

# AdvGen: Physical Adversarial Attack on Face Presentation Attack Detection Systems

Sai Amrit Patnaik<sup>1</sup>, Shivali Chansoriya<sup>1</sup>, Anoop M. Namboodiri<sup>1</sup> and Anil K. Jain<sup>2</sup>  
<sup>1</sup>IIT Hyderabad, India, <sup>2</sup>Michigan State University, USA

{sai.patnaik, shivali.chansoriya}@research.iit.ac.in, anoop@iit.ac.in, jain@cse.msu.edu

## Abstract

Evaluating the risk level of adversarial images is essential for safely deploying face authentication models in the real world. Popular approaches for physical-world attacks, such as print or replay attacks, suffer from some limitations, like including physical and geometrical artifacts. Recently adversarial attacks have gained attraction, which try to digitally deceive the learning strategy of a recognition system using slight modifications to the captured image. While most previous research assumes that the adversarial image could be digitally fed into the authentication systems, this is not always the case for systems deployed in the real world. This paper demonstrates the vulnerability of face authentication systems to adversarial images in physical world scenarios. We propose AdvGen, an automated Generative Adversarial Network, to simulate print and replay attacks and generate adversarial images that can fool state-of-the-art PADs in a physical domain attack setting. Using this attack strategy, the attack success rate goes up to 82.01%. We test AdvGen extensively on four datasets and ten state-of-the-art PADs. We also demonstrate the effectiveness of our attack by conducting experiments in a realistic, physical environment.

## 1. Introduction

Face recognition systems are extensively used in real-time applications, such as surveillance systems, forensics, automated border control, user authentication [43], payment processing, and security control systems. To prevent unauthorized access and attacks, Presentation Attack Detectors (PADs) are integrated into these systems (Figure 3) to detect and reject presentation attacks, such as print attacks and replay attacks. As presentation attacks try to bypass the authentication system, understanding and correcting the potential pitfalls of a PAD module is as essential as designing high-accuracy recognition algorithms.

Most of the current state-of-the-art approaches use auxil-

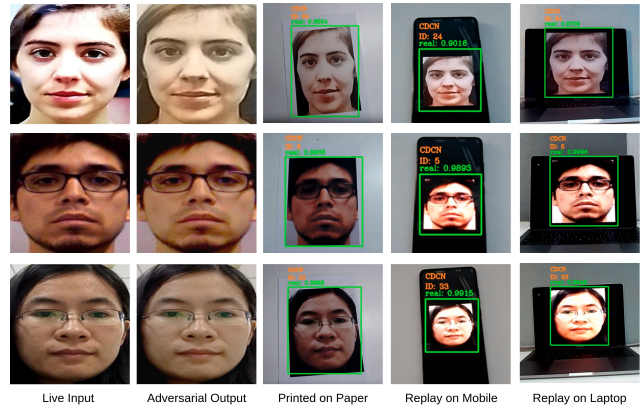


Figure 1. Example live images and corresponding adversarial images generated by AdvGen. First Column: live images from presentation attack datasets, second column: the corresponding adversarial images generated by AdvGen, third column: the predicted class along with the confidence score and recognized identity for a generated image (presenting an adversarial image generated by our model to the face recognition), fourth column: replay attack on a mobile screen, fifth column: replay attack on a laptop screen. The proposed method generates visually indistinguishable adversarial images from the input that is robust to distortions introduced after physical transformations.

ary information [55, 3, 53] to improve the performance and generalizability of the presentation attack detectors. Presentation and adversarial attacks on face recognition systems are still a significant concern. In a presentation attack, attacks are created using printed photographs, replayed videos, wearing a mask or makeup, etc. For generating presentation attacks, the hacker must actively participate by wearing a mask or replaying a photograph/video of the genuine individual, which may be conspicuous in scenarios involving human operators. Adversarial attacks, on the other hand, do not require active participation during verification.

The use of deep learning has significantly improved the accuracy of Presentation Attack Detectors. Adversarial attacks [40, 19, 33, 13], however, exploit the vulnerability of

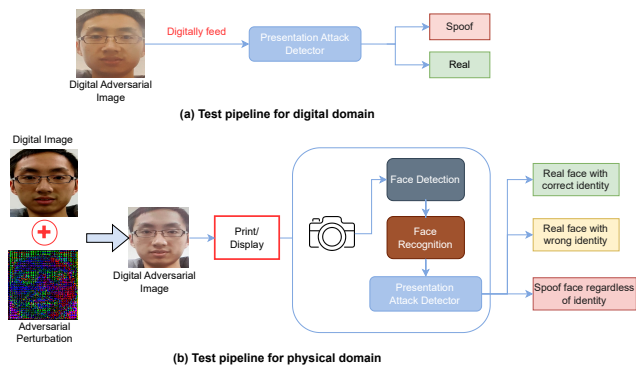


Figure 2. Experimental pipelines to evaluate the performance of the adversarial attacks. (a) shows the pipeline used when we attack a PAD in the digital domain, and (b) shows our testing pipeline in a physical domain. The digital image has to undergo two transformations and has to be effective after distortions are introduced in these processes.

these deep learning models and have recently emerged as a serious threat to face recognition systems. Adversarial examples are generated by adding perturbations to the input images, which are usually imperceptible to humans but can cause the model to make incorrect predictions. The majority of research on adversarial attacks [35, 40, 19] presumes that the attacker can directly input the digitally generated adversarial example into the machine learning model. Such attacks are typically referred to as digital domain attacks. However, this assumption does not hold in the case of anti-spoofing, where the system is designed to work in the physical world.

Adversarial attacks in the physical domain have gained significant attention in recent times due to their practicality and complexity. To attack the face anti-spoofing system in a physical world setting, the spoof image created by the attacker must be printed or displayed in the real world and then captured by the system’s camera. This process of converting digital images to physical and then back to digital is called image rebroadcast [1]. The changes made to the image during this rebroadcast process help the anti-spoofing detector to recognize that the digital image is fake by looking exactly for the spoofing artifacts introduced during the rebroadcast process and prevent unauthorized access to the system. As a new spoofing pattern may be introduced after the attack, adversarial attacks need to act in a pre-emptive manner. Therefore, it is challenging to create an adversarial example that can effectively attack an anti-spoofing system in a physical domain setting. We show the difference between a physical and digital domain attack in Figure 2.

After identifying the challenges associated with physical attacks, we present AdvGen, an automated method to create adversarial face images. AdvGen uses a Conditional Generative Adversarial Network to simulate presentation attacks and generate adversarial images that can fool

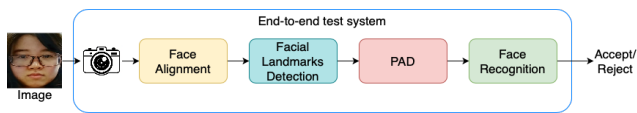


Figure 3. A typical Face authentication pipeline. Face PAD acts as a gatekeeper to face recognition module.

state-of-the-art PADs in a physical domain attack setting. Our proposed method, AdvGen, generates adversarial face images that mimic the process of physical presentation attacks, such as print and replay attacks. When a live image is passed through AdvGen, it simulates the printing and displaying process to create an adversarial image that retains the characteristics of a printed or displayed image but is classified as real when passed through a spoof classifier. Moreover, AdvGen ensures that the identity of the original face is preserved. The objective of AdvGen is to incorporate the properties of physical adversarial attacks into digital adversarial attacks. The contributions of the paper can be summarized as follows:

1. We design an identity preservation regularization term to enhance the identity preserving capability of a cycleGAN and name it IdGAN. IdGAN, given a real image, can generate a printed or replayed spoof version of it by preserving identity.
2. We propose AdvGen, a generative adversarial network trained to generate perturbations that are robust to distortions introduced to an image during physical transformations.
3. A systematic mathematical formulation for the problem of generation of adversarial physical perturbation and modeling it as the learning objective of a deep generative model.
4. We show that AdvGen is a more effective use of generating robust physical adversarial perturbations by comparing it against four datasets: SiW [57], MSU-MFSD [48], Replay-Attack [9] and OULU-NPU [5]. (Figure 1).

## 2. Related Works

**Adversarial Attacks** Many adversarial attack algorithms have indicated that deep learning models are broadly vulnerable to adversarial samples. For white-box attacks, where the attacker has complete knowledge of the target model, including its architecture and parameters, the gradient-based approaches [19, 8, 31, 13, 15, 6, 44] can be conducted by adding adversarial perturbations to the pixels of the original images, where all the perturbations are derived from the back-propagation gradients regarding the adversarial constraints. For black-box attacks, where the attacker has limited knowledge of the target model and must

make queries to the model to infer its behavior in order to craft an effective attack, one interesting direction is to utilize a substitute/surrogate model to perform transfer-based attacks. Recent works [59, 50, 14] claim that input diversity can further boost attack transferability. In the image classification domain, semi-whitebox approaches based on Generative Adversarial Networks (GANs) rely on softmax probabilities [49, 45, 39, 52]. Compared to digital attacks, physical attacks require much larger perturbation strengths to enhance the adversary’s resilience to various physical conditions such as lightness and object deformation [2, 51]. Min-max optimization problem and transferability phenomenon are being explored for adversarial training [6, 41]. These explorations focus mostly on the region around natural examples where the loss is (close to) linear.

**Generative Adversarial Networks (GANs)** Generative Adversarial Networks [18] are now being used in a wide variety of applications. These include image synthesis applications [36, 12], style transfer [42, 23, 17], image-to-image translation [20, 60], and representation learning [36, 37, 32]. Previous studies with GAN have shown that it is possible to generate high-resolution images up to  $1024 \times 1024$  resolution in various domains such as the human face, vehicles, and animals [25, 26]. In [19] proposes a Fast Gradient Sign Method (FGSM) to generate adversarial examples. It computes the gradient of the loss function with respect to pixels and moves a single step based on the sign of the gradient. While this method is fast, using only a single direction based on the linear approximation of the loss function often leads to sub-optimal results.

**Adversarial Attacks on Face Recognition** Current adversarial face synthesis methods include works by AdvFaces [10], which learns to perturb the salient regions of the face, unlike FGSM [19] and PGD [31], which perturbs every pixel in the image and image is generated by gradient-based methods. LatentHSJA [34] manipulates the latent vectors for fooling the classification model, and [56] which crafts replay-attack only to fool CNN-based face recognition system. Methods that rely on white-box manipulations of face recognition models are discussed first here. Bose et al. craft adversarial examples by solving constrained optimization such that a face detector cannot detect a face [4]. The adversarial eyeglasses can also be synthesized via generative networks [38]. But since these works are based on a white-box approach, it seems impractical in real-world scenarios. Dong et al. [15] proposed an evolutionary optimization method for generating adversarial faces in black-box settings. This method requires at least 1,000 queries to the target face recognition system before a realistic adversarial face can be synthesized. Song et al. [52] employed a conditional variation autoencoder GAN for crafting adversarial face images in a semi-whitebox setting. Here, they only focused on impersonation attacks and require at least five

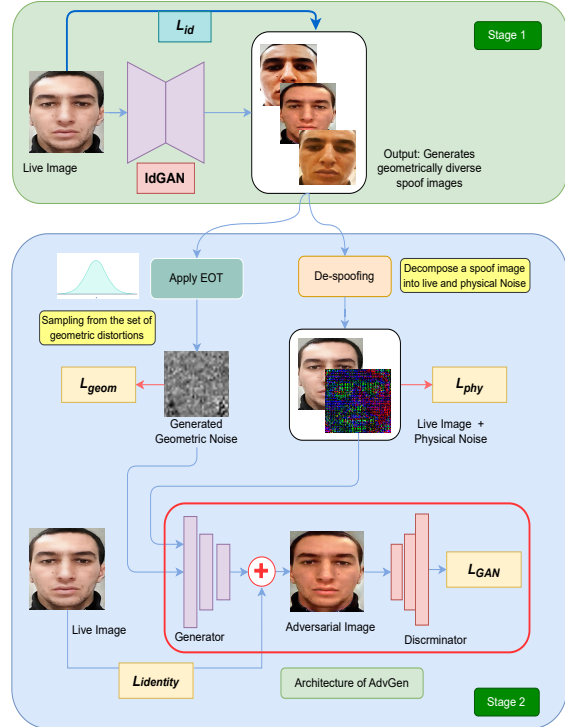


Figure 4. Synthesizing adversarial face images using AdvGen consists of two stages: **Stage 1:** Training of *IdGAN* which, given a live image, learns to generate geometrically diverse spoof images. These generated images produced by *IdGAN* simulate printing and replay. *Identity loss* is introduced as an identity regularizer to preserve the subject’s identity in the generated images. **Stage 2:** We apply de-spoofing and EOT on the generated spoof images to get the physical and geometric noises. These are fed into AdvGen’s generator to generate the adversarial perturbation. The generated image from AdvGen is robust to physical as well as geometric distortions.

images of the target subject for training and inference.

### 3. Methodology

AdvGen consists of three components i) a simulator network that emulates printing and replaying input images, ii) a decomposition network that can decompose spoof faces into noise signal and live faces, and iii) a generator network supervised using a formulated loss to generate physical adversarial perturbations.

We formulate the problem of generating a robust physical adversarial perturbation as an optimization objective in Section 3.1. Then we describe the architecture of the simulator network in Section 3.2. In Section 3.3, we elaborate on modeling the formulated optimization objective using a Generative Neural network.

#### 3.1. Problem Formulation

First, we formulate the creation of an adversarial image in the digital domain, and then we modify it to the physical

domain.

Let  $\mathcal{I}$  denote an input image and  $l_{true}$  its corresponding label. Let  $l_{target} \neq l_{true}$  be the target label of the attack. Let  $f(\cdot)$  denote the output of the target neural network. The process of generating an adversarial perturbation  $\delta$  involves solving the following optimization problem:

$$\begin{aligned} \arg \min_{\delta} \mathcal{L}(f(\mathcal{I} + \delta), l_{target}), \\ \text{subject to } \|\delta\|_p < \epsilon \end{aligned} \quad (1)$$

where  $\mathcal{L}(\cdot)$  is the neural network's loss function, and  $\|\cdot\|_p$  denotes the  $L_p$ -norm. To solve the above-constrained optimization problem efficiently, we reformulate it in the Lagrangian-relaxed form:

$$\arg \min_{\delta} \mathcal{L}(f(\mathcal{I} + \delta), l_{target}) + \lambda \|\delta\|_p \quad (2)$$

where  $\lambda$  is a hyper-parameter that controls the regularization of the distortion  $\|\delta\|_p$ .

In a physical domain setting, we denote a spoof image as  $\mathcal{I}_s$ . The spoof detection network is not fed directly with  $\mathcal{I}_{adv} = \mathcal{I}_s + \delta^*$  ( $\delta^*$  is the optimal digital perturbation obtained by using Eq. 2 with its physically recaptured version  $\mathcal{I}_r = \mathcal{P}(\mathcal{I}_{adv}) = \mathcal{P}(\mathcal{I}_s + \delta^*)$  where we use  $\mathcal{P}(\cdot)$  to denote the physical broadcasting and recapture procedure.  $\mathcal{P}(\cdot)$  is capable of destroying the effect of  $\rho^*$ ).

In order to ensure that the perturbation remains effective even after the image has been rebroadcasted, it is important to consider the possible transformations that the image may undergo during this process. This will allow us to create a robust perturbation that can withstand these transformations.  $\mathcal{T}$  denotes the set of all transformations in the physical process. Perturbation  $\rho$  can be obtained by optimizing the average loss over  $\mathcal{T}$ ,

$$\arg \min_{\rho} \mathbb{E}_{t \sim \mathcal{T}} [\mathcal{J}(f_s(t(\mathcal{I}) + \rho), l_{target})] + \lambda \|\rho\|_p \quad (3)$$

Here  $f_s$  denotes the output of a face presentation attack detector for a transformed image  $\mathcal{I}$  after applying a broadcasting transform  $t$  selected from a set of physical transforms  $\mathcal{T}$  and then applying a perturbation  $\rho$  obtained using Eq. 3.

### 3.2. Physical Simulator Network

We train *IdGAN*, an architecture derived from CycleGAN, to learn the simulation from real to spoof. This network learns to add physical and geometrical perturbations to an input image. It has two benefits: i) the simulated image will be useful in the next stage of attack generation, ii) This network is trained on data exposed to physical augmentations (rotation, random crop, resize, etc.), making the network capable of generating spoof images with physical variations.

**Generators:** The network consists of two generators  $\mathcal{G}_{rs}$

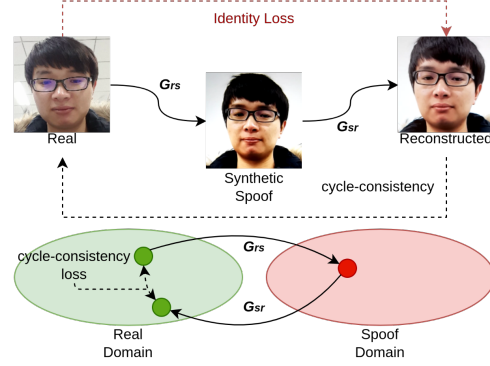


Figure 5. Loss terms used to train *IdGAN*. along with conventional  $\mathcal{L}_{adv}$  and  $\mathcal{L}_{cycle}$ , we introduce a  $\mathcal{L}_{id}$  to preserve identity in the generated image, which is a crucial step for the stage 2.

and  $\mathcal{G}_{sr}$ . Generators are based on Convolution based encoder-decoder architectures and generate a feature representation of the input image  $\mathcal{I}_r$ , and the decoder generates the corresponding presentation attack variants of the input  $\mathcal{I}_r$ . The discriminators  $\mathcal{D}_r$  and  $\mathcal{D}_s$  distinguish between the captured examples and the generated samples by the generators. The network is trained using three types of losses:

1. **Identity Regularizer:** The generated image should preserve the identity of the input. This would be a critical component in the adversarial attack generation. We introduce an identity-preserving regularization term to CycleGAN. The network, at every iteration, tries to preserve identity by minimizing the cosine similarity between the face embeddings of the generated image and the input image. The face embeddings are generated using a pretrained ArcFace [11]. The identity regularizer is defined as,

$$\begin{aligned} \mathcal{L}_{id}(\mathcal{G}_{rs}, \mathcal{G}_{sr}, \mathcal{I}_r, \mathcal{I}_s) = \mathbb{E}_x [1 - \mathcal{F}[\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)), \mathcal{I}_r]] \\ + \mathbb{E}_x [1 - \mathcal{F}[\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)), \mathcal{I}_s]] \end{aligned} \quad (4)$$

2. **Adversarial Loss:** Adversarial loss creates a 2-player adversary between the generator and discriminator, leading to better training through competition. An MSE-based adversarial loss is used and defined as,

$$\begin{aligned} \mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r) = \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)} \log[\mathcal{D}_s(\mathcal{I}_s)] + \\ \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)} \log[1 - \mathcal{D}_s(\mathcal{G}_{rs}(\mathcal{I}_r))] \\ \mathcal{L}_{adv}(\mathcal{G}_{sr}, \mathcal{D}_r, \mathcal{I}_s, \mathcal{D}_s) = \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)} \log[\mathcal{D}_r(\mathcal{I}_r)] + \\ \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)} \log[1 - \mathcal{D}_r(\mathcal{G}_{sr}(\mathcal{I}_s))] \\ \mathcal{L}_{adv} = \mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r) + \\ \mathcal{L}_{adv}(\mathcal{G}_{sr}, \mathcal{D}_r, \mathcal{I}_s, \mathcal{D}_s) \end{aligned} \quad (5)$$

3. **Cycle Consistency Loss:** Adversarial loss leaves the learning unconstrained. Hence the Cycle Consistency

Loss is added as a regularization term to the generator’s objectives shown in Figure 5. This loss is defined as,

$$\begin{aligned}\mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) &= \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)} [\|\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)) - \mathcal{I}_r\|_1] \\ \mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s) &= \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)} [\|\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)) - \mathcal{I}_s\|_1] \\ \mathcal{L}_{cycle} &= \mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) + \mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s)\end{aligned}\quad (6)$$

Here  $\|\cdot\|_1$  denotes  $\mathcal{L}_1$  norm

Finally, IdGAN is trained using the following objective,

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_{cycle} \times \mathcal{L}_{cycle} + \lambda_{id} \times \mathcal{L}_{id} \quad (7)$$

### 3.3. Modelling the Physical Transformation

A real image  $\mathcal{I}$  undergoes physical transformations such as color distortion and display, printing, and imaging artifacts to become a spoof image [24]. In addition, the presenter may wish to introduce geometric distortions like rotation, capture distance, folding the presentation medium, etc. These distortions need to be carefully modeled. To generate the perturbation, we use a generative neural network to model the optimization problem. AdvGen is optimized over the formulated loss. Figure 4 outlines the proposed architecture. AdvGen consists of a generator  $\mathcal{G}$ , a discriminator  $\mathcal{D}$ , a spoof noise synthesiser  $\mathcal{S}$  and a geometric distortion sampler  $\mathcal{F}$ . Together these modules model every necessary component in the formulated objective.

**Generator** The generator  $\mathcal{G}$  of AdvGen takes in an input image  $x \in \mathcal{X}$  and generates a perturbation  $\mathcal{G}(x)$ . In order to maintain the original visual quality of the input image and avoid generating a completely new face image, the generator produces an additive perturbation that is applied to the input image as  $x + \mathcal{G}(x)$ . The generator’s loss has the following components:

**Physical Perturbation Hinge Loss:** To generate perturbations that include physical distortions, we use a pretrained noise decomposition network [24]. It is in the synthesized spoof image from AdvGen, and returns decomposed physical noise and live faces. This synthesized noise serves as the perturbation to be added to the real image. This noise is an unbounded physical noise. Hence we introduce this noise to the generation pipeline using a soft hinge loss on the  $\mathcal{L}_2$  norm bounding the amount of physical noise introduced by [8, 29] formulated as:

$$\mathcal{L}_{phy} = \mathbb{E}_x [\max(\epsilon_1, \|\mathcal{P}hy(x)\|_2)] \quad (8)$$

$\epsilon_1$  is a user-specific bound on the added perturbation and  $\mathcal{P}hy(\cdot)$  denotes physical noise from the decomposition

network.

**Geometric Distortion Hinge Loss:** Presentation of a physical medium is always subject to geometric distortions such as rotation, zooming, folding, etc., due to human errors. To make the attack robust to geometric distortions, AdvGen is trained with geometric augmentations to generate spoof images with diverse geometric variations. To model these distortions, Expectation over Transforms(EOT) [2] is applied over the generated spoof images. Modeling these transformations diversifies the set of physical transforms modeled by the generator. The generated geometric perturbation is controlled using a geometric hinge loss

$$\mathcal{L}_{geom} = \mathbb{E}_x [\max(\epsilon_2, \|\mathcal{G}eom\|_2)] \quad (9)$$

$\epsilon_2$  is a user-specific bound on the added perturbation and  $\mathcal{G}eom(\cdot)$  denotes geometric perturbation obtained from EOT.

**Identity Regularizer Loss:** The perturbation must preserve the identity of the target. We introduce an identity regularizer to the generator loss, which maximizes the cosine similarity between the identity embeddings obtained from a pretrained ArcFace [11] matcher. We define it as,

$$\mathcal{L}_{identity} = \mathbb{E}_x [1 - \mathcal{F}(x, x + \mathcal{G}(x))] \quad (10)$$

**Discriminator:** We introduce a discriminator  $\mathcal{D}$  which distinguishes between the generated samples  $x + \mathcal{G}(x)$  and the corresponding real sample  $x$ . This Discriminator is based on PatchGAN and projects the input to a patch-based matrix where each value in the matrix corresponds to the score of the particular patch’s discriminative score. trained using the adversarial loss:

$$\mathcal{L}_{GAN} = \mathbb{E}_x [\log \mathcal{D}(x)] + \mathbb{E}_x [\log(1 - \mathcal{D}(x + \mathcal{G}(x)))] \quad (11)$$

AdvGen is trained to generate identity-preserving physical perturbation in an end-to-end on the following objective:

$$\begin{aligned}\mathcal{L} &= \lambda_{phy} \times \mathcal{L}_{phy} + \lambda_{geom} \times \mathcal{L}_{geom} + \\ &\lambda_{identity} \times \mathcal{L}_{identity} + \lambda_{GAN} \times \mathcal{L}_{GAN}\end{aligned}\quad (12)$$

## 4. Experiments

In this section, we first introduce the datasets used and the experimental setup. Then we evaluate the performance of our framework in different settings and explain the evaluation metrics:

### 4.1. Datasets and Baselines

We train AdvGen on OULU-NPU [5] and test on SiW [57], MSU-MFSD [48], Replay-Attack [9] and

<i>Attack Success Rate on OULU-NPU(%) and SSIM after attack</i>					
	BIM [28]	EOT [2]	$RP_2$ [16]	D2P [21]	Ours
CDCN [55]	41.19	55.82	63.12	68.37	<b>81.02</b>
CDCNpp [58]	37.47	51.61	59.39	64.26	<b>78.22</b>
C-CDN [54]	38.38	51.58	60.83	65.49	<b>79.34</b>
DC-CDN [54]	39.95	53.83	61.36	66.03	<b>80.55</b>
SSAN-M [47]	40.06	52.02	61.40	65.27	<b>80.42</b>
SSAN-R [47]	34.54	49.83	57.03	61.79	<b>75.15</b>
DBMNet [22]	38.78	52.69	59.89	62.74	<b>79.63</b>
STDN [30]	40.92	53.93	61.67	63.29	<b>80.98</b>
Meta-FAS [7]	35.38	47.67	57.25	59.53	<b>76.19</b>
De-Spoofing [24]	46.44	58.43	65.41	68.66	<b>84.67</b>
<b>SSIM in [0,1]</b>	<b>0.64</b>	<b>0.38</b>	<b>00.32</b>	<b>0.45</b>	<b>0.98</b>

Table 1. Comparison of attack success rates on different models and ours using four different datasets.

OULU-NPU [5]<sup>1</sup> datasets. **OULU-NPU [5]** face presentation attack detection database contains 4,950 real access and attack videos belonging to 55 different subjects. **SiW [57]** contains 4,478 15s long videos for 165 subjects. For each subject, there are eight live and up to 20 spoof videos. **MSU-MFSD [48]** contains 280 video recordings of genuine and attack faces for 35 individuals. **Replay-Attack [9]** consists of 1300 video clips of photo and video attacks for 50 clients under different lighting conditions.

We compare our proposed method with four state-of-the-art physical attack generation methods BIM [28], EOT [2],  $RP_2$  [16], D2P [21]. To compare our method’s effectiveness in the physical vs. digital domain, we implement four standard digital adversarial attacks FGSM [19], PGD [31], BIM [28], and Carlini & Wagner [8]. We use TorchAttack’s [27] implementations of the above methods by perturbing the necessary parameters to generate effective attacks. To establish the effectiveness and generalizability of our proposed attack across different spoof detection models, we compare the ASR of our generated images from OULU-NPU across ten state-of-the-art face anti-spoofing models in Table 1.

## 4.2. Evaluation Metrics

By comparing our network against state-of-the-art baselines, we quantify the adversarial attacks’ effectiveness via i) attack success rate (ASR) and ii) structural similarity (SSIM) [46].

The attack success rate (ASR) is computed as

$$ASR = \frac{\text{No. of attacks classified as real}}{\text{Total number of attacks}} \times 100\% \quad (13)$$

<sup>1</sup>We train on training and validations sets of Protocol 1 of OULU-NPU and test on the corresponding test set

To quantify the effectiveness of the generated adversarial images with the input image, we compute the Structural Similarity Index (SSIM) metric calculated between the adversarial image and the real image as proposed in research[46]:

## 4.3. Experimental Setup

All experiments are conducted on print and replay attack scenarios. We use an HP Smart Tank 580 printer to print all the images. For display, we use two mediums, MacBook Pro (Intel Iris Plus Graphics 640 1536 MB) and Redmi K20 pro (Super AMOLED, HDR10 display). All images are captured from a distance ranging from 20cm to 40cm.

To validate the effectiveness of our developed attack method, we deploy four state-of-the-art face anti-spoofing methods to a streamlit app. The app takes a real-time feed and returns the predicted identity of the person along with spoof/live prediction along with its confidence.

We create a test set of 300 images per dataset comprising different identities. From OULU-NPU, we sample 20 identities; from SiW, we sample 50 identities; from REPLAY-ATTACK, we sample 15 identities; from MSU-MFSD, we sample 15 identities. The sampled images are manually handpicked to ensure that maximum diversity is covered in terms of variations. To validate results for EOT, we manually perform physical distortions like rotation on the print and replay displays, change of brightness in the replay attacks, and folding the presentation medium in print attacks.

## 4.4. Experimental Settings

We use ADAM optimizers with  $\beta_1 = 0.5$  and  $\beta_2 = 0.9$ . Each mini-batch consists of 1 face image. We train AdvGen for 100 epochs with a fixed learning rate of 0.0002.

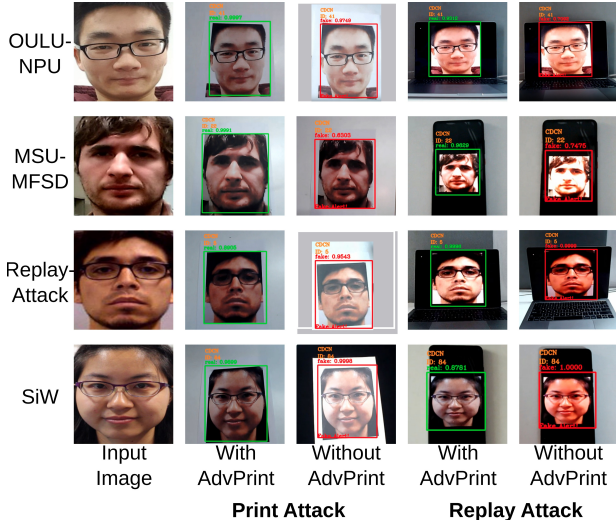


Figure 6. Experimental pipelines to evaluate the performance of the attacks. (a) shows the pipeline used when we attack a PAD in the digital domain, and (b) shows our testing pipeline in a physical world setting.

We also use identity loss with parameters  $\lambda_i = 1.0$ . We train two separate models for print and video-replay attacks. A unified model for both attacks is also trained with the same hyperparameters. We iteratively perform FGSM over AdvGen with  $\epsilon = 0.1$ . All experiments are conducted using PyTorch.

## 5. Results and Analysis

### 5.1. Effectiveness in Physical Domain

	Attack Success Rate (%)	
	Digital Domain	Physical Domain
BIM [28]	98.04	41.22
FGSM [19]	75.32	23.13
GA	79.56	26.92
IGSA	100.00	34.22
IGA	99.64	31.48
PGD [31]	98.63	36.42
<b>AdvGen</b>	<b>100</b>	<b>81.02</b>

Table 2. Performance of state-of-the-art adversarial attack methods in the digital and physical domain.

To evaluate the effectiveness of the proposed method in the physical domain, we perform a digital attack using conventional attack strategies and our method on the test set of 300 images curated from OULU-NPU. Then the adversarial images are printed and presented physically to a presentation attack detector. The performance of all attacks is optimal in the digital domain but significantly drops when

transferred to the physical domain, as demonstrated in Table 2. The ASR of the standard methods is less than 50 in the physical domain, while our method clearly outperforms these values. These empirical results clearly demonstrate that including physical spoofing noise makes the attack robust to transformations incurred through physical processes.

### 5.2. Comparison Studies

In Table 1, we present the findings from our comparative studies against state-of-the-art physical adversarial attack methods. Compared to the state-of-the-art methods, our method is significantly better at generating robust attacks in terms of achieved ASR. In terms of structural similarity, our method stands out in preserving visual information in the generated image and outperforms the other methods. Our method learns to generate imperceptible noise signals at locations on the face that are not significant for identity recognition. BIM [28] iteratively generates perturbations on the input image, hence preserving visual features to some extent, but the ASR on the generated images is low because of its inability to model physical perturbations. Attack images generated using EOT,  $RP_2$ , and D2P have higher ASR by virtue of their design to address generic physical distortions in their noise modeling. They are able to generate physically robust attacks as compared to BIM, but these are not specifically physical perturbations introduced on a face image due to physical transformations like printing or display on a screen. Our method models this noise and hence is better at modeling.

### 5.3. Effectiveness with Geometric Distortions

In physical presentations, geometric distortions like capturing viewpoint, rotation, scaling, and perspective changes of the display medium and folding of the printed medium are unavoidable. Being trained on distortions sampled by Expectation Over Transformation (EOT) [2], our method is robust to geometric distortions like viewpoint changes, rotation, and brightness. Figure 8 demonstrates the effectiveness of our methods through various geometric distortions.

### 5.4. Ablation Study

AdvGen is trained using four loss terms, each contributing to one component to be added to the generated perturbation. To analyze the importance of each module, we train four variants of AdvGen for comparison by dropping  $\mathcal{L}_{phy}$ ,  $\mathcal{L}_{geom}$ ,  $\mathcal{L}_{identity}$  and  $\mathcal{L}_{GAN}$  and show results in Figure 7. Without a discriminator, i.e., with  $\mathcal{L}_{GAN}$ , the visual quality of generated images is affected, and undesirable artifacts are introduced. Without a physical perturbation hinge  $\mathcal{L}_{phy}$ , the generated perturbation is not robust enough to physical transformation and gets classified as a "spoof." Perturbations generated without being regulated by any geometric distortion  $\mathcal{L}_{geom}$  fail even when even a small geomet-

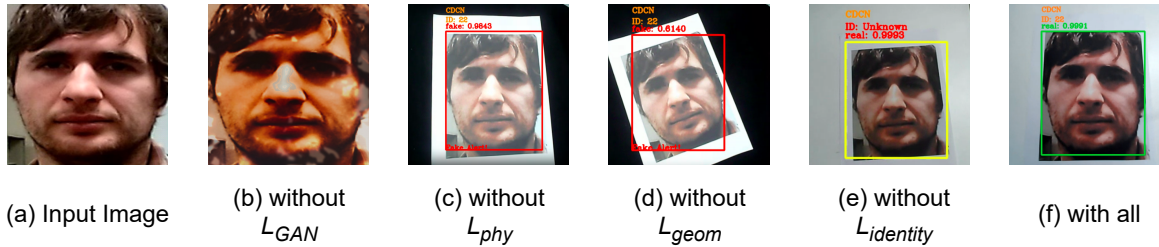


Figure 7. Variants of AdvGen trained without GAN loss, physical perturbation hinge loss, geometric distortion hinge loss, and identity loss, respectively.

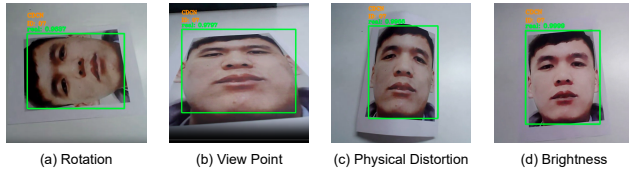


Figure 8. Effectiveness of AdvGen after applying geometric distortions. Adversarial image is classified as real (a) after rotation, (b) changing the viewpoint of the camera, (c) applying physical distortions, like folding the image, and (d) changing the brightness level of the setup.

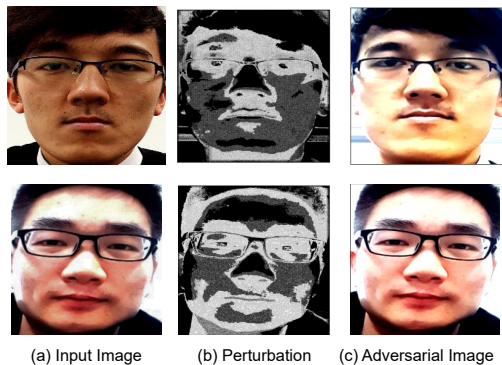


Figure 9. Visualization of the generated perturbation. (a) shows the input image, which can be live or spoof, (b) the locations of the input face resulting in perturbation we get from AdvGen, and (c) shows the final adversarial image.

ric distortion is performed. Without an identity regularizer, though, the generated perturbation is robust for a presentation attack generator but fails to pass the identity check. The generated perturbation by such a generator perturbs the identity. We conclude that to generate a perceptually realistic and robust perturbation, every component is necessary.

## 6. Future Works

Focusing on the print and reply attack scenario, we proposed AdvGen, which generates adversarial images to fool a face PAD. Below, we list a few points that we would like to pursue in the future:

1. Extending our attack to a scenario in which the attack is carried out by showing a 3D and paper mask, make-

up, mannequin, etc., of the adversarial example to the authentication system.

2. From the defender’s side, future research has to be performed to recover robustness against anti-spoofing and design new CNN-based face authentication systems capable of working in the presence of adversarial spoofing attacks.
3. Having demonstrated the threats posed by replay and print attacks exploiting adversarial examples, we plan to propose a defense for such attacks. We will create a system that would be capable of working in the presence of such adversarial print and replay images.

## 7. Conclusion

In this paper, we have created a physical attack on a CNN-based face authentication system that has an anti-spoofing module. We demonstrate that attacking an anti-spoofing face authentication system in the physical domain is more challenging and comes with additional difficulties than attacking systems in other application scenarios. Our new framework, called AdvGen, can produce adversarial images that mimic a printing and replay procedure. Through experimentation, we have demonstrated that AdvGen can generate synthetic adversarial prints that are capable of bypassing the Presentation Attack Detectors (PADs) and fooling a face recognition system, all while maintaining the subject’s identity.

## References

- [1] S. Agarwal, W. Fan, and H. Farid. A diverse large-scale dataset for evaluating rebroadcast attacks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1997–2001. IEEE, 2018. 2
- [2] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*, pages 284–293. PMLR, 2018. 3, 5, 6, 7
- [3] Y. Atoum, Y. Liu, A. Jourabloo, and X. Liu. Face anti-spoofing using patch and depth-based cnns. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 319–328. IEEE, 2017. 1



- [4] A. J. Bose and P. Aarabi. Adversarial attacks on face detectors using neural net based constrained optimization. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSp)*, pages 1–6. IEEE, 2018. 3
- [5] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid. Oulu-npu: A mobile face presentation attack database with real-world variations. In *2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017)*, pages 612–618. IEEE, 2017. 2, 5, 6
- [6] W. Brendel, J. Rauber, and M. Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017. 2, 3
- [7] R. Cai, Z. Li, R. Wan, H. Li, Y. Hu, and A. C. Kot. Learning meta pattern for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 17:1201–1213, 2022. 6
- [8] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. Ieee, 2017. 2, 5, 6
- [9] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. In *2012 BIOSIG-proceedings of the international conference of biometrics special interest group (BIOSIG)*, pages 1–7. IEEE, 2012. 2, 5, 6
- [10] D. Deb, J. Zhang, and A. K. Jain. Advfaces: Adversarial face synthesis. In *2020 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10. IEEE, 2020. 3
- [11] 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*, pages 4690–4699, 2019. 4, 5
- [12] E. L. Denton, S. Chintala, R. Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in neural information processing systems*, 28, 2015. 3
- [13] 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*, pages 9185–9193, 2018. 1, 2
- [14] Y. Dong, T. Pang, H. Su, and J. Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4312–4321, 2019. 3
- [15] 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*, pages 7714–7722, 2019. 2, 3
- [16] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1625–1634, 2018. 6
- [17] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 3
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3
- [19] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2, 3, 6, 7
- [20] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 3
- [21] S. T. Jan, J. Messou, Y.-C. Lin, J.-B. Huang, and G. Wang. Connecting the digital and physical world: Improving the robustness of adversarial attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 962–969, 2019. 6
- [22] Y. Jia, J. Zhang, and S. Shan. Dual-branch meta-learning network with distribution alignment for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 17:138–151, 2021. 6
- [23] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 3
- [24] A. Jourabloo, Y. Liu, and X. Liu. Face de-spoofing: Anti-spoofing via noise modeling. In *Proceedings of the European conference on computer vision (ECCV)*, pages 290–306, 2018. 5, 6
- [25] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 3
- [26] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 3
- [27] H. Kim. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020. 6
- [28] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 6, 7
- [29] Y. Liu, X. Chen, C. Liu, and D. Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016. 5
- [30] Y. Liu, J. Stehouwer, and X. Liu. On disentangling spoof trace for generic face anti-spoofing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 406–422. Springer, 2020. 6
- [31] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 2, 3, 6, 7
- [32] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun. Disentangling factors of variation in deep representation using adversarial training. *Advances in neural information processing systems*, 29, 2016. 3

- [33] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017. 1
- [34] D. Na, S. Ji, and J. Kim. Unrestricted black-box adversarial attack using gan with limited queries. *arXiv preprint arXiv:2208.11613*, 2022. 3
- [35] 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)*, pages 372–387. IEEE, 2016. 2
- [36] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 3
- [37] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 3
- [38] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter. A general framework for adversarial examples with objectives. *ACM Transactions on Privacy and Security (TOPS)*, 22(3):1–30, 2019. 3
- [39] Y. Song, R. Shu, N. Kushman, and S. Ermon. Constructing unrestricted adversarial examples with generative models. *Advances in Neural Information Processing Systems*, 31, 2018. 3
- [40] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1, 2
- [41] F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017. 3
- [42] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. *arXiv preprint arXiv:1603.03417*, 2016. 3
- [43] P. Wang, W.-H. Lin, K.-M. Chao, and C.-C. Lo. A face-recognition approach using deep reinforcement learning approach for user authentication. In *2017 IEEE 14th International Conference on e-Business Engineering (ICEBE)*, pages 183–188, 2017. 1
- [44] X. Wang and K. He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1924–1933, 2021. 2
- [45] X. Wang, K. He, C. Song, L. Wang, and J. E. Hopcroft. Atgan: An adversarial generator model for non-constrained adversarial examples. *arXiv preprint arXiv:1904.07793*, 2019. 3
- [46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6
- [47] Z. Wang, Z. Wang, Z. Yu, W. Deng, J. Li, S. Li, and Z. Wang. Domain generalization via shuffled style assembly for face anti-spoofing. In *CVPR*, 2022. 6
- [48] D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4):746–761, 2015. 2, 5, 6
- [49] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018. 3
- [50] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019. 3
- [51] K. Xu, G. Zhang, S. Liu, Q. Fan, M. Sun, H. Chen, P.-Y. Chen, Y. Wang, and X. Lin. Adversarial t-shirt! evading person detectors in a physical world. In *European conference on computer vision*, pages 665–681. Springer, 2020. 3
- [52] L. Yang, Q. Song, and Y. Wu. Attacks on state-of-the-art face recognition using attentional adversarial attack generative network. *Multimedia tools and applications*, 80(1):855–875, 2021. 3
- [53] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, and G. Zhao. Deep learning for face anti-spoofing: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1
- [54] Z. Yu, Y. Qin, H. Zhao, X. Li, and G. Zhao. Dual-cross central difference network for face anti-spoofing. *arXiv preprint arXiv:2105.01290*, 2021. 6
- [55] Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, F. Zhou, and G. Zhao. Searching central difference convolutional networks for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5295–5305, 2020. 1, 6
- [56] B. Zhang, B. Tondi, and M. Barni. Adversarial examples for replay attacks against cnn-based face recognition with anti-spoofing capability. *Computer Vision and Image Understanding*, 197:102988, 2020. 3
- [57] S. Zhang, X. Wang, A. Liu, C. Zhao, J. Wan, S. Escalera, H. Shi, Z. Wang, and S. Z. Li. A dataset and benchmark for large-scale multi-modal face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 919–928, 2019. 2, 5, 6
- [58] Y. Zhang, Z. Yin, J. Shao, Z. Liu, S. Yang, Y. Xiong, W. Xia, Y. Xu, M. Luo, J. Liu, et al. Celeba-spoof challenge 2020 on face anti-spoofing: Methods and results. *arXiv preprint arXiv:2102.12642*, 2021. 6
- [59] Y. Zhong and W. Deng. Towards transferable adversarial attack against deep face recognition. *IEEE Transactions on Information Forensics and Security*, 16:1452–1466, 2020. 3
- [60] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 3