2023 IEEE Symposium on Security and Privacy (SP) | 978-1-6654-9336-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/SP46215.2023.10179429

DepthFake: Spoofing 3D Face Authentication with a 2D Photo

Zhihao Wu[1], Yushi Cheng[2], Jiahui Yang[1], Xiaoyu Ji[1†], Wenyuan Xu[1†]
[1]Ubiquitous System Security Lab (USSLAB), Zhejiang University
[2]Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University
{zhiaowu, yushicheng, garfield_young, xji, wyxu}@zju.edu.cn

***Abstract*—Face authentication has been widely used in access control, and the latest 3D face authentication systems employ 3D liveness detection techniques to cope with the photo replay attacks, whereby an attacker uses a 2D photo to bypass the authentication. In this paper, we analyze the security of 3D liveness detection systems that utilize structured light depth cameras and discover a new attack surface against 3D face authentication systems. We propose DepthFake attacks that can spoof a 3D face authentication using only one single 2D photo. To achieve this goal, DepthFake first estimates the 3D depth information of a target victim's face from his 2D photo. Then, DepthFake projects the carefully-crafted scatter patterns embedded with the face depth information, in order to empower the 2D photo with 3D authentication properties. We overcome a collection of practical challenges, e.g., depth estimation errors from 2D photos, depth images forgery based on structured light, the alignment of the RGB image and depth images for a face, and implemented DepthFake in laboratory setups. We validated DepthFake on 3 commercial face authentication systems (i.e., Tencent Cloud, Baidu Cloud, and 3DiVi) and one commercial access control device. The results over 50 users demonstrate that DepthFake achieves an overall Depth attack success rate of 79.4% and RGB-D attack success rate of 59.4% in the real world.**

1. Introduction

A face authentication system verifies the legitimacy of a user by matching a human face, usually in the form of a digital image, against the ones in the database. Such a system has been used in unlocking devices, securing financial payments, and physical access control to critical infrastructures. Because face authentication requires no physical contact compared to other biometrics such as fingerprints, it has become one of the most popular authentication methods, especially in the COVID-19 era. The face authentication systems that purely rely on 2D images are known to be vulnerable to 2D replay attacks [4, 14, 43], e.g., an attacker may fool the systems with a printed photo of the legitimate user. To cope with such attacks, manufacturers such as Apple, Baidu, Tencent, etc., [5, 6, 52] start to utilize 3D liveness detection techniques to distinguish a real human from a printed photo, and we call such authentication systems *3D face authentication systems*. Specifically, they exploit a depth camera to obtain the RGB and depth images to extract 3D properties of human faces, e.g., the depth and the skin texture of faces. Since a printed 2D image does not contain 3D properties, 3D face authentication systems can block existing 2D spoofing attacks [37, 49].

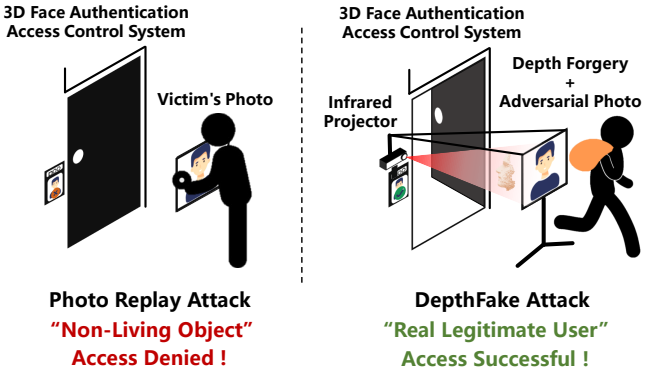


Figure 1. The DepthFake attack bypasses the 3D face authentication with liveness detection by projecting a crafted structured light scatter pattern onto a public 2D photo of the target victim.

In this paper, we seek to investigate *"Is it indeed impossible to spoof a 3D face authentication system using only one single 2D photo?"* Theoretically, if one can forge the depth information and play it to empower the 2D photo with 3D properties, such that the obtained depth images and RGB images match the ones of the legitimate users, and the 3D face authentication can be fooled. We call such an attack DepthFake and envision the following scenario, as shown in Fig. 1. An attacker can obtain an RGB photo of a victim and estimate the 3D properties of the victim beforehand. To impersonate the victim, she emits an infrared pattern with the victim's 3D depth information embedded and perfectly aligned with the 2D photo, such that it appears as if the victim is standing in front and thus bypass the 3D liveness detection.

DepthFake is made challenging because of the constraints of one photo and the limited information of the target authentication system. For instance, the authentication algorithm parameters and details are unknown, and it is unclear how the depth-based and the RGB-based liveness detection are performed. Nevertheless, the key of a successful DepthFake is to answer the following questions.

Demo: https://sites.google.com/view/depthfake.
†Corresponding author

DOI 10.1109/SP46215.2023.00098

a) How to obtain the depth information from one single photo? *b*) How to convert the depth information into an infrared pattern such that it can spoof the target depth camera? *c*) How to modify the printed photo such that it can fool RGB-based liveness detection simultaneously?

To overcome the above challenges, we first propose a CNN-based deep learning model with the weighted loss function to estimate the depth information out of a single photo. Then, we model the depth measurement process and design the mapping function that can convert the digital depth image into the desired scatter pattern such that the captured depth images appear similar to the ones of a live human. Finally, we propose an evolutionary-based RGB adversarial attack method with color calibration, and a face region alignment scheme to form an RGB-D attack, i.e., bypass both RGB-based and depth-based liveness detections. We validate the effectiveness of `DepthFake` attacks on three commercial face authentication systems, e.g., Tencent Cloud, Baidu Cloud, and 3DiVi, and a commercial access control device in a laboratory setup. The validation on 50 users shows that `DepthFake` achieve a success rate of 79.4% for the systems with depth-based liveness detection and a success rate of 59.4% for the systems with both depth-based and RGB-based liveness detection. In summary, our contributions include the points below:

- We identify the vulnerabilities in the 3D liveness detection of the face authentication system, and propose to spoof a 3D face authentication system using a single 2D photo.
- We design `DepthFake` that can spoof commercial face authentication systems by projecting the carefully-crafted structured light scatter pattern on a printed photo of a victim.
- We validate `DepthFake` on a commercial structured light depth camera (i.e., Astra Pro), three commercial face authentication systems (i.e., Baidu Cloud, Tencent Cloud, 3DiVi), and one commercial access control device in the laboratory set up, and achieve a success rate of 79.4% for the systems with depth-based liveness detection and a success rate of 59.4% for the systems with both depth-based and RGB-based liveness detection.

`DepthFake` attacks server as the first attempt to fool 3D face authentication systems. To enhance the security of existing systems, we recommend two defense methods: *a*) Randomizing scatter patterns to make it extremely difficult for attackers to forge depth information, while existing depth cameras use a fixed scatter pattern. *b*) Enhancing 3D liveness detection models to detect the forgery depth information and adversarial examples.

2. Background

In this section, we first present the work-flow of the 3D face authentication system and then introduce the liveness detection in detail.

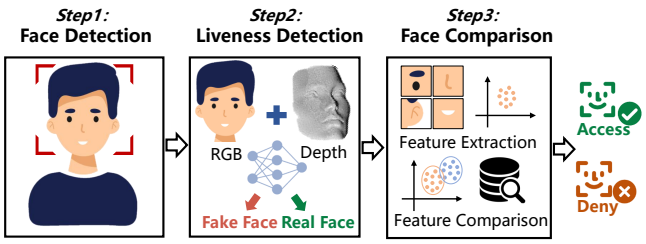


Figure 2. The 3D face authentication system first detects faces in the captured RGB and depth images, then determines whether those faces are from real people, and finally verifies whether they are from legitimate users.

2.1. 3D Face Authentication

Face authentication is a biometrical authentication scheme used to verify the identity of a user. To prevent a face authentication system from photo replay attacks, 3D face authentication is invented by adding the 3D liveness detection module.

A standard 3D face authentication system typically employs a visible light camera and an infrared camera. When a user attempts to authenticate, RGB and depth images are captured. As shown in Fig 2, the 3D face authentication process has the following three steps:

Step 1: Face Detection. The face detection step detects and crops the face regions in the RGB and depth images. For RGB images, face detection algorithms such as multi-task convolutional neural network (MTCNN) [58] are used to locate the face area in the RGB images and output the bounding-box coordinates. Based on the face area in an RGB image, the face area in a depth image is extracted by mapping the face bounding-box coordinates of a RGB image to that of the depth image.

Step 2: Liveness Detection. The liveness detection step analyzes the liveness properties, such as the edges, textures, Moiré patterns from an RGB image and the 3D depth information from a depth image. This step aims to defend against spoofing attacks such as photo replay attacks, video replays attack, and mask-wearing attacks [4, 8, 14, 38, 43].

Step 3: Face Comparison. After the liveness detection step, the face comparison step employs comparison algorithms such as FaceNet [45] to verify if the user is the legitimate one. The comparison algorithms often extract a feature vector from the newly captured face images and calculate the similarity distance between it and the enrolled feature vector stored in the database. Note that face comparison only relies on the RGB images because of their high resolution.

From the workflow of the face authentication system, we find that the key to fooling a 3D face authentication system is to bypass its liveness detection module. In the following, we introduce the liveness detection in detail.

2.2. Liveness Detection

Liveness detection determines whether the person to be authenticated is a live person or not. It aims to reject non-living objects that try to obfuscate the face authentication system, e.g., a printed photo. Existing liveness detection

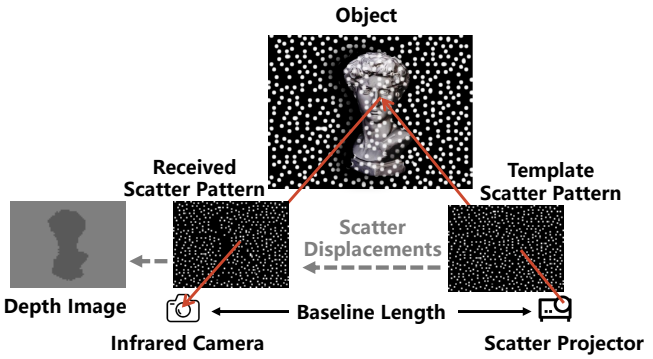


Figure 3. The structured light depth camera actively projects a constant scatter pattern to the object, e.g., a face, then captures the reflected scatter pattern and calculates the depth of the target by measuring its displacement from the template scatter pattern for each scatter point.

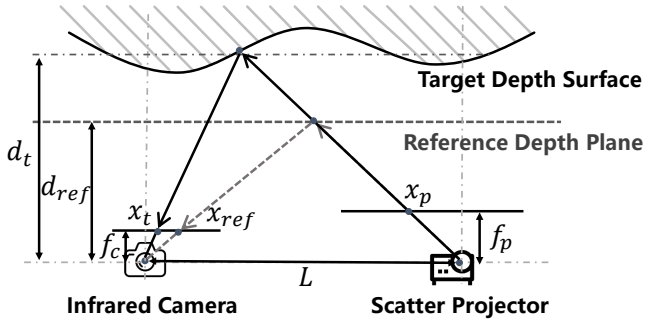


Figure 4. Imaging mechanism of the structured light depth camera. The scatter projector projects the scatter point to the target depth plane, then the infrared camera captures the reflected scatter point and calculates the target depth by measuring its displacement to the template scatter point.

methods include two categories: (1) Active liveness detection requires the user to perform a predefined action, e.g., blink or nod. This method needs multiple captured images to recognize a pre-defied action and therefore is commonly used in cloud-service-based face authentication due to its extra requirements on computing resources. (2) Passive liveness detection, on the contrary, only uses one-shot images to determine whether the user is alive. Since passive liveness detection is lightweight and requires little user interaction, it is a popular method used nowadays in smartphones, smart locks, access control devices, etc [15]. Thus, in this paper, we target the 3D liveness detection in a passive manner.

2.3. 3D Passive Liveness Detection

3D passive liveness detection utilizes both RGB and depth images, and it may use one of the following types of depth cameras: (1) structured light camera which uses the infrared scatter pattern to encode depth information, (2) time-of-flight (ToF) camera which calculates the depth by recording the infrared light echo time, and (3) binocular stereo camera, which obtains depth information by matching different perspectives photos taken by two cameras. Among them, the structured-light-based depth camera is the most widely used one for face authentication systems, because of its high resolution, insensitivity to visible light, and low cost. For instance, it is used by FaceID [12, 29], and Smartlock[54]. Since the structured-light-based depth cameras are reported to occupy almost 25% among all the depth camera market [42], in this paper, we focus on the passive liveness detection systems using structured-light-based depth cameras.

Structured-light-based 3D Liveness Detection. A standard structured light depth camera contains a scatter projector and an infrared camera, as shown in Fig. 3. Most scatter projector projects a constant infrared pattern containing tens of thousands of scatter points. During calibration in the factory, the infrared camera captures the scatter patten at the reference depth plane and stores it as a template scatter pattern. To obtain a depth image of a user, the infrared camera captures the reflected scatter pattern, which is distorted due to the different depths of the face, and it calculates the depth information by measuring the displacements between each scatter point with the one stored in the template scatter pattern.

To illustrate how the structured light depth camera calculates the depth information from scatter point displacements, we consider a single scatter point. As shown in Fig. 4, as an infrared beam of x_p is reflected at the plane of reference depth d_{ref} and tatget depth d_t, the scatter points on the image sensor of infrared camera can be denoted as x_{ref} and x_t, respectively. Based on the rule of the similar triangle, we can calculate d_t as follows:

$$d_t = \frac{d_{ref} L f_c}{L f_c - d_{ref} \Delta x_c} \tag{1}$$

where $\Delta x_c = x_{ref} - x_t$ is the displacement between x_{ref} and x_t. The baseline length L, the focal length f_c, and reference depth d_{ref} are constant and known to the depth camera. Thus, the camera can obtain the depth of a single point and multiple points across the entire face.

After obtaining RGB and depth images, the 3D liveness detection module utilizes both of them to determine the liveness. For an RGB image, deep learning algorithms such as Convolution Neural Networks (CNNs) [16, 35, 56] or Vision Transformer (Vit) [21], are used to extract features, e.g., edges, textures, Moiré patterns or features in the frequency domain in the RGB image, to determine whether the RGB stands for a real person. For a depth image, the region with the face is extracted by mapping to the RGB image, and then feature point matching algorithms [24] or CNNs [22, 51] are used to determine whether the face belongs to a real person by detecting the special geometric structure of the face.

3. Threat Model

The goal of the `DepthFake` attack is to spoof a 3D face authentication system using a 2D photo by bypassing its 3D liveness detection module. We consider the following attack scenario: An adversary wants to get inside a confidential place where the access control device is equipped with a 3D face authentication system. To achieve it, she launches a `DepthFake` attack by placing the target victim's printed

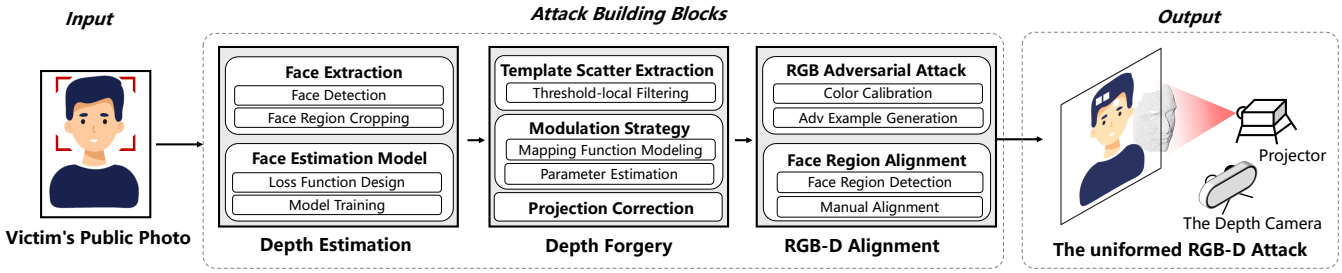


Figure 5. Overview of DepthFake attack: The adversary first estimates the depth image from the victim's 2D photo. Then, she extracts the template scatter pattern and modulates the depth information to the desired scatter pattern for depth forgery. To finally bypass the 3D face authentication, the adversary also uses the RGB adversarial attack and aligns it with the forgery depth image to launch a uniform RGB-D attack.

photo in front of the camera of the authentication system, as shown in Fig. 1 and projecting the carefully-crafted scatter pattern onto the printed photo to spoof the 3D face authentication system.

The victim authentication system is supposed to employ both RGB and depth liveness detection techniques. To achieve the aforementioned attack, the adversary has the following capabilities:

Depth Camera Awareness. The adversary can acquire a depth camera of the same model as the one used in the victim system. The attacker can obtain the template scatter pattern of the victim system from the substitute camera by capturing an infrared image.

Public Photo Access. The adversary can obtain a 2D photo of the victim from public platforms such as his social media like Facebook, Twitter, WeChat, etc.

Physical Access to the Victim Device. The adversary can physically get close to the victim system and set up the attack device, i.e., the printed photo displayed on a board and the infrared projector.

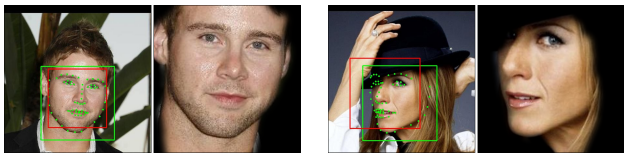
Black-box Setting. We assume the target liveness detection systems is black-box. For depth forgery attacks, the adversary does not require any feedback from the victim systems, and for RGB liveness detection attacks, we assume she can obtain the confidence score from the victim system, which is a common assumption in most prior work [3, 25, 46, 55].

4. DepthFake Attack Design

4.1. Overview

DepthFake attack investigates the feasibility of spoofing the 3D face authentication system using a 2D photo of a legitimate user by bypassing its 3D liveness detection module. To guarantee an effective and robust spoofing attack in the real world, it is important to answer the following questions:

- **Q 1:** How to obtain the depth information from one single photo?
- **Q 2:** How to convert the depth information into an infrared pattern such that it can spoof the target depth camera?
- **Q 3:** How to modify the printed photo such that it can fool RGB-based liveness detection simultaneously?



(a) Front face (b) Side face

Figure 6. Face regions extracted from the photo of victims. The green points are the 68 feature landmark points. The red and green bounding-boxes represent the face regions detected by Dlib and our method, respectively.

To address these challenges, DepthFake incorporates three major modules, as shown in Fig. 5. The **Depth Estimation** module detects and extracts the face region from a 2D public photo of the victim, and estimates its depth information through a CNN-based deep learning model. The **Depth Forgery** module forges a depth image of the target user by projecting the scatter pattern derived from the depth estimation module using an infrared projector. The **RGB-D Attack** module spoofs the liveness detection safeguarded by the RGB and depth simultaneously. First, we generate an RGB adversarial image using an evolutionary strategy to bypass the RGB liveness detection, and then we physically align the projected scatter pattern with the forged RGB image to launch the uniform RGB-D attack. In the following sections, we present these attack building blocks in detail.

4.2. Depth Estimation

To spoof 3D face authentication systems using a single photo, we first estimate and reconstruct a depth image of the victim from his 2D photo. In general, the depth estimation has two steps: (1) Face extraction which extracts the victim's face region from his/her 2D photo to eliminate the influence of background elements, and (2) Depth estimation which estimates the depth information of the face region and generates a pixel-to-pixel depth image by using a CNN model.

4.2.1. Face Extraction. To estimate user depth information via a single photo, we first detect and crop the face region to improve the processing efficiency.

Face Detection. For face detection, a commonly-used tool is the Dlib Library [31]. However, using this tool alone cannot detect the face region completely, especially the side face, as shown in the red bounding box in Fig. 6. To obtain

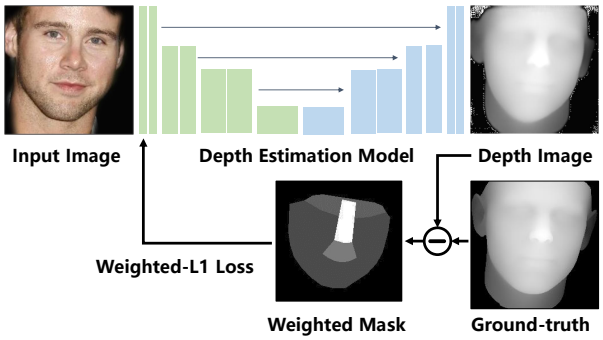


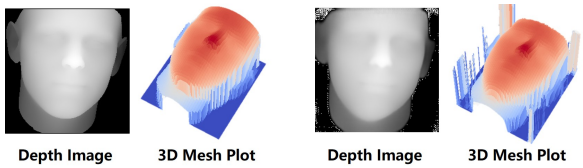
Figure 7. Model architecture for depth estimation.The depth estimation model generates the depth image from the input RGB image, and use its weighted mask to build the loss function.

the face region precisely, we improve the face detection function in the Dlib Library by considering the following two aspects: (1) the size of the bounding box shall be appropriate to contain the entire face including the contours, and (2) the face shall locate at the center of the bounding box to reduce background elements. To achieve it, we first utilize the shape predictor of the Dlib Library to landmark 68 key feature points of the face. Then, we use these feature points to determine the coordinates of the center point and the side length of the bounding box such that the face can be entirely contained and located in the center of the bounding box. As shown in Fig. 6, our face detection method can extract the face region precisely and completely for both front and side face images.

Face Region Cropping. After extracting the face region from the victim's photo, we crop and resize it to 224 $\times$ 224 pixels, i.e., the default input size for the following depth estimation model.

4.2.2. CNN-based Face Depth Estimation. To estimate the depth information from the cropped face region, we then propose a pixel-to-pixel method based on convolution neural networks (CNNs). We employ the UNet [44] as our basic model architecture. It uses the ResNet-50 [30] as the encoder to reduce a 224 $\times$ 224 input image into a 7 $\times$ 7 embedding feature map, and uses a decoder formed by 5 transposed convolutional layers and 10 convolutional layers to reconstruct the feature map into a 224 $\times$ 224 depth image, where each pixel represents the absolute value of the depth. Then, we design the loss function to optimize the aforementioned estimation process.

Loss Function Design. In general, U-Net utilizes the L1 Loss as the loss function to optimize the depth estimation task. However, the L1 Loss treats all pixels in the image equally, leading to depth estimation errors in regions such as noses, eyes, and mouths. Those regions, however, are important in feature representation and depth reconstruction. To address it, we propose to employ a weighted L1 Loss, which assigns more weights to the central parts of the face (i.e., the nose, eyes, and mouth) compared with other regions. Specifically, we use the landmark feature points to locate the key regions and assign different weights to them to form a weight mask. The weight mask is determined



(a) Ground truth (b) Estimation result

Figure 8. Depth estimation results. (a) is the ground-truth depth image and its 3D mesh plot of the target image. (b) is the estimated depth image and its 3D mesh plot from our model.

by experiments and we employ the best one as the default setting, i.e., 2 for the nose region, 1.5 for the eye and mouth regions, 1 for other face regions, and 0 for non-face regions, as shown in Fig. 7. In this way, we make the model pay more attention to reconstructing the depth of the nose, eyes and, mouth regions. The proposed weighted L1 Loss is as follows:

$$Weight{-}L1 = \frac{1}{N}\sum_{i=1}^{N}|y_i - f(x_i)| \times WeightMask \quad (2)$$

where x and y are the input image and the ground-truth depth image, and $f(\cdot)$ is the depth estimation model.

Model Training. We split the 300W-3D dataset [60] into 3 separated sub-datasets with a ratio of 7:1:2 : (1) Training dataset, (2) Validation dataset, and (3) Testing dataset. Specifically, we use (1) and (2) for training, and (3) for testing and evaluation in Sec. 5. Additionally, we employ Adam with a learning rate of $1e{-}5$ as the optimizer to train the depth estimation model. An illustration of the depth estimation result is shown in Fig. 8. Compared to the ground-truth image and its 3D mesh plot, the depth image estimated from the 2D image shows a normalized mean error of less than 2% and can spoof the depth-based liveness detection module in commercial face authentication SDKs (i.e., Tencent, Baidu, and 3DiVi) with nearly 100% attack success rates.

4.3. Depth Forgery

The structured light depth camera calculates the depth information by measuring the displacements of scatter points between the template scatter pattern and the reflected one. To spoof it, the adversary shall modulate the estimated depth information into the template scatter pattern such that it can be captured and accepted by the depth camera. To achieve it, we propose a depth forgery method consisting of two steps: (1) extracting the template scatter pattern of the victim camera, and (2) modulating the estimated depth image into the template scatter pattern to form the spoofing scatter pattern.

4.3.1. Template Scatter Extraction. Different structured light depth cameras use different template scatter patterns. Thus, we shall first obtain the template scatter pattern of the victim camera. A naive but effective method is to capture the image containing a template scatter pattern with an infrared camera. However, the raw infrared image usually

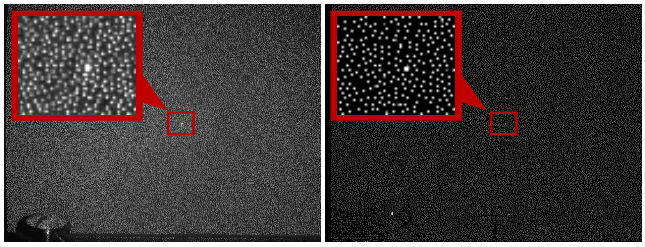


Figure 9. Template Extraction. Images from left to right are the raw infrared image of the template scatter pattern and the extracted template scatter pattern respectively.

suffers from various noises, rendering the captured template scatter pattern not precise and thus introducing extra errors in depth forgery. To extract a clear and precise template scatter pattern, we propose an image noise reduction method called local-threshold filtering.

Since the scatter point is usually the brightest in its surrounding neighborhood while the noises can be dim, we can use the non-maximum suppression (NMS) [23] to remove the background noises. NMS is a mathematical method for picking the maximum value within an array while suppressing other values. For instance, if we have an array A as $\{A_1, A_2, ..., A_n\}$ and the maximum value of A is A_{max}, the NMS algorithm will only retain A_{max} and set the other value to 0 as follows:

$$\textbf{NMS}(A) = \{0, 0, ..., A_{max}, ..., 0\} \tag{3}$$

To employ NMS for image processing, we extend the one-dimensional NMS into two-dimensional by using the sliding window. Specifically, we extract the pixels with the maximum grayscale value within each small region. Since a scatter may cross more than one pixel when captured by the infrared camera, the pixels around the maximum value pixel may also have larger grayscale values than other pixels. To ensure the precise of the extracted template scatter pattern, we retain the pixels whose grayscale value larger than a threshold and set the others to 0. We set the threshold dynamically with the following equation since the grayscale of the scatter points in the sliding window is related to the background brightness.

$$threshold = -0.001(p_{max})^2 + 1.0001p_{max} \tag{4}$$

where $p_{max} \in [0, 255]$ is the maximum value in the sliding window. An illustration of the extracted template scatter pattern is shown in Fig. 9.

4.3.2. Depth-to-Scatter Modulation Strategy. Then, we modulate the estimated depth information into the extracted template scatter pattern by shifting the locations of its scatter points. Two key questions here are, for each scatter point in the template: (1) What's its depth value? (2) What is the corresponding displacement that represents the depth value?

For the first question, we address it by coordinate alignment. We first extract the coordinates of each scatter point in the template as a set $T = \{(x_1, y_1)).., (x_n, y_n)\}$. Then, based on the coordinates in T, we extract the corresponding depth information in the depth map as the set $D = (d_1, ..., d_n)$. The depth-to-scatter modulation process is then can be represented as follows:

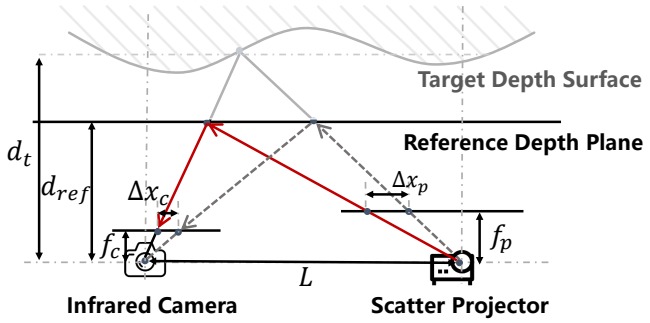


Figure 10. Depth forgery. The scatter point p is shifted by Δx_p and projected onto the plane of depth d_{ref}, making it produce a displacement of Δx_c on the imaging plane of the infrared camera. Thus, the camera can be spoofed to calculate the forgery depth d_t.

$$S = T + \Phi(D) \tag{5}$$

where $\Phi(\cdot)$ is the mapping function that converts the depth information to the scatter displacement, and S is the desired scatter pattern. Thus, the key to modulate the estimated depth image into the scatter pattern is to build the mapping function $\Phi(\cdot)$.

Mapping Function Modeling. Based on Eq. 1 in Sec. 2, the structured light camera uses the reference depth d_{ref} and the displacement Δx_c to calculate the target depth d_t. Thus, if a scatter point has a depth value of d_t, we can obtain its displacement in the camera Δx_c as follows:

$$\Delta x_c = k_c L f_c (\frac{1}{d_{ref}} - \frac{1}{d_t}) \tag{6}$$

where k_c is the number of pixels within a physical length ($1\ mm$) in the camera, L is the baseline distance between the camera and the projector, and f_c is the focal length of the camera.

The displacement in the camera Δx_c is then can be converted to the displacement in the projector Δx_p as follows:

$$\Delta x_p = \frac{k_p f_p}{k_c f_c} \Delta x_c \tag{7}$$

where the f_p is the focal length of the projector, and k_p is the number of pixels within a physical length in the projector.

Thus, the mapping function between the depth value d_t and the scatter point displacement Δx_p can be expressed as follows:

$$\Delta x_p = k_p L f_p (\frac{1}{d_{ref}} - \frac{1}{d_t}) \tag{8}$$

Note that parameters k_p, L, and f_p are related to the attack device (i.e., the infrared projector) only. Thus, the adversary does not need to know any internal parameters about the target device, making the attack more practical in the real world.

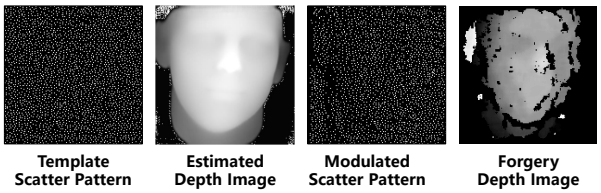


Figure 11. Depth-to-Scatter Mapping. The digital depth image uses the mapping function to convert the estimated depth information into a desired scatter pattern, which is then captured by the depth camera as a forged depth image.

Parameter Estimation. Among the above three parameters, the baseline length L and the focal length of the projector f_p can be measured directly while the number of pixels within a physical length (k_p) cannot. To address it, we take k_pLf_p as a joint parameter and estimate it integrally. We record the scatter patterns on planes at depths of $1000mm$, $900mm$, and $800mm$, and use the displacements between them to determine the joint parameter. Specifically, we use the 10×10 pixels region on the center of the $1000mm$ depth plane as the template scatter pattern. Then, we use the Peak Singal-to-Noise Ratio (PSNR) as a matching function to search for the region with the highest match score and measure the displacement between them. Based on the displacements in different depths, we can estimate the joint parameter k_pLf_p. For an instance, in this paper, we use a baseline length L of $20mm$, and a reference depth of $1000mm$. By measuring the displacements in depths of $1000mm$, $900mm$, and $800mm$, we can estimate the joint parameter k_pLf_p as 4400 and obtain the mapping function $\Phi(\cdot)$ as follows:

$$\Phi(d) = 4400(\frac{1}{1000} - \frac{1}{d}) \tag{9}$$

where the d is the target depth. By modulating the depth information of each scatter point in the set T, we can obtain the desired scatter pattern S, as shown in Fig. 11.

4.3.3. Projection Correction. When projecting the forged scatter pattern in practice, there can be a physical distance L between the projector and the victim depth camera, causing projection distortion on the captured scatter pattern. To address it, we employ the perspective transformation [11] before projecting, which is commonly used for image correction in computer vision. Based on the perspective transformation function shown in Eq. 10, we capture both the origin and distorted scatter patterns, select four vertices of the face bounding box as the reference points, and compute the parameter matrix by comparing the pixel offsets between each pair of the vertices.

$$\begin{bmatrix}\tilde{x} & \tilde{y} & \tilde{w}\end{bmatrix} = \begin{bmatrix}x & y & w\end{bmatrix} \begin{bmatrix}a_{11} & a_{11} & a_{11} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33}\end{bmatrix} \tag{10}$$

where (x, y) is the point in the original image, and $w = 1$. Then, we use Eq. 11 to compensate for the origin scatter pattern to ensure the captured one is not distorted.

$$x^{'} = \frac{\tilde{x}}{\tilde{w}}; y^{'} = \frac{\tilde{y}}{\tilde{w}} \tag{11}$$

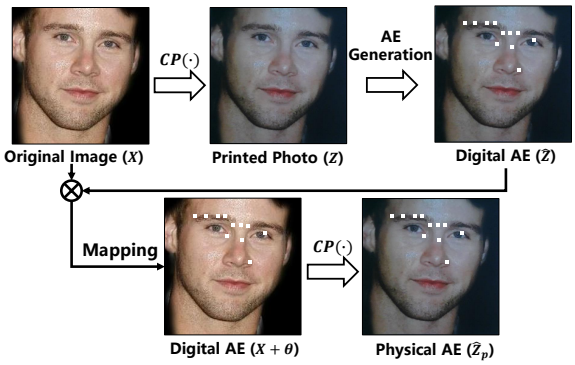


Figure 12. The adversarial perturbation θ generated based on the printed photo Z is applied to the original image X to form the digital adversarial example $\widehat{Z}$, which is then printed and captured as the physical adversarial photo $\widehat{Z}_p$.

4.4. RGB-D Attack

With the above two attack building blocks, we can spoof depth-based liveness detection. However, some commercial systems may use both RGB and depth for liveness detection. In this case, we shall spoof the RGB-based liveness detection in addition to conducting the depth forgery attack. To achieve this, we propose a black-box optimization algorithm to generate RGB adversarial examples. Then, we print and align it with the forged scatter pattern to launch a uniform RGB-D attack.

4.4.1. RGB Adversarial Attack. As the RGB-based liveness detection usually use CNN as its backbone, we can spoof it with adversarial examples. In general, an adversarial example in our case can be denoted as follows:

$$\widehat{Z} = Z + \theta \tag{12}$$

where Z is a printed photo detected as "non-live object" in normal circumstances, θ is the optimized adversarial perturbation, and $\widehat{Z}$ is the adversarial example that can successfully bypass the RGB-based liveness detection in the digital world. However, directly printing the adversarial example as an adversarial photo is not enough to guarantee an effective attack in the real world, since it suffers from color distortions from the printing-capturing process, resulting in a decrease in the attack effectiveness. To address it, we first build a printing-capturing process model and then compensate for the color distortions in both the printed photo Z and the adversarial perturbation θ.

Color Calibration. To calibrate the color distortion caused by the printing-capturing process, we first build a model to describe it as follows:

$$\widehat{Z}_p = C[P(Z + \theta)] = C[P(Z)] + C[P(\theta)] \tag{13}$$

where $P(\cdot)$ and $C(\cdot)$ are the abstract transfer functions without loss of generality. Since the printing-capturing process mainly processes each pixel independently [57], we can decouple them. Based on the Eq. 13, we can remove the effect of color distortions on the printed photo Z and the adversarial perturbation θ separately.

Algorithm 1: Black-box Adversarial Perturbation Generation

Input: Printing-Capturing Process $C[P(\cdot)]$, the legitimate user's RGB image X, RGB-based liveness detection black-box model M and its liveness $Threshold$, perturbation uint size (w, h)

Capture the printed RGB image X as $Z = C[(P(X)]$
Input Z into M, and the model returns face position coordinates (x_1, y_1, x_2, y_2) and confidence score S_l.
$Z_{init} \leftarrow Z$
$x \leftarrow x_1$
$y \leftarrow y_1$
repeat
  $Z_{temp} \leftarrow Z$
  Update Z_{temp}: Set the $Z_{temp}^{(i,j)}$ into $(R, G, B)_{white}$, where $x <= i <= x + w, y <= j <= y + h$
  Input Z_{temp} into M, and get confidence score S_t.
  if $S_t > S_l$ **then**
    Update $Z \leftarrow Z_{temp}$, $S_l \leftarrow S_t$
  end if
  $x \leftarrow x + w$ and $y \leftarrow y + h$
until $S_l > Threshold$
$\theta \leftarrow Z - Z_{init}$
Map the θ to original RGB image X to form the digital adversarial example $X + \theta$.

Output: The digital world adversarial example $X + \theta$.

As shown in Fig. 12, since the printed photo Z is the original image X after a printing-capturing process, i.e., $Z = C[P(X)]$, we can replace Z with the origin image X in Eq. 13 to eliminate the color distortions. For the adversarial perturbation θ, we use white adversarial units instead of pixel-level adversarial perturbations for two reasons: (1) The color "white" can resist the color shift caused by the printing-capturing process. For printing, white is represented by 0 in the CMYK [57] color mode used by printers, indicating that no ink is jetted and thus no color shift will occur. For capturing, white is used as the base color by cameras for automatic white balance, thus it is least affected by ambient lights. (2) Using a square (3*3 pixels) instead of a pixel as the basic unit of the adversarial perturbations can better resist the surrounding noises.

By implementing the color calibration, we can get a desired adversarial photo as follows:

$$\widehat{Z}_p = C[P(X + \theta)] = C[P(X)] + C[P(\theta)] = Z + \theta \quad (14)$$

Black-box Adversarial Example Generation. With the selected perturbation color and size, we then generate the adversarial perturbations θ that can bypass the RGB-based liveness detection in the digital world. In this paper, we consider the RGB-based liveness detection to be a black-box and thus we can only get the confidence scores of the liveness detection results. As a result, we propose a query-based evolutionary strategy to generate the adversarial perturbations, as shown in Algorithm 1. Specifically, we use a printed photo Z as the input and utilize a 2D adversarial perturbation unit with a size of $w \times h$ to scan through its face region boxed by the face authentication system. Each time after adding an adversarial perturbation unit, we get the liveness confidence score from the SDKs or APIs and retain the adversarial perturbation unit that can raise the liveness confidence score. Different from other black-box adversarial attacks, we do not have to consider the stealthiness of adversarial examples in this paper. As a result, we do not consider the shortest distance between the adversarial example and the original image during optimization.

After scanning the whole face region, we extract the adversarial perturbation θ and apply it to the original image X to form the digital adversarial example, which is then printed and captured as the physical adversarial photo $\widehat{Z}_p$.

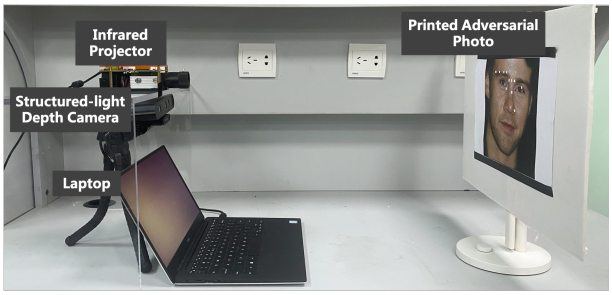


Figure 13. Experimental setup. An infrared projector is used to project a structured light scatter pattern onto an adversarial photo of a legitimate user present in front of the target module to launch RGB-D attacks.

TABLE 1. DEFAULT PARAMETERS DURING EVALUATION

Resolution	640×480 pixels (480p)	
Light Condition	illumination intensity	$300\ lx$
	color temperature	$6500\ K$
Thresholds	Tencent Cloud	RGB: 0.4, Depth: 0.5
	Baidu Cloud	RGB: 0.8, Depth: 0.8
	3DiVi	RGB: 0.9, Depth: 0.5

4.4.2. Face Region Alignment. To align the RGB adversarial photo and the depth scatter pattern to ensure a uniform RGB-D attack, we localize five key face feature points (i.e., eyes, nose tip, and mouth corners) in both depth and RGB images. Then, we fix the distance between the projector and the printed RGB adversarial photo, then align the feature points by adjusting the position and angle of the photo.

5. Evaluation

In this section, we evaluate `DepthFake` against three commercial liveness detection modules in the real world. We use the attack success rate (SR) as the metric to evaluate our attack, which is the ratio of the number of successful attacks over the total number of conducted attacks.

5.1. Experimental Setup

Target Systems. We use a commercial 3D camera Orbbec Astra Pro [18] equipped with an RGB camera and a structured light depth camera as the hardware of our target

TABLE 2. OVERALL PERFORMANCE OF DEPTHFAKE ATTACKS WITH 50 USERS AGAINST THREE DIFFERENT LIVENESS DETECTION MODULES.

Datasets	Modalities	Target Systems		
		Tencent Cloud	Baidu Cloud	3DiVi
300W-3D	Depth	*74.1%* (THR:0.5)	*68.0%* (THR:0.8)	*95.2%* (THR:0.5)
	RGB-D	*49.0%* (THR:0.4, 0.5)	*45.0%* (THR:0.8, 0.8)	*71.2%* (THR:0.9, 0.5)
Texas-3DFR	Depth	*73.2%* (THR:0.5)	*70.2%* (THR:0.8)	*94.3%* (THR:0.5)
	RGB-D	*45.1%* (THR:0.4, 0.5)	*38.0%* (THR:0.8, 0.8)	*56.2%* (THR:0.9, 0.5)
Volunteers	Depth	*72.5%* (THR:0.5)	*68.3% (THR:0.8)*	*98.5%* (THR:0.5)
	RGB-D	*72.5%* (THR:0.4, 0.5)	*61.1%* (THR:0.8, 0.8)	*78.1%* (THR:0.9, 0.5)

systems, as shown in Fig. 13. We use three commercial face authentication SDKs/APIs as the software of our target systems and try to spoof their liveness detection modules. The three SDKs/APIs are (1) Tencent Cloud [52], (2) Baidu Cloud [6], and (3) 3DiVi Face [2]. We implement these SDKs/APIs on a DELL XPS 13 laptop and acquire their liveness confidence scores by calling interfaces.

Attack Devices. We use an infrared projector DLP4500SL02 Evaluation Module [17] as the attack device to project the modulated structured light scatter pattern, which has a resolution of 1280×800 pixels, and a projecting image size of 270 $mm \times 168$ mm. In addition, we use a printed adversarial photo of a legitimate user placed in front of the camera to spoof the RGB-based liveness detection, as shown in Fig. 13.

Default Attack Setting. During the experiments, we put the infrared projector above the target 3D camera with a distance of 20 mm and fix the distance between the projector and the adversarial photo as 500 mm. The adversarial photo is printed with a size of 400 $mm \times 300$ mm. Other default settings including the camera resolution, the light condition, and the thresholds of target systems are shown in Tab. 1.

Datasets. We evaluate our attacks on two famous face datasets 300W-3D [60] and Texas-3DFR [27, 28]. For each dataset, we select 20 users with different genders, ages, and races to launch `DepthFake`.

Volunteers. We recruit ten volunteers including six males and four females to evaluate the effectiveness of `DepthFake` attacks in the real world.

5.2. Overall Performance

We first evaluate the overall performance of the `DepthFake` attack on different people against three commercial liveness detection modules, i.e., Tencent Cloud, Baidu Cloud, and 3DiVi, under the default setting, and record attack success rates of 1,000 frames per user. The results shown in Tab. 2 demonstrate that the Depth attack can achieve an overall attack success rate of 73.3% against Tencent Cloud, 68.8% against Baidu Cloud, and 96% against 3DiVi. The RGB-D attack can achieve an overall attack success rate of 55.5% against Tencent Cloud, 48% against Baidu Cloud, and 68.5% against 3DiVi. We find that the attack performance of the RGB-D attack on volunteers is better than that of the 300W-3D dataset or the Texas-3DFR dataset. The reason is that some images in these datasets are of poor quality, leading to the reduced performance of the RGB-D attack. However, the results from ten real-humans show that our attack can achieve the average attack success rate at 69.1%, indicating that `DepthFake` attack performs well both at Depth and RGB-D attack. Among the three tested systems, 3DiVi is the most vulnerable while Baidu Cloud is the least. The reason is that Baidu Cloud uses higher default thresholds for both the RGB and Depth liveness detection. Therefore, we evaluate the attack performance under the common thresholds among the three tested systems, the results can be found in Appendix A. Another finding is that RGB-D-based liveness detection is more difficult to attack compared with the Depth-based one. The reason is that the RGB-D attack requires the alignment of RGB and depth images but the projection distortion caused by the alignment process can lead to the reduction of attack performance.

Overall, our attack can achieve an overall attack success rate over 79.4% for the Depth attack and 57.4% for the RGB-D attack against different people and systems.

5.3. Impact of Light Condition

Then, we evaluate the light condition that may influence the attack effectiveness of `DepthFake`, including the illumination intensity and the color temperature. During the experiments, we use Tencent Cloud SDK with default thresholds as our target system and one volunteer as the victim. To avoid the mutual interference between the light intensity and color temperature, we keep the color temperature to 6500 K when evaluating the impact of the illumination intensity, and keep the illumination intensity to 300 lx when evaluating the impact of the color temperature.

Illumination Intensity. To investigate the impact of the illumination intensity, we conduct experiments by setting the background illumination intensity from 50 lx to 400 lx. From the results shown in Fig. 14 (left), we find that both the Depth attacks and the RGB-D attacks can achieve average attack success rates of over 70%. When the illumination intensity drops to 100 lx, the attack success rate of the RGB-D attack shows a slight decrease. It is because that the camera suffers from more noise when capturing images at a low illumination intensity. Another finding is that the attack performance at 50 lx is better than that of 100 lx since the camera will compensate for the exposure when the illumination intensity is too low. On the contrary, the attack success rate of the Depth attack remains $\geq 70\%$ across different illumination intensities. The reason is that the structured light depth camera uses an infrared scatter

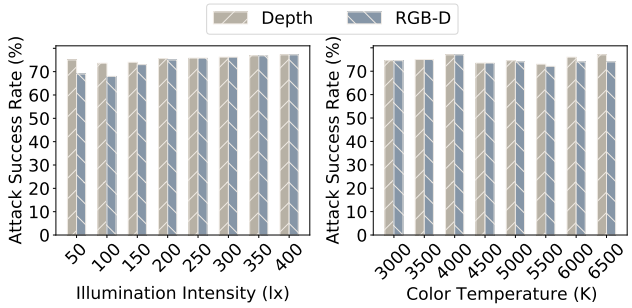


Figure 14. Impact of DepthFake attacks under various light conditions.

pattern to generate depth information, which is not subject to interference from visible lights.

Color Temperature. Similar to the illumination intensity, our attack may be affected by the color temperature. To investigate its impact, we conduct experiments by setting the background color temperature from 3000 K to 6500 K. The results are shown in Fig. 14 (right). From the results, we find that the attack performance of both the Depth and RGB-D attacks are unaffected as the color temperature changes, with attack success rates higher than 70%.

Therefore, for various light conditions, we find that (1) they have impacts on the RGB-D attack, but almost no influence on the Depth attack because it is based on infrared lights, and (2) DepthFake attacks can achieve an attack success rate of over 70% under most light conditions.

5.4. Impact of Camera Resolution

Commercial face authentication systems may use cameras of various resolutions to capture images, which may influence the performance of our attack. To study its impact, we conduct experiments by using cameras of different resolutions including 120p, 240p, 480p, 720p, and 960p. During the experiments, we employ Tencent Cloud and Baidu Cloud as our target models, where the former uses the entire image for detection while the latter only uses the face region.

From the results shown in Fig. 15, we find that the attack success rates of both the Depth and RGB-D attacks against two victim systems decline when the camera resolution drops to 240p. The reasons are: (1) For depth images, the lower camera resolution may cause the forged depth image to lose the 3D geometry structure of the human face. (2) For RGB adversarial photos, the reduction in camera resolutions can weaken the attack effectiveness of the adversarial perturbations.

In addition, we find that with a camera resolution of 720p, the attack success rate against Tencent Cloud is lower than that of Baidu Cloud. The reason is that 720p images use a aspect ratio of 16 : 9, while 480p images (default resolution) use 4 : 3. Since Tencent Cloud uses the full image for detection, the change in the aspect ratio will make the image different from the printed adversarial example, leading to a decrease in the effectiveness of the RGB attack. On the contrary, Baidu Cloud uses the face area for detection, which is not affected by the aspect ratio since the face area ratio is fixed.

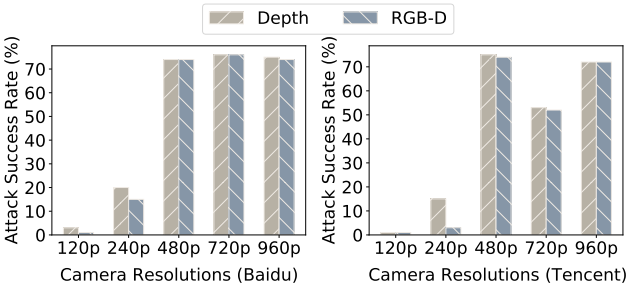


Figure 15. The attack effectiveness of DepthFake attacks under various camera resolutions.

Therefore, the DepthFake attack works better with cameras with high resolutions. However, with today's trend of using high-resolution cameras for surveillance, we assume DepthFake still has its threat in the real world.

5.5. Impact of Photo Quality

The public photos we obtained from victims' social medias are usually with different face angles and resolutions. In this subsection, we evaluate the attack effectiveness of our attacks under photos with different face angles and resolutions.

Face Angles. To investigate the impact of face angles, we conduct experiments using public images with different face angles in both horizontal (from $-90°$ to $90°$) and vertical ($-60°$ to $60°$) directions.

The results are shown in Fig. 16. For the Depth attack, we find that the attack success rate can achieve over 70% at any face angle. For the RGB-D attack, the attack success rate is susceptible to the face angle. Specifically, the attack success rate can achieve over 35% when the face angle is between $-30°$ to $30°$ in both the horizontal and vertical directions. However, when the face angle is larger than $60°$, the performance of the RGB-D attack decreases. The reason is that when the face angle increases, the facial information contained in the photo decreases, increasing the difficulty of the RGB adversarial attack and thus the RGB-D attack.

Photo Resolutions. To evaluate the impact of photo resolutions, we conduct experiments using public photos with different resolutions, i.e., 960p, 720p, 480p, 240p, and 120p. From the results shown in Fig. 16, we find that both Depth and RGB-D attacks can achieve attack success rates over 70% on photos with resolutions larger than 480p. However, when the image resolution drops to 240p, the performance of the RGB-D attack decreases. The reason is that the RGB-based liveness detection model can detect the blurred image and output it as a 'non-living object'. For the Depth attack, it is not affected by low image resolutions since: (1) In the depth estimation phase, our training dataset contains photos of various resolutions, making the depth estimation model robust to low-resolution photos. (2) In the depth forgery phase, the scatter pattern we modulate is not related to the resolution of the original RGB photo.

Therefore, DepthFake attacks work better with public photos with face angles $\leq 30°$ and image resolutions $\geq$480p.

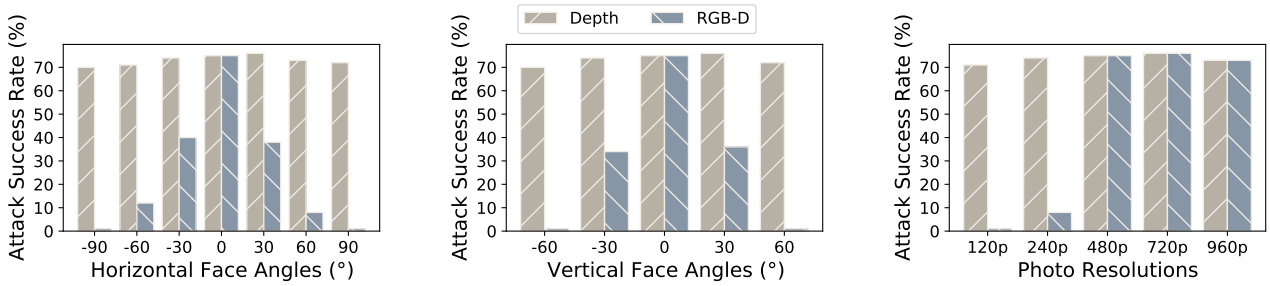


Figure 16. The attack effectiveness of DepthFake attacks under various photo qualities.

5.6. Impact of Relative Position

DepthFake attacks use an external infrared projector to project the forged scatter pattern to an adversarial photo to launch RGB-D attacks. As a result, the relative position between the camera and the projector may distort the scatter pattern and thus influence the depth generation. In this subsection, we evaluate the attack effectiveness of Depth and RGB-D attacks under different camera-projector distances.

Horizontal Distance. We put the projector 2 cm above the camera and change their horizontal distance from 0 cm to 12 cm to evaluate the impact of horizontal distances. From the results shown in Fig. 17 (top), we find that the attack success rates for both the Depth and RGB-D attacks drop as the horizontal distance increases. With a horizontal distance of 4 cm, both the Depth and RGB-D attacks can achieve attack success rates of about 70%. However, when the relative horizontal distance is larger than 6 cm, the attack success rate is reduced to 30%. The reason is that the depth generation highly depends on the location of the scatter pattern. The severe perspective distortion caused by the long horizontal distance will shift the projected scatter, resulting in a performance decrease.

Vertical Distance. To evaluate the impact of vertical distances, we set the horizontal distance between the projector and camera to be 0 cm and change its vertical distance from 2 cm to 8 cm. The minimal vertical distance is set to 2 cm instead of 0 cm since the 3D camera has a shell of 2 cm.The results shown in Fig. 17 (bottom) demonstrate that the performance of the Depth attack and the RGB-D attack declines as the vertical distance increases. Nevertheless, both the Depth and RGB-D attacks can achieve attack success rates over 60% when the vertical distance is less than 5 cm.

Therefore, DepthFake attacks can tolerate a horizontal or vertical distance within 5 cm between the projector and the camera.

5.7. End-to-End Face Authentication

The DepthFake attack targets the 3D liveness detection module in the commercial face authentication systems, based on the hypothesis that if we bypass the 3D liveness detection, the face authentication system can be spoofed with a single photo. To verify it, we conduct experiments against end-to-end face authentication systems. Specifically, we evaluate the attack effectiveness of DepthFake attacks against each module in the standard workflow of face authentication systems, including face detection, liveness detection, and face comparison.

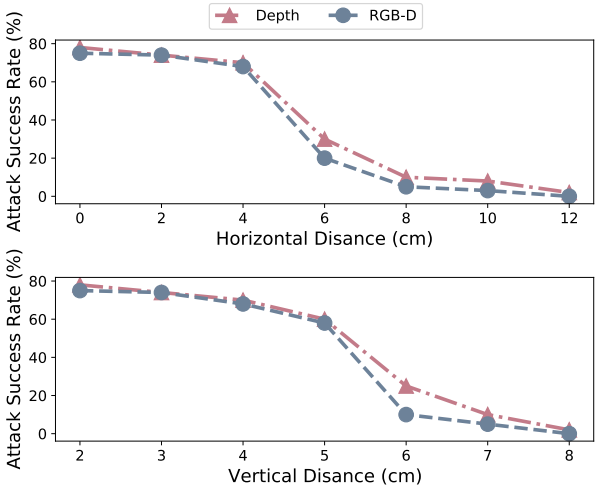


Figure 17. Impact of relative positions between the camera and projector on DepthFake attacks.

TABLE 3. ATTACK EFFECTIVENESS OF DEPTHFAKE ATTACKS ON END-TO-END FACE AUTHENTICATION SYSTEMS.

Target Systems	Workflow of Face Authentication System		
	Face Detection	Liveness Detection	Face Comparison
Tencent Cloud	100%	72.5%	100%
Baidu Cloud	100%	61.1%	100%
3DiVi	100%	78.1%	100%

We conduct experiments with ten users against three face authentication systems with their default thresholds. For the results shown in Tab. 3, we find that DepthFake attacks can pass every step of the face authentication system and successfully spoof the entire system. Moreover, we find that once the liveness detection step is bypassed, the following face comparison step can be 100% spoofed. The reason is that DepthFake attacks do not make any change to the face presentation features of the legitimate user. Furthermore, most commercial face authentication systems only rely on RGB images for face comparisons. The adversarial perturbations we generate are small and sparse, and thus do not affect the face comparison step.

Figure 18. Experimental setup against an commercial access control device. An infrared projector is used to project a structured light scatter pattern onto an adversarial photo of a legitimate user present in front of the target module.

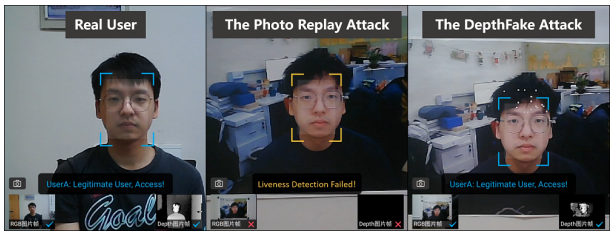


Figure 19. An illustration of the face authentication results on the real user, the photo replay attack and the DepthFake attack.

6. Case Study: Access Control Device

In this section, we evaluate the effectiveness of DepthFake attacks on a commercial access control device Baidu Rattlesnake Application Kit in the real world as a case study.

6.1. Experimental Setup

Target Device. We analyze the commercial access control device Baidu Rattlesnake Application Kit equipped with a Baidu Face Authentication SDK and an Orbbec Astra depth camera in this paper, which has been used in airports, metros, banks, and other critical infrastructures in China [7]. The liveness detection module is set to the RGB-D mode with default thresholds (i.e., RGB:0.5, Depth:0.5).

Attack Devices. For Depth attacks, we use an infrared projector DLP4500SL02 Evaluation Module and place it on the top of the camera with a distance of 20 mm. For RGB attacks, we use a 400 $mm \times 300$ mm printed adversarial photo of the legitimate user and place it in front of the target device with a distance of 500 mm, as shown in Fig. 18.

6.2. Attack Performance

We conduct experiments with the Baidu Rattlesnake Application Kit on five legitimate users. Before experiments, we first generate the users' modulated structured light scatter patterns and their RGB adversarial photos. During attacks, we cover the scatter projector of the target depth camera and use the infrared projector to project the modulated scatter pattern to the printed adversarial photo. Then, the depth camera will feed the captured RGB and depth images into the face authentication system of the target device, and output the authentication results.

An illustration of the real-world DepthFake attack against the commercial access control device is shown in Fig. 19. The results show that 3D liveness detection can defend against the naive photo replay attack, but is not effective when against the DepthFake attack.

To quantitatively illustrate the effectiveness of the DepthFake attack, we launch attacks for 1000 frames per use, and record the attack success rate and the maximum consecutive success frames.

As shown in Tab. 4 of Appendix, we find that our attack is effective against ten users, and can achieve an average attack success rate of 61.16%. Compared to the simulation, the performance decrease in the real-world setting is due to the projection distortion caused by the distance between the infrared projector and the victim device. Meanwhile, the results demonstrate that our attack can succeed over 6 continuous frames, indicating the feasibility of our attacks against commercial products utilizing multiple frames for liveness detection.

7. Discussion

In this section, we discuss the potential countermeasures and limitations against the DepthFake attack.

7.1. Countermeasure

Detection Model Improvement. DepthFake attacks exploit the vulnerabilities of the deep-learning-based RGB liveness detection algorithms. As a result, liveness detection methods with handcrafted features, e.g., LBP [9, 19], SIFT [41], SURF [10], and HOG [33], may help increase the robustness of the model and thus defend our attacks.

Adversarial Examples Detection. DepthFake attacks utilize the adversarial photos to attack the RGB liveness detection. As a result, adversarial example detection methods such as Feature Distillation [34], Local Intrinsic Dimentionaloty [36], and DkNN [40] may help detect the existence of RGB adversarial photos and thus our attacks.

Depth Area Size Detection. DepthFake attacks forge the depth information by projecting the modulated structured light scatter pattern with an infrared projector. Since the commercial infrared projector usually has a limited projection size, it cannot generate depth information covering the entire image. As a result, the size of the area containing depth information can be detected to help determine whether the system is suffering from DepthFake attacks.

Randomized Template Scatter Pattern. DepthFake attacks spoof the depth camera by modulating the depth information into the template scatter pattern. As a result, using randomized template scatter patterns can increase the attack difficulty, since the attacker needs to obtain the current template scatter pattern and modulate depth information in time.

7.2. Limitation

Our attack has the following limitations at present. First, our attack targets RGB-D liveness detection as it is the most common method for 3D liveness detection nowadays. However, some commercial products may use liveness detection of other modalities such as the IR-D liveness detection. In this case, our attack shall be extended to include the IR-D attack by using the infrared projector to replay the IR images. Second, our RGB adversarial photo is optimized based on the confidence score of the RGB liveness detection, which may not always be available on commercial devices. In this case, we shall try to generate adversarial photos by using transfer-based black-box optimization methods. Third, the portability of our attack can be further improved. We will explore using portable attack devices such as mini infrared projectors to make our attack more flexible and practical. Lastly, while our attack has yielded good performances when using only a single 2D photo, using multiple victim photos to fuse depth images of various face angles can help to improve face depth estimation accuracy, and further enhance the attack performances. We remain the aforementioned issues as our future work.

8. Related Work

In this section, we summarize the related work on face spoofing attacks, including adversarial attacks against face authentication systems and spoofing attacks against depth sensors.

8.1. Adversarial attacks against face recognition systems

In the earlier time, adversaries use photos [4, 14], videos [43], and 3D masks [8, 38], a.k.a., facial presentation attacks, to spoof face authentication systems, which however can be well detected and defended by today's deep learning algorithms. Deep-learning-based face authentication systems are effective in detecting those facial presentation attacks yet vulnerable to adversarial attacks, and much prior work has demonstrated the feasibility of spoofing face authentication systems with adversarial examples in both digital and physical worlds.

Digital adversarial attacks usually employ subtle adversarial perturbations at the pixel level, and spoof face authentication systems without human perception. Compared to white-box attacks, black-box ones are more challenging. In this area, DFANet [59] applied adversarial examples to black-box models by using the transferability of the adversarial attacks. Dong et al. [20] proposed an evolutionary optimization method to generate adversarial faces against decision-based black-box models.

Physical adversarial attacks focus on their capabilities to be deployed in the real physical world. In this area, the adversarial patch draws much attention since pixel-level adversarial perturbations are difficult to achieve in the physical world. A common method is to attach or print the adversarial patch on wearable stuffs such as eyeglasses [46, 48], face masks [61], hats [32], stickers [26], etc, to spoofing face authentication systems. Another method is to use external light sources to produce adversarial patterns. Nguyen et al. [39] projected adversarial patterns onto faces to impersonate or obfuscate targets. Vla [47] projected adversarial perturbations onto the full face and composed a face with the target features.

Compared with prior work that mainly focuses on spoofing the face comparison step, our work tries to fool the face authentication system with a printed photo of a legitimate user by bypassing its liveness detection step.

8.2. Spoofing attacks against depth sensors

Common depth sensors that can acquire information about the depth of a target include structured light depth cameras, stereo cameras, LiDARs, etc. In the area of spoofing depth sensors, much work has been done on the LiDAR [13, 50, 53] by actively emitting laser signals and utilizing the vulnerability of its deep learning algorithms. DoubleStar [1] exploited the weakness of the stereo matching and used it to manipulate the drone, but it can only produce coarse-grained fake depths. Our work is the first one to spoof structured light depth cameras and can forge fine-grained depth information. In addition to the face authentication system analyzed in this paper, our work can be extended to other systems equipped with structured-light-based depth cameras.

9. Conclusion

In this paper, we investigate the feasibility of spoofing 3D face authentication systems with a single photo and find the key is to bypass its 3D liveness detection module. Therefore, we propose the `DepthFake` attack, which estimates the depth information from a single photo, modulates it to a structured light scatter pattern, and projects such a scatter pattern on the adversarial photo of a legitimate user to spoof the 3D face authentication system. Evaluation with three commercial face authentication systems (Tencent Cloud, Baidu Cloud, and 3DiVi) and one commercial access control device demonstrates the effectiveness of `DepthFake` attacks in the real world. In addition to the face authentication system analyzed in this paper, our work can be extended to other systems equipped with structured light depth cameras. Future directions include exploring the security of the full workflow of face authentication and the security of other depth systems.

10. 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 and China Postdoctoral Science Foundation Grant BX2021158.

References

[1] DoubleStar: Long-Range attack towards depth estimation based obstacle avoidance in autonomous systems. In *31st USENIX Security Symposium (USENIX Security 22)*, Boston, MA, August 2022. USENIX Association.

[2] 3divi. 3divi face recognition platform. [EB/OL]. https://face.3divi.com.

[3] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, pages 484–501. Springer, 2020.

[4] André Anjos and Sébastien Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. In *2011 international joint conference on Biometrics (IJCB)*, pages 1–7. IEEE, 2011.

[5] Apple. About face id advanced technology. [EB/OL]. https://support.apple.com/en-us/HT208108,.

[6] Baidu. Baidu ai cloud face recognition platform. [EB/OL]. https://intl.cloud.baidu.com/product/face.html.

[7] Baidu. Customer cases of baidu face authentication application kit. [EB/OL]. https://ai.baidu.com/customer?industry=0&technology=2&clickType=tech&caseOrderType=1,.

[8] Sushil Bhattacharjee, Amir Mohammadi, and Sébastien Marcel. Spoofing deep face recognition with custom silicone masks. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–7. IEEE, 2018.

[9] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face anti-spoofing based on color texture analysis. In *2015 IEEE international conference on image processing (ICIP)*, pages 2636–2640. IEEE, 2015.

[10] Zinelabidine Boulkenafet, Jukka Komulainen, and Abdenour Hadid. Face antispoofing using speeded-up robust features and fisher vector encoding. *IEEE Signal Processing Letters*, 24(2):141–145, 2016.

[11] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[12] Andrew Bud. Facing the future: The impact of apple faceid. *Biometric technology today*, 2018(1):5–7, 2018.

[13] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 2267–2281, 2019.

[14] Murali Mohan Chakka, Andre Anjos, Sebastien Marcel, Roberto Tronci, Daniele Muntoni, Gianluca Fadda, Maurizio Pili, Nicola Sirena, Gabriele Murgia, Marco Ristori, et al. Competition on counter measures to 2-d facial spoofing attacks. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–6. IEEE, 2011.

[15] Saptarshi Chakraborty and Dhrubajyoti Das. An overview of face liveness detection. *arXiv preprint arXiv:1405.2227*, 2014.

[16] Baoliang Chen, Wenhan Yang, and Shiqi Wang. Face anti-spoofing by fusing high and low frequency features for advanced generalization capability. In *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 199–204. IEEE, 2020.

[17] Sang Hoon Chong, Ashwin B Parthasarathy, Venkaiah C Kavuri, Frank A Moscatelli, Sunil Singhal, and Arjun G Yodh. Intraoperative nir diffuse optical tomography system based on spatially modulated illumination using the dlp4500 evaluation module (conference presentation). In *Emerging Digital Micromirror Device Based Systems and Applications IX*, volume 10117, page 101170D. International Society for Optics and Photonics, 2017.

[18] José Gomes da Silva Neto, Pedro Jorge da Lima Silva, Filipe Figueredo, João Marcelo Xavier Natário Teixeira, and Veronica Teichrieb. Comparison of rgb-d sensors for 3d reconstruction. In *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*, pages 252–261. IEEE, 2020.

[19] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel. Lbp- top based countermeasure against face spoofing attacks. In *Asian Conference on Computer Vision*, pages 121–132. Springer, 2012.

[20] Yinpeng Dong, Hang Su, Baoyuan Wu, Zhifeng Li, Wei Liu, Tong Zhang, and Jun Zhu. Efficient decision-based black-box adversarial attacks on face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7714–7722, 2019.

[21] Anjith George and Sébastien Marcel. On the effectiveness of vision transformers for zero-shot face anti-spoofing. *arXiv preprint arXiv:2011.08019*, 2020.

[22] Anjith George and Sébastien Marcel. Cross modal focal loss for rgbd face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7882–7891, 2021.

[23] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[24] Gaurav Goswami, Mayank Vatsa, and Richa Singh. Rgb-d face recognition with texture and attribute features. *IEEE Transactions on Information Forensics and Security*, 9(10):1629–1640, 2014.

[25] Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. Simple black-box adversarial attacks. In *International Conference on Machine Learning*, pages 2484–2493. PMLR, 2019.

[26] Ying Guo, Xingxing Wei, Guoqiu Wang, and Bo Zhang. Meaningful adversarial stickers for

face recognition in physical world. *arXiv preprint arXiv:2104.06728*, 2021.

[27] Shalini Gupta, Kenneth R Castleman, Mia K Markey, and Alan C Bovik. Texas 3d face recognition database. In *2010 IEEE Southwest Symposium on Image Analysis & Interpretation (SSIAI)*, pages 97–100. IEEE, 2010.

[28] Shalini Gupta, Mia K Markey, and Alan C Bovik. Anthropometric 3d face recognition. *International journal of computer vision*, 90(3):331–349, 2010.

[29] Song-Yi Han, Hyun-Ae Park, Dal-ho Cho, Kang Ryoung Park, and Sangyoun Lee. Face recognition based on near-infrared light using mobile phone. In *International Conference on Adaptive and Natural Computing Algorithms*, pages 440–448. Springer, 2007.

[30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[31] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[32] Stepan Komkov and Aleksandr Petiushko. Advhat: Real-world adversarial attack on arcface face id system. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 819–826. IEEE, 2021.

[33] Jukka Komulainen, Abdenour Hadid, and Matti Pietikäinen. Context based face anti-spoofing. In *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–8. IEEE, 2013.

[34] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 860–868. IEEE, 2019.

[35] Shiying Luo, Meina Kan, Shuzhe Wu, Xilin Chen, and Shiguang Shan. Face anti-spoofing with multi-scale information. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3402–3407. IEEE, 2018.

[36] Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*, 2018.

[37] Sébastien Marcel, Mark S Nixon, and Stan Z Li. *Handbook of biometric anti-spoofing*, volume 1. Springer, 2014.

[38] Erdogmus Nesli and Sébastien Marcel. Spoofing in 2d face recognition with 3d masks and anti-spoofing with kinect. In *IEEE 6th International Conference on Biometrics: Theory, Applications and Systems (BTAS'13)*, pages 1–8, 2013.

[39] Dinh-Luan Nguyen, Sunpreet S Arora, Yuhang Wu, and Hao Yang. Adversarial light projection attacks on face recognition systems: A feasibility study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 814–815, 2020.

[40] Nicolas Papernot and Patrick McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

[41] Keyurkumar Patel, Hu Han, and Anil K Jain. Secure face unlock: Spoof detection on smartphones. *IEEE transactions on information forensics and security*, 11(10):2268–2283, 2016.

[42] Psmarketresearch. 3d camera market size, share, development, growth and demand forecast to 2022 – industry insights by technology (time of flight, stereo vision and structured light imaging), by type (free camera and target camera), and by application (professional cameras, smartphone, tablets, computer and other). [EB/OL]. https://www.psmarketresearch.com/market-analysis/3d-camera-market,.

[43] Ramachandra Raghavendra, Kiran B Raja, and Christoph Busch. Presentation attack detection for face recognition using light field camera. *IEEE Transactions on Image Processing*, 24(3):1060–1075, 2015.

[44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[45] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[46] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael 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*, pages 1528–1540, 2016.

[47] Meng Shen, Zelin Liao, Liehuang Zhu, Ke Xu, and Xiaojiang Du. Vla: A practical visible light-based attack on face recognition systems in physical world. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–19, 2019.

[48] Inderjeet Singh, Toshinori Araki, and Kazuya Kakizaki. Powerful physical adversarial examples against practical face recognition systems. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 301–310, 2022.

[49] Luiz Souza, Luciano Oliveira, Mauricio Pamplona, and Joao Papa. How far did we get in face spoofing detection? *Engineering Applications of Artificial Intelligence*, 72:368–381, 2018.

[50] Jiachen Sun, Yulong Cao, Qi Alfred Chen, and Z Morley Mao. Towards robust lidar-based perception in autonomous driving: General black-box adversarial sensor attack and countermeasures. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 877–894, 2020.

[51] Gusi Te, Wei Hu, and Zongming Guo. Exploring hypergraph representation on face anti-spoofing beyond 2d attacks. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2020.
[52] Tencent. Tencent cloud face recognition platform. [EB/OL]. https://intl.cloud.tencent.com/products/facerecognition.html.
[53] James Tu, Mengye Ren, Sivabalan Manivasagam, Ming Liang, Bin Yang, Richard Du, Frank Cheng, and Raquel Urtasun. Physically realizable adversarial examples for lidar object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13716–13725, 2020.
[54] Muhammad Waseem, Sundar Ali Khowaja, Ramesh Kumar Ayyasamy, and Farhan Bashir. Face recognition for smart door lock system using hierarchical network. In *2020 International Conference on Computational Intelligence (ICCI)*, pages 51–56. IEEE, 2020.
[55] Xingxing Wei, Ying Guo, and Jie Yu. Adversarial sticker: A stealthy attack method in the physical world. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
[56] Jianwei Yang, Zhen Lei, and Stan Z Li. Learn convolutional neural network for face anti-spoofing. *arXiv preprint arXiv:1408.5601*, 2014.
[57] Feng Ling Yin and Bing Quan Huo. Image transformation for digital printing machine. In *Applied mechanics and materials*, volume 401, pages 180–183. Trans Tech Publ, 2013.
[58] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.
[59] Yaoyao Zhong and Weihong Deng. Towards transferable adversarial attack against deep face recognition. *IEEE Transactions on Information Forensics and Security*, 16:1452–1466, 2020.
[60] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.
[61] Alon Zolfi, Shai Avidan, Yuval Elovici, and Asaf Shabtai. Adversarial mask: Real-world adversarial attack against face recognition models. *arXiv preprint arXiv:2111.10759*, 2021.

Appendix

1. Attack performance under common thresholds

To evaluate the attack performance against three tested systems under common thresholds, we conduct experiments by setting the same threshold for the three tested systems, which ranges from 0.4 to 0.8. From the results shown in Tab. 5, we find the effectiveness of our attack decrease when the thresholds are raised, but the average attack success rate still reaches 68.04% for the Depth attack and 48.44% for the RGB-D attack. Similar to Sec. 5.2, due to the poor quality images in datasets, the attack performance of RGB-D attack on volunteers is better than datasets, especially when the thresholds increase. However, our attack can achieve an attack success rate of over 48.5% against Depth attack and 47.5% against RGB-D attack on ten real humans at the threshold of 0.8, indicating that our attack is a real threat in the real-world scenario

TABLE 4. ATTACK EFFECTIVENESS OF DEPTHFAKE ON A COMMERCIAL ACCESS CONTROL DEVICE

Victim Users	Attack Success Rate	Maximum Consecutive Success Frames
Person A	52.2%	9
Person B	62.4%	12
Person C	61.1%	15
Person D	45.5%	6
Person E	82.0%	26
Person F	62.0%	18
Person G	56.8%	9
Person H	72.1%	19
Person I	48.7%	6
Person J	68.8%	18

TABLE 5. THE PERFORMANCE OF DEPTHFAKE ATTACKS AGAINST THREE DIFFERENT LIVENESS DETECTION MODULES UNDER COMMON THRESHOLD.

Datasets	**Target Systems**	**Thresholds**					
		Depth			RGB-D		
		0.4	0.6	0.8	RGB:0.4 Depth:0.4	RGB:0.6 Depth:0.6	RGB:0.8 Depth:0.8
300W-3D	Tencent Cloud	*78.5%*	*68.1%*	*45.0%*	*62.5%*	*36.7%*	*15.5%*
	Baidu Cloud	*91.5%*	*77.2%*	*68.0%*	*91.5%*	*68.5%*	*45.0%*
	3DiVi	*99.1%*	*95.2%*	*92.2%*	*99.1%*	*83.5%*	*75.5%*
Texas-3DFR	Tencent Cloud	*76.8%*	*65.5%*	*38.5%*	*48.5%*	*32.8%*	*12.3%*
	Baidu Cloud	*91.3%*	*75.5%*	*70.2%*	*91.3%*	*54.0%*	*38.0%*
	3DiVi	*99.5%*	*93.8%*	*89.9%*	*99.1%*	*77.7%*	*59.5%*
Volunteers	Tencent Cloud	*80.5%*	*68.5%*	*48.5%*	*71.3%*	*68.0%*	*47.5%*
	Baidu Cloud	*95.8%*	*84.3%*	*64.5%*	*95.8%*	*84.3%*	*63.8%*
	3DiVi	*99.5%*	*98.8%*	*95.6%*	*99.5%*	*89.8%*	*78.9%*