbb{A}$ can be recognized as the insider as follows:

$$AS = \mathbb{1}\{\sum_{A\in\mathbb{A}} \mathbb{1}_\theta(cos[f(V+\delta^*), f(A)] > \theta) = L(\mathbb{A})\} \tag{17}$$

where the $\mathbb{1}$ is the indicator function, $L(\mathbb{A})$ is the attacker number of combination $\mathbb{A}$, and θ is the threshold of the target model. Based on this, we use the attack success rate (ASR) as the metric in the simulation evaluation, which is the ratio of the number of successful attacks over the total number of conducted attacks. Thus, the metric can be formulated as:

$$ASR = \frac{1}{N} \sum_{n=1}^{N} AS_n \times 100\% \tag{18}$$

where the AS_n is the n-th of the conducted attack. Specifically, each attack we denote is an attack launched against a specific insider and its corresponding attacker combination.

TABLE III: Overall Performance of `UniID` with Two Datasets Against Six Different Face Recognition Models.

Target Models	Datasets	Number of Attackers									
		1	2	3	4	5	6	7	8	9	10
FaceNet	LFW	99%	97%	88%	74%	61%	43%	32%	23%	13%	11%
	CelebA	99%	86%	73%	53%	38%	26%	15%	16%	14%	10%
Mobile-FaceNet	LFW	100%	85%	56%	31%	15%	5%	2%	1%	0%	0%
	CelebA	97%	75%	58%	47%	28%	14%	14%	8%	4%	4%
ArcFace-18	LFW	100%	100%	93%	84%	66%	50%	36%	22%	17%	11%
	CelebA	100%	99%	90%	82%	73%	65%	56%	50%	50%	36%
ArcFace-50	LFW	99%	85%	57%	13%	5%	2%	0%	0%	0%	0%
	CelebA	99%	82%	48%	28%	15%	2%	1%	1%	0%	0%
MagFace-18	LFW	100%	100%	93%	86%	69%	51%	42%	33%	18%	12%
	CelebA	100%	99%	92%	88%	79%	74%	64%	63%	54%	52%
MagFace-50	LFW	100%	85%	50%	16%	6%	2%	1%	0%	0%	0%
	CelebA	97%	77%	35%	17%	10%	6%	3%	0%	0%	0%

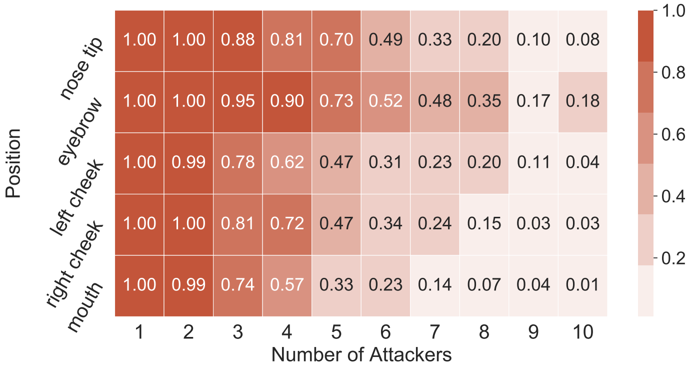


Fig. 9: Impact of different patch position.

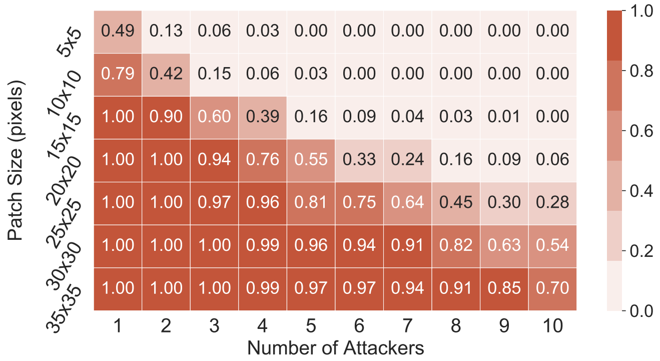


Fig. 10: Impact of different patch size.

2) Overall Performance: In this subsection, we first evaluate the effectiveness of our attack on different target models under their default thresholds in the white-box setting. We select 1 to 10 attackers for each of the 100 insiders in the LFW and CelebA datasets to launch our attack and calculate the attack success rates. To ensure consistency, during the experiments, the patch size is constant in 25 × 25 pixels and the patch position is located in the eyebrow region. The results shown in Tab. III demonstrate that our attack can achieve an overall average success rate of 99.1%, 87.1%, 69.4%, 51.6%, 38.8%, 28.3%, 22.2%, 18.1%, 14.2%, and 11.3% for 1 to 10 attackers, respectively. Among the target models, we find the ArcFace-18 and MagFace-18 are more vulnerable than other models because they use a simple backbone structure IRSE-18 and are trained by a small-scale dataset CASIA that only contains 49,4414 face images for 10,575 identities. The above two reasons limit the models' ability to distinguish different faces and lead them easier to be attacked. For the testing datasets, we find that the attack performance of CelebA is better than LFW. The reason is that CelebA has more identities, which makes it easier to select suitable attacker candidates. Additionally, to evaluate the generalization capabilities across diverse face images, we conduct the cross-dataset evaluation between LFW and CelebA, and the results demonstrate that our attack performs similarly across different datasets as it does on a single dataset. All the detials about the experiments can be found in Appendix A.

Overall, our attack is feasible to be successfully launched under 6 attackers for all six models and two datasets and can achieve to be conducted under more than 10 attackers with certain models.

3) Impact of Patch Details: In this section, we explore various factors that could affect the effectiveness of our attack. Specifically, we investigate the impact of patch size and position on the adversarial effects of our attack. To streamline our analysis, we focus on the MagFace-18 model and measure the ASR using the LFW dataset. To ensure a fair evaluation, we maintain a constant patch size of 25 × 25 pixels when assessing the effect of patch position. Similarly, we restrict the patch position to the eyebrow region when evaluating the impact of patch size. This helps us avoid any mutual interference between these two variables and obtain reliable results.

Impact of Patch Position. To investigate how patch positions can impact attack success rates, we conduct experiments by placing the patch on five distinct facial regions: the eyebrow, nose, left cheek, right cheek, and mouth. Our results, as depicted in Fig. 9, indicate that all regions can achieve similar ASR >90% when the number of attackers is less than 4.

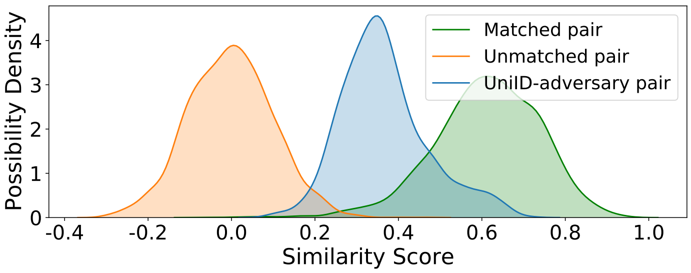


Fig. 11: The distribution of similarity scores on matched pairs, unmatched pairs and `UniID`-adversary pairs.

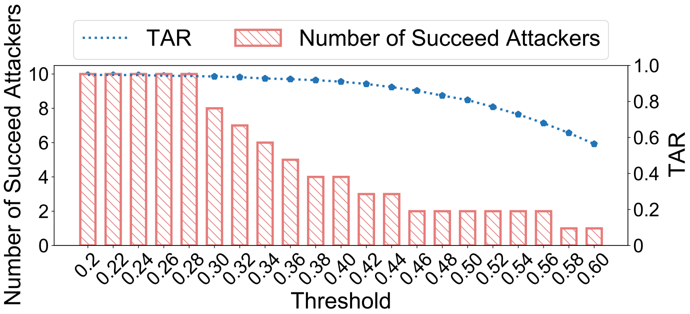


Fig. 12: Impact of different thresholds.

However, as the number of attackers increases, performance significantly drops first in the mouth region and then in the cheek region. This is because these areas are less crucial for face feature representation, and modifying their pixel values does not lead to the anchor feature shifting into the center of a significant number of attackers, which is typically located far away from the origin anchor. In contrast, the eyebrow region and nose tip region show better performance because they are important for distinguishing between different identities in face recognition models, and slight pixel value changes in these regions can result in a significant feature shift in high-dimensional space. However, since the nose tip region typically protrudes from the face, it may not be practical to wear an adversarial patch in this location.

Thus, considering both attack performance and patch portability, the eyebrow region emerges as the most suitable area for launching our attack among the five regions tested.

Impact of Patch Size. The size of the patch is a critical factor that determines attackers' ability to manipulate pixels and directly affects the success rate of our attack. In order to investigate its impact, we conducted experiments with patch sizes ranging from 5 × 5 pixels (0.2% of the face image) to 35 × 35 pixels (9.7% of the face image).

Our results, as shown in Fig. 10, indicate that an excessively small patch size cannot successfully launch an attack. When the patch size is larger than 20 × 20 pixels, our attack can be successfully launched even with more than 10 attackers. The reason for this is that a 5 × 5 patch only covers 0.2% of the face image, such minor changes in the face image have little effect on feature representation. As the patch size increases, we can manipulate more pixels, which gives the patch the ability to significantly influence the feature representation.

4) Impact on Different Thresholds: Face authentication systems use thresholds to determine whether a visitor is a legitimate user, and different thresholds can affect the performance of our attack. To study its impact, we initially analyze the distribution of similarity scores for matched pairs, unmatched pairs and `UniID`-adversary pairs. From the results shown in Fig. 11, we observe that the similarity scores of `UniID`-adversary pairs predominantly range between 0.2 and 0.6. Additionally, the number of unmatched pairs rises significantly when the similarity score exceeds 0.3.

Therefore, we conduct experiments by setting the thresholds from 0.2 to 0.6. During the experiments, we use the MagFace-18 model as the target for simplicity and counted the number of successful attacks under different thresholds.

From the results shown in Fig. 12, we find that the performance of our attack decreases as the threshold increases. Specifically, our attack can still be successful even if the threshold is increased to 0.6, and it can remain effective with more than 2 attackers when the threshold is ≤ 0.56. Raising the threshold makes the face authentication model more stringent when distinguishing between different identities, leading to a decrease in the performance of our attack. However, increasing thresholds may seriously weaken the generalization ability of the face authentication model, resulting in missed identification for the same person. Therefore, we calculated the True Accept Rate (TAR) of the target system, which indicates the percentage of authorized users with correct matches to legitimate users, as follows:

$$TAR = \frac{TP}{TP + FN} \tag{19}$$

where TP is the true positive samples, i.e., the number of authorized users with correct matches to legitimate users, and $TP + FN$ is the number of legitimate users.

Commercial face authentication systems typically require a TAR of over 99.9% to ensure reliable user authentication under different lighting conditions, facial angles, and makeup variations. However, as illustrated in Fig 12, when the threshold is increased, the TAR of the system decreases to 0.6. This means that nearly 40% of legitimate users are unable to authenticate themselves using their own facial image, making this unacceptable for commercial face authentication systems. Therefore, merely increasing the threshold as a defense approach is unrealistic given practical constraints.

5) Black-box Transferability: In most commercial face authentication systems, attackers are unable to gain any knowledge about their underlying models. Instead, they can only obtain the output decision. In this paper, we optimize the effectiveness of an adversarial patch using multiple agent models and evaluate its performance in black-box models that were not used in the optimization process. Specifically, we optimize the patch across 11 agent models with varying backbones and training datasets. We then select three face authentication models (ArcFace-18, MagFace-18, and MagFace-50) and two commercial face authentication API/SDK systems (ArcSoft [3] and Face++ [13]) as our black-box models for launching the attack. We calculate the ASR under various numbers of attackers and use the same default thresholds as those in Tab.II. For ArcSoft and Face++, we set the thresholds at 0.8 and 0.63, respectively, as recommended by their manufacturers.

TABLE IV: The Attack Performance in Black-box Models

Target Models	Number of Attackers		
	1	2	3
ArcFace-18	95%	79%	45%
MagFace-18	98%	71%	36%
MagFace-50	95%	62%	20%
Face++	81%	45%	20%
ArcSoft	86%	27%	12%

TABLE V: Overall Performance of `UniID` with 20 Volunteers Against Two Commercial Systems in Real World

Metric	Target Models	Number of Attackers	
		1	2
ASR	Face++	87%	41%
	ArcSoft	86%	47%
F_{succ}	Face++	84.3%	71.1%
	ArcSoft	86.5%	61.5%

Models. According to the results presented in Table IV, our attack successfully transferred to black box face recognition models, with an average Attack Success Rate (ASR) of 96%, 70.7%, and 33.7% under scenarios involving one, two, and three attackers, respectively. However, when compared to the attack performance under white box conditions, we observe a drop in ASR of nearly 4% for one attacker, 25% for two attackers, and 40% for three attackers. This drop in performance can be attributed to the varying structures and parameters of the backbones used by different models, as well as the diverse datasets they were trained on. As a result, the feature representation spaces they map are heterogeneous, which can impact the effectiveness of our attack. In particular, as the number of attackers increases, the impact of heterogeneity in feature space becomes more significant.

Commercial systems. Our attack is also able to transfer to commercial face authentication systems, as demonstrated by the results in Tab. IV. Under scenarios involving one, two, and three attackers, we achieve an average ASR of 81%, 45%, and 20% against Face++, and 86%, 27%, and 12% against ArcSoft, respectively. However, compared to the models, the transferability of our attack appear to be worse. This can be attributed to the use of more robust models and stricter thresholds by commercial systems to prevent misidentification. Nonetheless, our attack is still capable of confusing these commercial systems under such severe conditions, highlighting the potential threat that it poses to real face authentication systems.

B. Real-world Evaluation

1) Experimental Setup: During our real-world evaluations, we perform attacks in the physical world to target commercial systems. Specifically, for each insider-attacker combination, we optimize an adversarial patch with a size of 40×20 pixels from agent models. We then print the patch as a $70mm \times 35mm$ physical patch and affix it to the eyebrow and nose region of the insider in the real world. After the insider has enrolled their face into the target models or systems, attackers verify the identity without carrying any suspicious devices or wearing any makeup, as part of a normal face authentication process. During our experiments, since it is not feasible to physically validate on the datasets, we have recruited volunteers to conduct the attacks. In the following sections, we provide detailed information on the target models, volunteers, and metrics used in our evaluation.

Target model. We evaluate our attack under two commercial face authentication API/SDK ArcFace and Face++ in the real-world setting.

Volunteers. We have recruited 20 volunteers to capture their facial images for evaluating the effectiveness of our attack in real-world scenarios, including 15 males and 5 females between the ages of 20 and 30. Approval of all ethical and experimental procedures and protocols was granted by Science and Technology Ethics Committee of Zhejiang University.

Metric. In the real-world evaluation, we use two metrics: (1) The attack success rate (ASR) defined in simulation evaluation, and (2) The attack success rate in consecutive frames (F_{succ}) which indicates the attack ability against the target systems in the real-world setting.

$$F_{succ} = \frac{1}{C}\sum_{c=1}^{C} \mathbb{1}_{\theta}(cos[f(V+\delta^{*}), f(A)] > \theta) \tag{20}$$

where C is the number of consecutive frames, $\mathbb{1}$ is the indicator function, and θ is the threshold of the target model. Note that a higher F_{succ} indicates that our attack can be successfully launched more easily in the real world. In other words, a higher F_{succ} score suggests that the adversarial patch created by our agent model is more effective at evading face authentication systems, making it easier for attackers to impersonate the insider without being detected.

2) Overall Performance: To assess the overall performance of our approach, we carry out cross-validation experiments involving 20 volunteers. Specifically, we calculate the average ASR and F_{succ} under both one- and two-attacker scenarios. All face images, including those of insiders and attackers, are captured under typical indoor lighting conditions of $300lx$ illumination intensity. By evaluating performance across multiple volunteers, we can gain insights into how well our approach generalizes to different individuals. Additionally, capturing images under common lighting conditions is important to ensure that our results reflect real-world scenarios where lighting conditions could vary.

The results presented in Tab. V demonstrate the effectiveness of our attack against different commercial face authentication systems. For Face++, we achieve an average ASR of 87% and 41% under one- and two-attacker scenarios, respectively. Similarly, for ArcSoft, we attain an average ASR of 86% and 47% under one- and two-attacker scenarios, respectively. Notably, we observe that the attack performances against both target systems are similar. This suggests that our approach can transfer to different commercial face authentication systems while maintaining its effectiveness. Regarding F_{succ}, our attack achieves a success rate of over 84% and 60% under one- and two-attacker scenarios, respectively. This indicates that our adversarial patch can successfully deceive the target systems

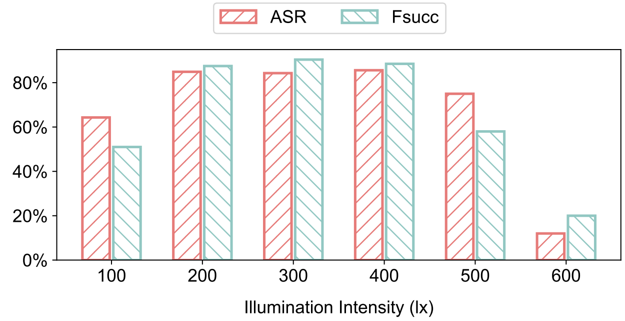


Fig. 13: The attack effectiveness under various light conditions.

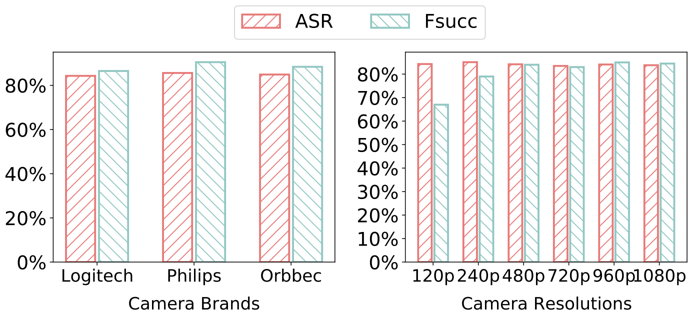


Fig. 14: The attack effectiveness under various camera models and resolutions.

within less than two frames, making it highly effective in real-world scenarios.

We also evaluate our approach under multi-attacker scenarios and find that the ASR drops to $\leq$ 2% when the number of attackers $\geq$ 3. This may be due to the weakening of the adversarial patch's effectiveness during the printing-capturing process involved in physical deployment. Nevertheless, our approach remains a significant threat to real-world face authentication systems as it can successfully work against commercial systems.

3) Impact on Light Conditions: Commercial face authentication systems can be deployed both indoors and outdoors, making attackers vulnerable to different light conditions during the authentication process. To evaluate the impact of light conditions on our attack, we conduct experiments by varying the background illumination intensity from 100 lx to 600 lx, which encompasses most light conditions in both indoor and outdoor scenarios.

Our results, as shown in Fig. 13, indicate that our attack maintains an ASR of above 80% and a F_{succ} of over 85% when the illumination intensity is between 200 lx to 400 lx lx. However, when the illumination intensity is either excessive or deficient, our attack's performance starts to decline. In particular, at 600 lx lx, the ASR drops to nearly 10%, while F_{succ} drops to nearly 20%. This happens because an excessive or deficient illumination intensity can cause the camera to capture overexposed or underexposed images, leading to a decrease in the attack's effectiveness.

Notably, the illumination intensity in most indoor scenarios is around 300 lx, while outdoor illumination intensity is generally less than 600 lx unless the camera faces direct sunlight. As such, our attack can be successfully launched in most realistic light conditions and can tolerate some variation in illumination intensity.

4) Impact on Cameras: In order to attack commercial face authentication systems, an adversary may need to contend with various camera brands and models, each with different resolution settings. This is because face authentication providers may use cameras from different vendors. To evaluate the impact of these factors on our attack effectiveness, we considered both camera models and resolutions.

To ensure that there is no mutual interference between camera models, we set all cameras to a resolution of 480p during our evaluations. When evaluating the impact of camera resolutions, we use the same camera (the Logitech model) for consistency. For simplicity, we select ArcSoft as the target system for our experiments.

Impact of camera models. Different brands of cameras can vary in terms of image size, sharpness, color saturation, and other factors that may affect the effectiveness of our attack. To investigate this impact, we conducted our attack using three camera models: Logitech C92E, Philips SPL6306BM, and Orbbec Astrapro.

As shown in Figure 14 (left), we find that our attack was similarly effective across different camera models, with an ASR greater than 80% and F_{succ} >85%. This is likely because commercial systems are typically trained on datasets collected from a wide range of image sources, including various camera brands and models, giving them robust generalization ability. In fact, our attack manipulates the feature anchor of an insider into the vicinity of the attacker's feature spaces. As long as the target system can accurately map input face images into the correct feature space, our attack can be successfully launched. The robustness of commercial systems satisfies this condition, which allows our attack to cope with different camera models.

Impact of camera resolutions. In addition to camera models, face authentication systems may also be set to different resolutions, which can affect the effectiveness of our attack. To study this impact, we conducted experiments using cameras with resolutions ranging from 120p to 1080p.

As shown in Figure 14 (right), we find that our attack maintained an ASR greater than 80% under all resolutions we tested. However, F_{succ} dropped to 67% when the resolution was set to 120p, indicating that our attack is more difficult to successfully launch under low-resolution cameras. The reason is that the face authentication system first detects and crops the face area before scaling it to a fixed size (e.g., 112 $\times$ 112 pixels) and then feeding it into the feature extractor. When low-resolution images are scaled up to a larger size through interpolation, they introduce noise and affect the authentication results. On the other hand, high-resolution images are compressed to a fixed size, ensuring that the scaled face images are nearly the same as those with increased resolutions. Since the camera resolutions used in face authentication are all larger than 240p, our attack remains threatening.

TABLE VI: Summary of Questionnaire Survey

No.	Question
1	Have you used face photos to enroll for a FAS? *(Yes / No)*
2	Which type of enrollment methods did you use? *(Uploading photos / Taking photos on the spot / Both)*
3	Were you supervised during the enrollment process? *(Yes / No)*

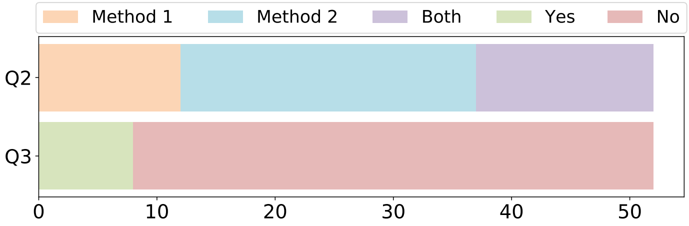


Fig. 15: The results of user study. (a) Question 2: 48.08% of participants chose Method 2 (taking photos on the spot), 23.08% of participants chose Method 1 (uploading photos), and 28.84% of participants chose both. (b) Question 3: 15.38% of participants chose Yes and 84.63% of participants chose No.

VII. User Study

In this section, we conduct a user study to measure users' attitudes towards the enrollment phase of the face authentication systems. Our aim is to investigate (1) what types of enrollment methods are widely used (Q2 in Tab. VI), and (2) whether the enrollment is supervised in real-world scenarios (Q3 in Tab. VI). To achieve these goals, we recruited 52 volunteers and conducted a questionnaire survey for each participant. The volunteers include 32 males and 20 females with ages ranging from 20 to 60 years old, and they all have prior experience in using face authentication systems. Approval of all ethical and experimental procedures and protocols was granted by Science and Technology Ethics Committee of Zhejiang University.

From the results shown in Fig. 15 (up), we find 23.08% of participants chose to upload the photos, 48.08% of participants chose to take photos on the spot, and 28.84% of participants chose both methods. It indicates that both types of enrollment methods are widely used in real-world face authentication systems. For the method of uploading photos, to the best of our knowledge, users can upload their face photos in a private environment, and the uploaded photos will not be checked by humans or machines yet. Therefore, it is feasible for an insider to wear a disguise and complete the enrollment process under this type of enrollment method. For the method of taking photos on the spot, according to the results shown in Fig. 15 (bottom), we find the face enrollment process is often unsupervised by security guards in real-world scenarios, with 84.63% of participants choosing the unsupervised option. Therefore, in this paper, we assume the insider can wear the disguise and complete enrollment without raising suspicions, e.g., when security guards are not present or not paying attention. Additionally, given guards typically lack information security backgrounds, it is possible that the attack could succeed even under their supervision, as they might not recognize it as a malicious attempt.

VIII. Discussion

In this section, we discuss the potential countermeasures against our attack and the limitation of our attack.

A. Countermeasures

Improving Model Robustness. Our attack exploits vulnerabilities in CNN-based feature extraction algorithms employed by face authentication systems. To enhance the robustness of these models, we can implement two approaches: (1) train more robust models that maximize the separation between individual identities within the feature space, and (2) incorporate handcrafted features such as SIFT [27], LBP [5, 8], and HOG [22], as adversarial attacks are not effective against these features.

Detecting Adversarial Patches. Our attack employs adversarial patches during the enrollment phase to manipulate the feature anchors of the insider, enabling adversaries to bypass face authentication. Detection techniques such as PatchGard [37] and PatchCleaner [38] can aid in identifying the presence of these adversarial patches during the enrollment. We conduct evaluations of such defense methods on the MagFace-50 model and observe that the attack success rate of `UniID` remains at 79% under a single adversary scenario. Therefore, using a patch detection methods designed for image classification tasks cannot easily defeat our attack.

Using Multiple Models. Adopting a multi-model fusion technique may help to fortify the defense against our attack. Such fusion process can be able to detect and counteract attacks that may have succeeded at the individual model level. Moreover, the increased complexity resulting from the deployment of diverse models and the fusion features presents hurdles for attackers, thereby reducing the possibility of successful breaches.

B. Limitations

Our attack has the following limitations at present. First, affixing an adversarial patch to the victim's face may alter security guards, although they do not presently monitor users during the enrollment phase. To enhance the stealthiness of our attack, we can generate the adversarial patch with more natural content by employing content constraint methods such as Adv-makeup [42]. Second, our attack need an insider to enroll the `UniID`, which introduces additional challenges when attempting to execute the attack in real-world scenarios. To alleviate this difficulty, we intend to explore the feasibility of projecting the adversarial patch onto the face of legitimate users during the enrollment process, thus making them the unwitting victim. Lastly, the target systems we attack are all RGB-based face authentication systems. However, some systems employ infrared light for identifying individuals during nighttime recognition. In such cases, we can employ an infrared projector to project infrared adversarial patches onto the insider's face, thereby achieving our attack. We will address these issues and continue to improve and refine our attack in future work.

IX. Related Works

In this section,we summarize the related work on face spoofing attacks, including the multiple-identity face spoofing attack and the backdoor attacks against face recognition.

A. Multiple-identity face spoofing attack

Face Morphing Attacks. Adversaries can fool the face authentication systems by fusing the several images to a synthetic face image that contains characteristics of multiple identities, which is known as the face morphing attack. In such attacks, there are two primary paradigms: (1) Morphing in the image space: In this approach, the synthetic face image is generated by interpolating between the face images of different identities [30, 36]. (2) Morphing in the feature space: This technique fuses the feature vectors of different identities and uses Generative Adversarial Networks (GANs) [2, 28, 35] or adversarial training [33] to reconstruct a photorealistic synthesized face image. These attacks essentially create a realistic "average face" in the digital world and have proven to be effective in bypassing face authentication systems, even fooling human experts [29]. However, these attacks can only be executed by injecting the morphed face image through cyber hijacking, which limits their ability to pose a threat to face authentication systems in the physical world.

Universal Adversarial Attacks. An Universal adversarial attack is the attack strategy that allows for targeting multiple objectives using a single perturbation or patch. To against the face recognition, Amada T et al. [1] utilized global pixel-level adversarial perturbations to launch the multiple-identity attack against four face recognition models, both in white-box and black-box setting. However, the practical deployment of pixel-level perturbations in the physical world remains challenging, rendering them ineffective for attacking real-world face authentication systems. In contrast, Yang et al. [40], and Adv-Mask[45] developed a universal patch-based adversarial attack against face detection model in the physical world, enabling adversaries to evade face detection in surveillance systems. Although their methods effectively conceal identity, they do not succeed in fooling face authentication systems.

Compared to existing works, our attack generates a universal adversarial patch capable of executing a multi-identity attack against face recognition tasks. Moreover, our attack is a physical attack against black-box systems, which boosts the threat to face authentication systems in the real world.

B. Backdoor attacks against face recognition

Backdoor attacks [16, 39] inject specific triggers into face recognition models by poisoning a small ratio of the training data. This manipulation causes the target system to misidentify individuals when these triggers are present. However, these backdoor attacks often assume that adversaries can directly access the training data, which is unrealistic in commercial systems. In contrast, Li et al [23] proposed a method where backdoor triggers are injected during the enrollment phase without requiring direct access to the training data. Nonetheless, the adversary still needs to carry the specific trigger during the recognition phase. Unlike these prior works, our attack employs an adversarial patch that can be considered a special backdoor trigger injected during the enrollment phase. However, our method distinguishes itself by removing the need for the adversary to disguise themselves in order to successfully deceive the face authentication system during the recognition phase.

X. Conclusion

In this paper, we investigate the feasibility of allowing multiple adversaries to perform face spoofing attacks without any additional disguise and propose `UniID` which enables insiders to register a universal identity into the face recognition database by wearing an adversarial patch. Evaluations with six face recognition models and two commercial face authentication systems, ArcSoft and Face++, demonstrate the effectiveness of `UniID` in both the simulation and the real world. This work injects a universal identity by leveraging the necessary enrollment phase within the workflow of face authentication, which can be extended to other enrollment-verification systems. Future directions include investigating the threat of `UniID` to other open-set tasks beyond face authentication.

Acknowledgments

We thank the anonymous shepherd and reviewers for their valuable comments. This work is supported by the National Natural Science Foundation of China (NSFC) Grant 61925109, 62071428, 62222114, 62271280.

References

[1] T. Amada, S. P. Liew, K. Kakizaki, and T. Araki, "Universal adversarial spoofing attacks against face recognition," in *2021 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2021, pp. 1–7.

[2] J. T. Andrews, T. Tanay, and L. D. Griffin, "Multiple-identity image attacks against face-based identity verification," *arXiv preprint arXiv:1906.08507*, 2019.

[3] ArcSoft, "Arcsoft technology." [EB/OL], https://www.arcsoft.com/technology/face.html.

[4] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *International conference on machine learning*. PMLR, 2018, pp. 284–293.

[5] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face anti-spoofing based on color texture analysis," in *2015 IEEE international conference on image processing (ICIP)*. IEEE, 2015, pp. 2636–2640.

[6] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *International Conference on Automatic Face and Gesture Recognition*, 2018.

[7] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[8] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, "Lbp- top based countermeasure against face spoofing attacks," in *Computer Vision-ACCV 2012 Workshops: ACCV 2012 International Workshops, Daejeon, Korea, November 5-6, 2012, Revised Selected Papers, Part I 11*. Springer, 2013, pp. 121–132.

[9] D. Deb, J. Zhang, and A. K. Jain, "Advfaces: Adversarial face synthesis," in *2020 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2020, pp. 1–10.

[10] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[11] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[12] Y. Dong, H. Su, B. Wu, Z. Li, W. Liu, T. Zhang, and J. Zhu, "Efficient decision-based black-box adversarial attacks on face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7714–7722.

[13] Face++, "Face++ face compare sdk." [EB/OL], https://www.faceplusplus.com/sdk/face-comparing/.

[14] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[15] C. Guo, J. Gardner, Y. You, A. G. Wilson, and K. Weinberger, "Simple black-box adversarial attacks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2484–2493.

[16] W. Guo, B. Tondi, and M. Barni, "A master key backdoor for universal impersonation attack against dnn-based face verification," *Pattern Recognition Letters*, vol. 144, pp. 61–67, 2021.

[17] HikVision, "Hikvision face recognition terminals." [EB/OL], https://www.hikvision.com/en/products/Access-Control-Products/Face-Recognition-Terminals/.

[18] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Effificient convolutional neural networks for mobile vision applications. arxiv 2017," *arXiv preprint arXiv:1704.04861*.

[19] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[20] Invixium, "Invixium remote face enrollment service." [EB/OL], https://www.invixium.com/remote-face-enrollment/.

[21] S. Komkov and A. Petiushko, "Advhat: Real-world adversarial attack on arcface face id system," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 819–826.

[22] J. Komulainen, A. Hadid, and M. Pietikäinen, "Context based face anti-spoofing," in *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*. IEEE, 2013, pp. 1–8.

[23] H. Li, Y. Wang, X. Xie, Y. Liu, S. Wang, R. Wan, L.-P. Chau, and A. C. Kot, "Light can hack your face! black-box backdoor attack on face recognition systems," *arXiv preprint arXiv:2009.06996*, 2020.

[24] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[25] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1. Atlanta, Georgia, USA, 2013, p. 3.

[26] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European symposium on security and privacy (EuroS&P)*. IEEE, 2016, pp. 372–387.

[27] K. Patel, H. Han, and A. K. Jain, "Secure face unlock: Spoof detection on smartphones," *IEEE transactions on information forensics and security*, vol. 11, no. 10, pp. 2268–2283, 2016.

[28] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional gans for image editing," *arXiv preprint arXiv:1611.06355*, 2016.

[29] D. J. Robertson, A. Mungall, D. G. Watson, K. A. Wade, S. J. Nightingale, and S. Butler, "Detecting morphed passport photos: a training and individual differences approach," *Cognitive research: principles and implications*, vol. 3, no. 1, pp. 1–11, 2018.

[30] U. Scherhag, R. Raghavendra, K. B. Raja, M. Gomez-Barrero, C. Rathgeb, and C. Busch, "On the vulnerability of face recognition systems towards morphed face attacks," in *2017 5th international workshop on biometrics and forensics (IWBF)*. IEEE, 2017, pp. 1–6.

[31] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[32] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proceedings of the 2016 acm sigsac conference on computer and communications security*, 2016, pp. 1528–1540.

[33] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras, "Neural face editing with intrinsic image disentangling," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5541–5550.

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

[35] Z. Wang, X. Tang, W. Luo, and S. Gao, "Face aging with identity-preserved conditional generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7939–7947.

[36] G. Wolberg, "Image morphing: a survey," *The visual computer*, vol. 14, no. 8-9, pp. 360–372, 1998.

[37] C. Xiang, A. N. Bhagoji, V. Sehwag, and P. Mittal, "Patchguard: A provably robust defense against adversarial patches via small receptive fields and masking." in *USENIX Security Symposium*, 2021, pp. 2237–2254.

[38] C. Xiang, S. Mahloujifar, and P. Mittal, "{PatchCleanser}: Certifiably robust defense against adversarial patches for any image classifier," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2065–2082.

[39] M. Xue, C. He, J. Wang, and W. Liu, "Backdoors hidden in facial features: a novel invisible backdoor attack against face recognition systems," *Peer-to-Peer*

Networking and Applications, vol. 14, pp. 1458–1474, 2021.

[40] X. Yang, F. Wei, H. Zhang, and J. Zhu, "Design and interpretation of universal adversarial patches in face detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*. Springer, 2020, pp. 174–191.

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

[42] B. Yin, W. Wang, T. Yao, J. Guo, Z. Kong, S. Ding, J. Li, and C. Liu, "Adv-makeup: A new imperceptible and transferable attack on face recognition," *arXiv preprint arXiv:2105.03162*, 2021.

[43] Y. Zhong and W. Deng, "Towards transferable adversarial attack against deep face recognition," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1452–1466, 2020.

[44] ——, "Towards transferable adversarial attack against deep face recognition," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1452–1466, 2021.

[45] A. Zolfi, S. Avidan, Y. Elovici, and A. Shabtai, "Adversarial mask: Real-world universal adversarial attack on face recognition model," *arXiv preprint arXiv:2111.10759*, 2021.

Appendix

A. Attack Performance of Cross-datasets Evaluation

To evaluate the attack performance of cross-datasets against the six tested models, we conducted experiments by selecting insiders and attackers from different datasets. Specifically, we randomly selected 100 individuals from either the LFW or CelebA dataset as insiders and selected the attacker combinations from the other dataset. Similar to Section VI-A1, the patch size is fixed at 25 x 25 pixels, with the patch position located in the eyebrow region. We used the ASR metric to evaluate the attack performance as well in this section.

The results demonstrate that the success rate of our attack varies depending on the dataset used to select insiders and attackers. When we select insiders from the LFW dataset and attackers from the CelebA dataset, our attack achieved an average success rate of 99.8%, 97.8%, 90.7%, 85.7%, 77.7%, 71.8%, 66.2%, 52.3%, 55.8%, 54.3% for 1 to 10 attackers, respectively. when we selected insiders from the CelebA dataset and attackers from the LFW dataset, the average attack success rates is 99.3%, 88.3%, 71.7%, 50.3%, 40.7%, 31.2%, 24.5%, 19.2%, 14.3%, 11.2% for 1 to 10 attackers, respectively. This difference in results can be attributed to the CelebA dataset having more identities, allowing for a more suitable combination of attackers when used as an attacker dataset.