Physical Adversarial Attacks on Face Presentation Attack Detection Systems

Thesis submitted in partial fulfillment
of the requirements for the degree of

*Master of Science in **Computer Science and Engineering** by Research*

by

SAI AMRIT PATNAIK
2020701026
sai.patnaik@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
February 2024

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled **"Physical Adversarial Attacks on Face Presentation Attack Detection Systems"** by Sai Amrit Patnaik, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Dr. Anoop Namboodiri

To my family and friends

Acknowledgments

I feel immense happiness in compiling my entire research journey into this thesis. From having no research clue to writing and publishing papers, the journey of research at IIITH has been one of the most memorable phases of my life. I would first like to express my heartfelt thanks to my research advisor, Dr. Anoop Namboodiri, for his support and guidance in shaping my research career. I am grateful to Dr. Anoop for his constant motivation and direction in every phase of my research, which have helped me grow as an individual and an academic.

I would also like to acknowledge and thank all my friends for making my college experience memorable and wonderful, even during the tough times of the pandemic. My sincere appreciation to my co-author, Shivali Chansoriya, for her participation throughout the whole progress of this academic exploration with remarkable efforts and help. I would also like to mention Praguna, Sarath Sivaprasad, Sindhu Hegde, Mounika Kalakanti, Ekta Gavas, Apoorva Srivastava, Shaily Mishra for their help and support and let me learn different and new information in their fields.

I am also grateful to my Yuktahar mess and people (Siddhant Bansal, Chirag Jain, Mangal, Praguna, Ayappa, Raghav Mittal, Arhant Jain and many more) whom I met in Yuktahar for having cheerful and fruitful discussions over the food table.

I want to give a big thank you to my sister, Nidhi Goyal. She was there for me all through my master's degree, offering help and support. We had many long talks about writing research papers and doing research in general, which made me a better researcher. Whenever I faced difficult situations, she encouraged me and kept me grounded and stable with her pep talks. And on top of all that, she cooked delicious meals for me when I didn't want to eat mess food. I'm so thankful to her for introducing me to Shiv Baba.

This journey would not have been possible without the love and blessings of my dearest family members. Their support and guidance have always been one of the most significant factors in all my achievements in life.

To my Supreme father, Shivbaba, Words cannot adequately express the gratitude I feel towards you, the vital force of my life. I would like to dedicate my deepest emotions to thank you for the companionship, empowerment, blessings, inspiration, enlightenment, and positive energy you have brought into my life. You hold a special place in my soul, and I dedicate this thesis to you.

Abstract

In the realm of biometric security, face recognition technology plays an increasingly pivotal role. However, as its adoption grows, so does the need to safeguard it against adversarial attacks. Attacks involve presenting images of a person printed on a medium or displayed on a screen. Detection of such attacks relies on identifying artifacts introduced in the image during the printing or display and capture process. Adversarial Attacks try to deceive the learning strategy of a recognition system using slight modifications to the captured image. Evaluating the risk level of adversarial images is essential for safely deploying face authentication models in the real world. Among these, physical adversarial attacks present a particularly insidious threat to face antispoofing systems. Popular approaches for physical-world attacks, such as print or replay attacks, suffer from some limitations, like including physical and geometrical artifacts. The presence of a physical process (printing and capture) between the image generation and the PAD module makes traditional adversarial attacks non-viable. 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 thesis delves into the intriguing domain of physical adversarial attacks on face antispoofing systems, aiming to expose their vulnerabilities and implications. Our research unveils novel methodologies using white box and black box approaches to craft adversarial inputs capable of deceiving even the most robust face antispoofing systems. Unlike traditional adversarial attacks that manipulate digital inputs, our approach operates in the physical domain, where printed images and replayed videos are utilized to mimic real-world presentation attacks. By dissecting and understanding the vulnerabilities inherent in face antispoofing systems, we can develop more resilient defenses, contributing to the security of biometric authentication in an increasingly interconnected world. This thesis not only highlights the pressing need to address these vulnerabilities but also motivates towards a pioneering approach by exploring simple yet effective attack strategy to advancing the state of the art in face antispoofing security.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In a world where faces unlock secrets and identities of an individual, automatic face recognition systems stand as the gatekeepers of truth and security. In many spiritual traditions, the face is considered a sacred vessel that carries the imprint of our experiences, emotions, and spiritual journey. It is believed that the face can convey a person's true nature, revealing their inner beauty, wisdom, and even their spiritual advancement. Our thoughts, emotions, and intentions shape our inner being. They are believed to leave an imprint on our faces. Hence, our facial expressions, lines, and radiance are seen as manifestations of the state of our soul.

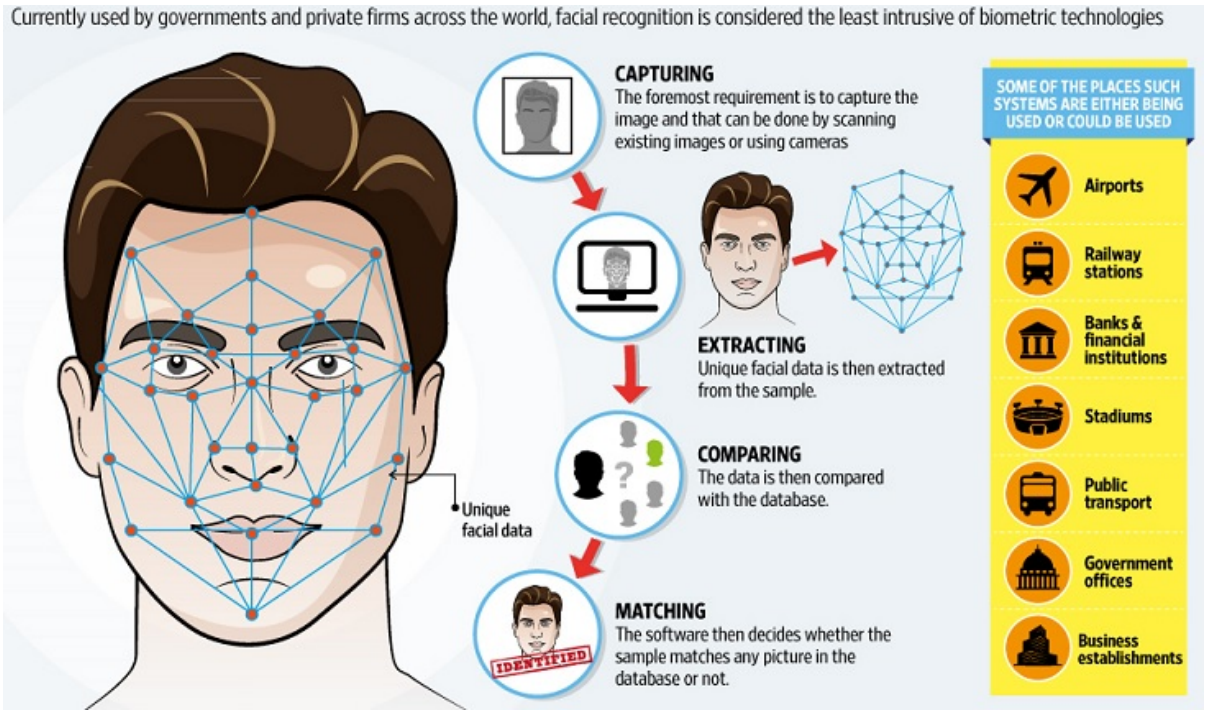


Figure 1.1: Automatic Face Recognition Systems

Automatic face recognition systems are powerful technologies that can identify and verify individuals by analyzing their facial features. These systems have become increasingly popular and are used in various applications, such as security, law enforcement, and personal device authentication. One of the most notable uses is in the field of security and access control. Airports, for instance, employ face recognition technology to match travelers' faces against databases of known individuals, enhancing security and streamlining the boarding process. Law enforcement agencies also leverage the power of automatic face recognition in their investigative efforts. By comparing faces captured in surveillance footage or photographs with databases of known individuals, these systems aid in identifying suspects, locating missing persons, and solving crimes. Moreover, automatic face recognition has found its way into our everyday lives through personal devices like smartphones and tablets. Many of these devices incorporate facial recognition as a secure and convenient means of unlocking the device or accessing sensitive information. By analyzing the unique features of the user's face, these systems ensure that only authorized individuals can gain access, safeguarding personal data and enhancing privacy. In addition to security and personal devices, face recognition systems have impacted the areas of marketing and retail. Retailers employ these technologies to analyze customer demographics, understand shopping patterns, and offer personalized experiences. By studying facial features and expressions, businesses can tailor their marketing strategies and improve customer engagement. Furthermore, automatic face recognition has become a common feature in social media platforms. Users can upload photos, and the system suggests tags by recognizing and identifying individuals present in the images. This simplifies the process of connecting with friends and family and enhances the overall social media experience. Additionally, face recognition algorithms are used to create entertaining filters and augmented reality effects, adding fun and creativity to online interactions. The basic idea behind automatic face recognition is to capture an image or video of a person's face and analyze it to extract unique characteristics that differentiate them from others. These characteristics include the distance between the eyes, the shape of the nose, and the contours of the face, which are then transformed into mathematical algorithms that serve as a digital representation of the individual's face. To recognize a face, the system compares the captured facial representation, called the probe face representation, with a database of known faces known as gallery images. This comparison involves complex algorithms that measure the similarity between the captured image and the stored facial templates.

1.2 Automated Face Recognition (AFR) Pipeline

The goal of all automatic face recognition (AFR) systems is to compute the similarity between two face images. A reliable AFR system aims to produce a high similarity measurement when comparing images of the same person (known as a genuine pair) and a low similarity measurement when comparing images of different individuals (known as impostor pairs). Achieving this involves multiple components within the system, each playing a crucial role in determining the final face similarity measurement and, consequently, the accuracy of the face recognition results. A typical AFR system consists of the following

modules: (i) face detection, (ii) face alignment, (iii) face liveness Detection, (iv) face feature extraction and face representation learning, (v) similarity matching.

- **Face Detection:** The first step is to detect and localize faces within an image or video frame. Face detection algorithms analyze the input data and identify potential face regions, typically marking them with bounding boxes or facial landmarks. The task is trivial for humans but poses significant challenges to machines. The complexity of this task arises because of the interplay of various factors, including diverse lighting conditions, poses, occlusions, and varying facial expressions. These algorithms scan an image pixel by pixel, looking for patterns that resemble facial features. They focus on characteristics like color, contrast, and texture to distinguish between facial and non-facial regions. Some algorithms employ machine learning techniques like convolutional neural networks, where the model is trained with a large set of labeled images to recognize key facial traits. Once a face or multiple faces are detected, the algorithm typically creates a bounding box around each face to indicate its location. This output enables subsequent tasks in the face recognition pipeline.

- **Face Alignment:** Faces in images or videos can appear in different pose, illumination, and expression due to factors like camera angles, head tilts, and facial expressions. Face alignment ensures that all faces are presented in a standardized and consistent manner. These algorithms utilize facial landmarks, which are specific points on the face, like the corners of the eyes, the tip of the nose, and the edges of the mouth. By identifying these landmarks within the detected face, the algorithm calculates the necessary adjustments needed to align the face. This may involve rotating, scaling, and translating the face to fit a predefined template. Once the alignment is complete, the face is presented in a standardized pose, allowing subsequent recognition algorithms to work with a consistent face representation. This standardization is essential as it minimizes variations caused by differences in pose, lighting, and facial expressions, ensuring accurate and reliable subsequent analysis.

- **Face Liveness Detection:** Face antispoofing is a sophisticated defense mechanism in face recognition systems to avoid fraudulent activities. After the initial stages of detecting and aligning faces, face antispoofing distinguishes between authentic human faces and deceptive tactical inputs like using photos, videos, or masks to trick the system. This crucial phase adds a layer of protection, ensuring that the system only grants access to legitimate individuals and maintains the integrity of the face recognition process. Face antispoofing involves algorithms that examine specific visual cues to differentiate between real and fake faces. These algorithms analyze elements such as texture, movement, and depth information. For instance, they can identify the lack of natural facial movements that occur in living individuals, such as blinking or micro-expressions. These movements are generally absent in static images or impersonations. These algorithms are trained on extensive datasets that equip the algorithms with the ability to recognize intricate patterns that set genuine faces apart from manipulated ones. By integrating face antispoofing into the face

recognition pipeline, the technology ensures that only actual individuals can access the system, significantly bolstering security and bolstering the credibility of the overall process.

- **Feature Extraction and Encoding:** In this stage, relevant facial features are extracted from the preprocessed face images. Feature extraction algorithms analyze the unique characteristics of the face, such as texture, shape, or appearance, and transform them into a compact and representative feature vector. Popular techniques for feature extraction include Local Binary Patterns (LBP), Scale-Invariant Feature Transform (SIFT), or deep Convolutional Neural Networks (CNN). Essentially, feature extraction breaks down the complex face structure into disciminative numerical values, which computers can easily process. The extracted facial features are then encoded into a format suitable for efficient comparison and matching. Encoding methods may involve reducing the dimensionality of the feature vector, applying mathematical transformations, or encoding it into a compact representation, such as a face template or a feature descriptor. The encoded features retain the essential discriminative information required for accurate matching while minimizing computational complexity.

- **Similarity Matching:** When a query face is presented, the face recognition system compares its feature vector or encoded representation with the computed feature vectors. Similarity matching algorithms calculate the similarity or distance between the query face and each stored face representation, determining the most similar or matching faces. Common distance metrics used for matching include Euclidean distance, cosine similarity, or Mahalanobis distance. Cosine Similarity is most commonly used in recent literature.

1.3 Vulnerabilities in Face Biometrics

In the thesis we concentrate on Presentation Attacks, i.e., attacks against the sensor of a FRS [28] (point V1 in Figure 1.2). Indirect attacks (corresponding to points V2-V7 in Fig. 1.2) target the internal components of the Face Recognition System (FRS), including the preprocessing module, feature extractor, classifier, and enrolling database. For a comprehensive definition of indirect attacks in face systems, please refer to [35]. Enhancing specific aspects of the FRS, such as communication channels, equipment, infrastructure, and perimeter security, can help mitigate indirect attacks [62]. The strategies required to bolster these modules align more with traditional cybersecurity practices rather than biometric techniques. While these attacks and their countermeasures fall outside the scope of this thesis, their significance should not be underestimated.

Presentation attacks represent a unique biometric vulnerability that requires dedicated countermeasures, distinct from other IT security solutions. In these attacks, perpetrators employ various artifacts, often artificial in nature (such as a facial photo, mask, synthetic fingerprint, or printed iris image), or attempt to mimic the genuine characteristics of authorized users (like gait, signature, or facial expression

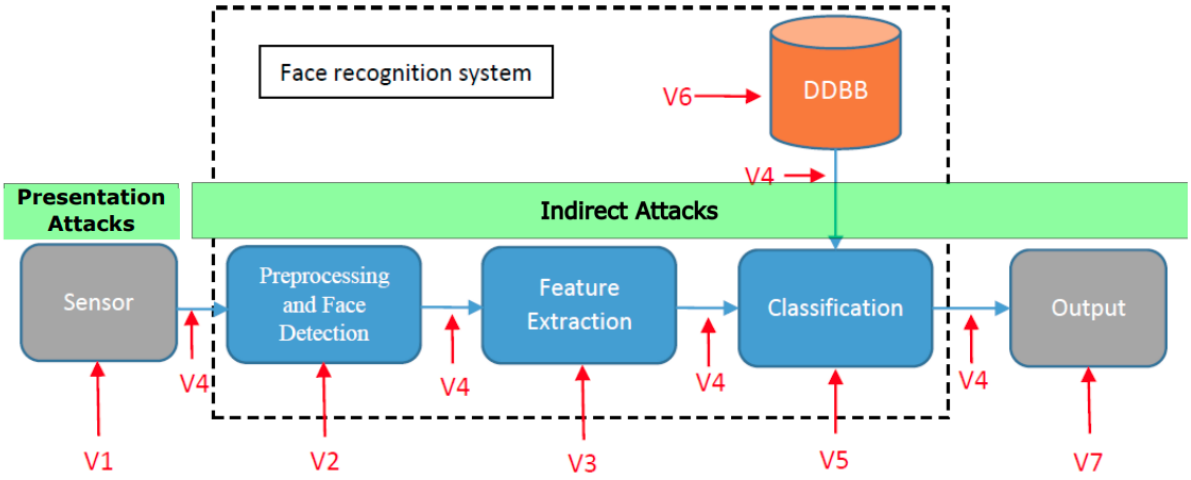


Figure 1.2: Diagram of a typical biometric system structure. This system comprises multiple modules and potential points susceptible to attacks, denoted as Vulnerabilities V1 to V7. Presentation attacks occur at the sensor level (V1) and don't require access to the system's internal modules. Indirect attacks (V2 to V7), however, can target databases, matchers, communication channels, etc., demanding access to the system's inner workings and, in many instances, specific knowledge of their operations.

[68]). They present these elements to the acquisition scanner in an effort to deceitfully gain access to the biometric system.

A significant volume of biometric data, including photographs, videos, and audio recordings of individuals' faces, eyes, voices, and behaviors, are readily available through platforms like social media. Presentation attackers are well aware of this reality and leverage these sources of information to attempt to bypass face recognition systems [64]. This underscores a well-known limitation of biometrics: "biometric characteristics are not secrets" [39]. Interestingly, the very qualities that make the face an attractive biometric trait for person recognition, namely its non-intrusive and distance-capturing nature, also render it susceptible to misuse by attackers.

In addition to the accessibility of legitimate users' facial images, it is recognized that face recognition systems exhibit vulnerabilities to presentation attacks, falling into one of three primary categories:

- **Photograph-Based Attacks:** Attackers may use a photograph of the intended user to impersonate them [84].
- **Video-Based Attacks:** Videos featuring the target user can be employed for impersonation attempts [14, 58].
- **3D Model-Based Attacks:** Attackers can construct and use 3D models of the target's face, such as hyper-realistic masks, as a means of attack [22]."

Most state-of-the-art facial biometric systems are susceptible to straightforward attacks that would appear to an average person unless Presentation Attack Detection (PAD) measures are integrated. For

instance, attempting to impersonate someone using a mere photograph of their face is a vulnerability. Thus, when designing a secure Face Recognition System (FRS) for real-world applications, such as replacing password-based authentication, prioritizing Presentation Attack Detection (PAD) techniques is paramount right from the system's inception.

1.4 Face Presentation attack

Face presentation attacks (PAs) aim to deceive real-world automatic face recognition (AFR) systems by introducing facial images onto physical mediums like photographs, videos, or 3D masks of targeted or concealed individuals in front of imaging sensors. Fig. 1.3 illustrates some representative PA types. These PAs can be categorized into two main groups based on the attackers' intentions:

- Impersonation: In this category, attackers seek to mimic targeted identities by replicating a genuine user's facial features on presentation instruments like photos, electronic screens, or 3D masks.

- Obfuscation: Here, the goal is to conceal or remove the attacker's own identity by decorating the face, which may involve wearing glasses, wigs, makeup, or tattoos.

PAs can be further classified into two-dimensional (2D) and three-dimensional (3D) attacks based on geometric depth. Common 2D PAs include print and replay attacks, such as using flat or wrapped printed photos, photos with eye or mouth cutouts, or replaying face videos on electronic screens. In contrast, 3D presentation attacks involve more realistic representations, like 3D face masks and mannequins. These 3D objects exhibit lifelike color, texture, and geometric structures, constructed from various materials like paper, resin, plaster, plastic, silicone, and latex.

Another categorization of PAs considers the proportion of the facial region they cover. Some PAs cover the entire face, while others target only specific facial regions. For instance, attackers may cut out eye regions from a printed face photo to spoof eye-blinking-based PAD systems. Alternatively, they might wear eyeglasses with adversarial patterns in the eye region to challenge face PAD algorithms. Compared to attacks on the entire face, partial attacks are more elusive and pose greater challenges for defense."

1.4.1 Presentation Attack Methods

Typically, a Face Recognition System (FRS) can be deceived by presenting a photograph, video, or 3D mask of a targeted individual to the sensor (e.g., a camera) as illustrated in Fig. 1.2. There are alternative methods to circumvent FRS, including the use of makeup [18, 13] or even resorting to plastic surgery. However, the most prevalent types of attacks involve photographs and videos due to the widespread availability of facial imagery (e.g., on social media and in video surveillance) and the affordability of high-resolution digital cameras, printers, and digital screens.

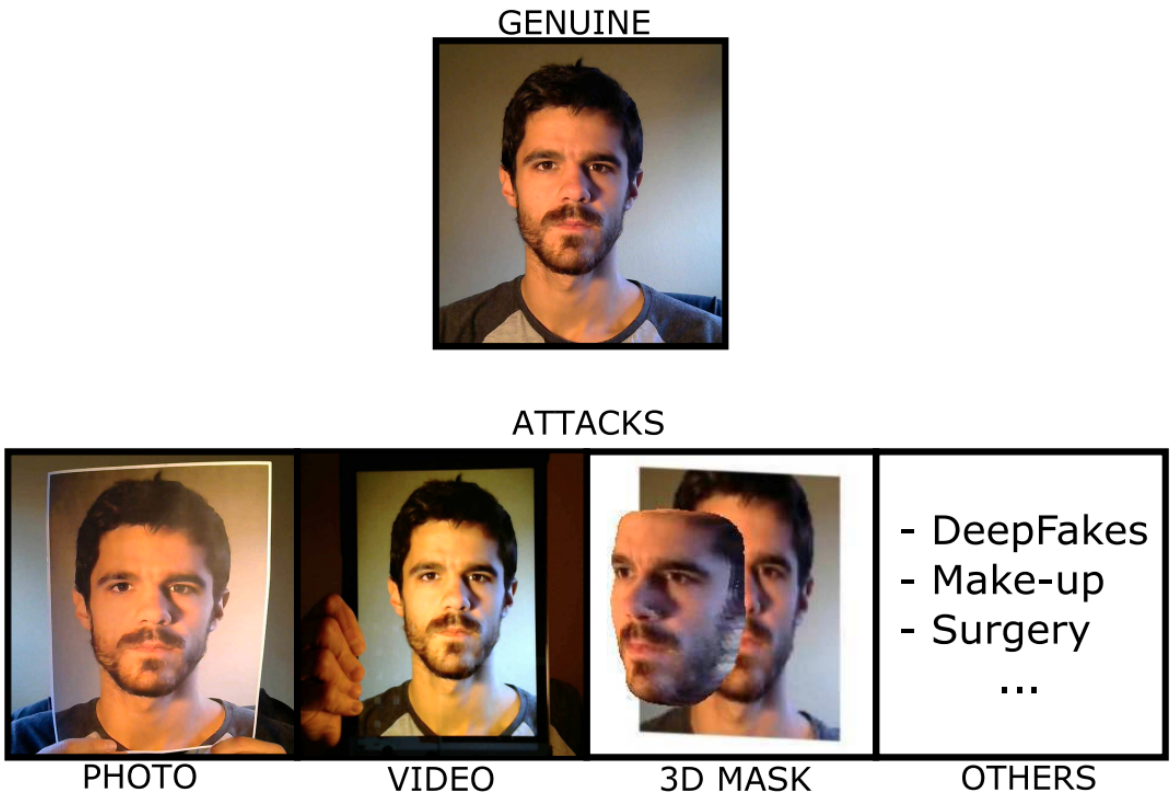


Figure 1.3: Examples of face presentation attacks: The top image depicts a genuine user, while the images below showcase various presentation attacks using different Presentation Attack Instruments (PAIs) presented to the sensor. These include photos, videos, 3D masks, DeepFakes, makeup, surgery, and others.

In terms of attack classification, a general categorization considers the nature and complexity level of the Presentation Attack Instrument (PAI) used for the attack. This classification typically includes photo-based, video-based, and mask-based attacks, as depicted in Fig. 1.2. It's important to note that while these categories encompass the most common attack types, some sophisticated attacks may not neatly fit into any single category or may belong to multiple categories simultaneously. One such example is DeepFake techniques, which are capable of creating fraudulent videos by swapping one person's face with another's. These methods can be classified as photo attacks, video attacks, or even mask attacks. In this chapter, we classify these more intricate and specialized attacks under the category 'Other attacks.

Photo Attacks

A photo attack involves presenting a photograph of the targeted individual to the sensor of a face recognition system [5, 69], as shown in the example in Fig. 1.2. Photo attacks are particularly concerning for several reasons. Firstly, printing color images of a genuine user's face is both inexpensive and easily

achievable, often referred to as print attacks in the literature [6]. Alternatively, photos can be displayed on high-resolution screens of devices such as smartphones, tablets, or laptops [5, 14, 97]. Secondly, the proliferation of social media platforms like Facebook, Twitter, and Instagram has made it effortless to obtain authentic face samples [64]. Furthermore, the recent reduction in the price and size of digital cameras has made it feasible to capture high-quality photos of legitimate users using concealed cameras.

Within the realm of photo attacks, there exist more sophisticated techniques like photographic masks. This approach entails printing a photograph of the subject's face and then creating openings for the eyes and mouth [97]. This method effectively circumvents presentation attack detection techniques reliant on blink detection and the tracking of eye and mouth movements [78].

While these attacks may seem overly simplistic, some studies have indicated vulnerabilities in many state-of-the-art systems [65, 77, 74]. Given their simplicity, implementing robust countermeasures that effectively mitigate these attacks should be a priority for any facial recognition system.

Information needed to perform the attack: An image of the face of the subject to be impersonated. Generation and acquisition of the PAIs: Various options exist to obtain high-quality face images of users to be impersonated, including social networks, internet profiles, and hidden cameras. These photographs can then be printed or displayed on a screen to present them to the sensor of the Face Recognition System (FRS). Expected impact of the attack: Most basic face recognition systems are susceptible to this type of attack if specific countermeasures are not in place. However, the literature offers a wide array of approaches with good detection rates for printed photo attacks [5, 7].

Video based Attacks

Much like photo attacks, the acquisition of videos featuring individuals intended for impersonation has become increasingly convenient due to the proliferation of public video-sharing platforms, social networks, and the availability of hidden cameras. Another advantage of using this form of attack is that it enhances the likelihood of success by introducing a sense of liveliness to the fake biometric sample [79, 70].

Once a video of the genuine user is procured, an attacker can play it on any device capable of video playback, such as a smartphone, tablet, or laptop, and then present it to the sensor or camera [52], as depicted in Fig. 1.2. This category of attacks is commonly known as replay attacks, which represent a more sophisticated iteration of simple photo attacks. Replay attacks are considerably more challenging to detect compared to photo attacks, as they not only mimic facial texture and shape but also replicate dynamic aspects, such as eye blinking, mouth movements, and other facial expressions [14]. Given their heightened complexity, it's reasonable to assume that systems vulnerable to photo attacks will exhibit even greater susceptibility to video attacks. Additionally, resilience against photo attacks does not necessarily translate to robustness against video attacks [97]. Consequently, specific countermeasures must be developed and implemented, such as authentication protocols based on challenge-response mechanisms [78].

- **Information needed to perform the attack:** A video of the face of the subject to be impersonated.

- **Generation and acquisition of the PAIs:** Similar to photo attacks, obtaining face videos of users to be impersonated is relatively straightforward, thanks to the prevalence of video sharing platforms (e.g., YouTube, Twitch), social networks (e.g., Facebook, Instagram), and the availability of hidden cameras. These videos are then displayed on a screen to present them to the sensor of the Face Recognition System (FRS).
- **Expected impact of the attack:** Much like photo attacks, most face recognition systems are inherently vulnerable to these attacks. Consequently, countermeasures, such as challenge-response-based authentication or mechanisms considering facial appearances, are typically implemented. With these countermeasures in place, classic video attacks tend to have a low success rate.

1.5 Adversarial Attacks

Adversarial attacks in deep learning are a profound challenge that has sent ripples through the realm of artificial intelligence. At their core, these attacks involve sneaky tweaks to input data, carefully designed to trick even the most advanced neural networks. The targeted vulnerability here is the extraordinary sensitivity of these AI systems. In the realm of deep learning, the existence of adversarial examples serves as a profound reminder of the complex and sometimes fragile nature of artificial neural networks. Think about it like this: we, humans, can easily recognize everyday things like stop signs. However, imagine a slight, nearly invisible change to that stop sign, one that we wouldn't notice but would confuse an autonomous vehicle's neural network. It might suddenly see that stop sign as something entirely different, like a yield sign, and you can imagine the potential for chaos on the road. These attacks are like optical illusions for machines, and they pose a serious challenge to the reliability and safety of AI systems in various fields, from computer vision to understanding human language.

The inception of adversarial attacks can be traced to a seminal paper in 2013, which introduced the concept of "adversarial examples." These are input data, like images or sequences, that have undergone carefully calculated perturbations. To the human eye, these perturbed inputs appear indistinguishable from their clean counterparts, but they are engineered to perplex and mislead deep neural networks. The perturbations are so subtle that they mirror imperceptible changes one might encounter in the real world, such as slight lighting variations or noise. Yet, these nearly invisible alterations can lead to catastrophic failures in AI models.

To illustrate the perplexity of these attacks, let's consider the famous example of a panda. Imagine you have a perfectly normal image of a panda, one that any human would identify without hesitation. Now, an adversarial attack subtly tweaks a few pixels in this image, changes so tiny that you'd barely notice. But when you show this slightly modified image to a deep learning model, it confidently declares that it's looking at a gibbon instead of a panda! It's like magic, but a kind that exposes a chink in the armor of AI. These attacks are disconcerting because they reveal that our most sophisticated AI systems are vulnerable to tiny, calculated manipulations that we can hardly perceive. It's as if a slight change in the brushstroke of a painting could make us see a completely different picture. This vulnerability raises

profound questions about the reliability and security of AI, echoing through fields where AI is used to make critical decisions, from self-driving cars to medical diagnosis to face recognition.

Adversarial examples are carefully crafted perturbations applied to input data that are imperceptible to humans but cause deep learning models to make incorrect predictions with high confidence. This phenomenon raises critical questions about the robustness and security of deep neural networks. we delve into the mathematics behind adversarial examples, shedding light on how these subtle manipulations can deceive even the most advanced neural networks.

1.5.1 Notation and Preliminaries

The goal of an adversarial attack is to find a perturbation δ to add to the original input x in a way that maximizes the model's prediction error. This can be formulated as an optimization problem:

$$\begin{aligned} &\text{maximize } \mathcal{J}(x+\delta, y; \theta), \\ &\text{subject to } \| \delta \| < \epsilon \end{aligned} \tag{1.1}$$

where:

- x: Original input data
- y: The true label associated with x
- $\mathcal{J}(x+\delta), y; \theta)$: The loss function measuring the model's error on the input-label pair (x, y) with parameters θ
- δ: The perturbation to be added to x
- ϵ: A small constant controlling the magnitude of the perturbation to ensure it remains imperceptible to humans.

1.5.2 Types of Adversarial Attacks

Adversarial attacks can be classified into two broad categories: white-box and black-box.

White Box Adversarial Attacks on Face Presentation Attack Detection Systems

White-box adversarial attacks are a category of adversarial attacks in deep learning where the attacker has complete knowledge of the target model being attacked. This knowledge includes detailed information about the model's architecture, its parameters (weights and biases), and access to the training data used to train the model. White-box attacks are often considered the most straightforward and well-understood type of adversarial attacks.

White-box adversarial attacks on face presentation attack detection (PAD) systems are attempts to craft malicious inputs (adversarial examples) that can successfully deceive these systems while the

attacker has complete knowledge of the PAD model. These attacks exploit the sensitivity of PAD models to subtle input changes and are aimed at undermining the security and reliability of facial recognition systems designed to distinguish between genuine and fake face presentations (spoofing attacks).

Key Components of White-Box Adversarial Attacks on PAD Systems:

- **Access to PAD Model Architecture:** In a white-box attack scenario, the attacker knows the precise architecture of the target PAD model. This includes the type of neural network used (e.g., convolutional neural network or recurrent neural network), the number of layers, the number of neurons in each layer, activation functions, and any other architectural details.
- **Knowledge of Model Parameters:** The attacker is aware of the exact values of the model's parameters, including the weights and biases associated with each neuron in the network. This information is typically considered highly sensitive and is not publicly disclosed for most machine learning models.
- **Training Data:** The attacker has access to the training data that was used to train the PAD model. This training data typically consists of genuine face images and various types of spoofing attack samples (e.g., printed photos, videos, masks) designed to mimic real-world presentation attacks.

Process of White-Box Adversarial Attacks on PAD Systems The attacker utilizes their in-depth understanding of the PAD model to generate adversarial examples that can bypass the system's spoof detection. Here's a step-by-step explanation of how this process works:

- **Identify Vulnerabilities:** The attacker analyzes the PAD model to identify vulnerabilities or areas where the model might fail to detect presentation attacks. These vulnerabilities could be related to the model's decision boundaries, feature extraction, or any weaknesses in its design.
- **Compute Gradients:** Similar to white-box attacks in general deep learning, the attacker calculates gradients of the model's loss function with respect to the input data. In the context of PAD, this loss function aims to differentiate between genuine and spoofed face presentations.

$$\nabla_x \mathcal{J}(x, y; \theta) \tag{1.2}$$

  Where:

  - ∇_x is the gradient operator with respect to the input data x.
  - $\mathcal{J}(x, y; \theta)$ is the loss function measuring the model's error for input x and true label y with model parameters θ.

- Generate Adversarial Perturbation: Using the computed gradients, the attacker generates an adversarial perturbation δ designed to make the PAD model misclassify a spoofed face presentation

as genuine. The perturbation is added to the original input x to create the adversarial example.

$$\begin{aligned} \delta &= \epsilon \cdot \text{sign}(\nabla_x \mathcal{J}(x, y, \theta)) \\ x_{adv} &= x + \delta \end{aligned} \tag{1.3}$$

Here, ϵ controls the magnitude of the perturbation to ensure it remains inconspicuous.

- Generate Adversarial Example: The attacker crafts the adversarial example by adding the perturbation δ to the original input x. This adversarial example is designed to evade the PAD model's detection.

Black Box Adversarial Attacks on Face Presentation Attack Detection Systems

Black-box adversarial attacks on face presentation attack detection (PAD) systems are a challenging yet crucial area of research in computer vision and security. These attacks involve crafting adversarial examples to deceive PAD systems while having limited knowledge about the target system. In this explanation, we'll delve into the details of black-box adversarial attacks on PAD systems, focusing on their use of Generative Adversarial Networks (GANs):

Understanding Black-Box Adversarial Attacks on PAD Systems

- **Limited Knowledge:** In a black-box scenario, the attacker possesses minimal information about the PAD system they intend to deceive. They may know that a PAD system is in place but lack knowledge about its architecture, parameters, or training data. This mimics real-world scenarios where attackers often have limited access to system details.

- **Crafting Adversarial Perturbations:** The central objective of a black-box adversarial attack is to generate adversarial examples that can bypass the PAD system's detection while still appearing as legitimate inputs. To achieve this, attackers utilize generative models like GANs to create perturbations.

- **Generative Adversarial Networks (GANs):** GANs consist of two networks - a generator and a discriminator - engaged in a constant struggle. The generator aims to create realistic data (in this case, adversarial examples) to deceive the discriminator, which, in turn, strives to distinguish between real and generated data. This adversarial training process results in the generation of highly realistic, hard-to-distinguish adversarial examples.

- **Train the GAN:** The attacker trains a GAN on the collected genuine samples. The generator learns to produce perturbations that, when added to genuine samples, make them appear different from the original but still classified as genuine by the PAD system.

- **Adversarial Perturbation Generation:** The generator of the GAN creates adversarial perturbations designed to transform live samples into convincing spoofed samples. These perturbations are carefully crafted to maximize the likelihood of being classified as live by the PAD system.

- **Adding Perturbations:** The attacker adds the adversarial perturbations to live images, creating adversarial examples. These examples are specifically designed to confuse the PAD system, making it perceive the adversarial examples as live even though they have been altered.

1.6 Physical Adversarial Attacks

Unlike digital attacks that manipulate images through software, physical attacks are aimed at fooling PAD systems by introducing real-world modifications to the presentation of a biometric trait, such as a human face. These attacks are designed to mimic the traits that a PAD system is supposed to detect as presentation attacks, thereby bypassing the system's defense mechanisms.

One of the primary forms of physical adversarial attacks involves print attacks. In this scenario, an attacker creates a physical replica of a legitimate user's face, often using high-resolution prints on various materials like paper, silicone, or 3D masks. These replicas are carefully crafted to closely resemble the appearance of the authorized user. When presented to the PAD system, these print attacks aim to be classified as genuine, thereby tricking the system into granting unauthorized access. The success of such attacks is concerning, as it highlights the need for PAD systems to be robust not only against digital manipulations but also against real-world physical presentation attacks.

Another type of physical adversarial attack is the replay attack. In a replay attack, an attacker captures a legitimate user's biometric data, typically through video or audio recordings, and replays this data during an authentication attempt. This attack exploits the fact that some PAD systems may struggle to distinguish between live biometric data and previously recorded data, especially if they lack liveness detection mechanisms. As a result, the replayed data may be accepted as genuine, leading to unauthorized access.

Physical adversarial attacks on PAD systems have real-world implications for security and privacy. They highlight the need for PAD systems to incorporate robust liveness detection mechanisms that can differentiate between live biometric data and presentation attacks. Additionally, research and development efforts are ongoing to improve the resilience of PAD systems against physical attacks by leveraging techniques like texture analysis, thermal imaging, or 3D depth sensing to detect and deter presentation attacks effectively. As biometric authentication becomes more prevalent in various applications, including smartphones, access control systems, and financial services, addressing the vulnerability of these systems to physical adversarial attacks is of paramount importance to ensure the security and trustworthiness of biometric-based authentication mechanisms.

1.7 Motivation

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

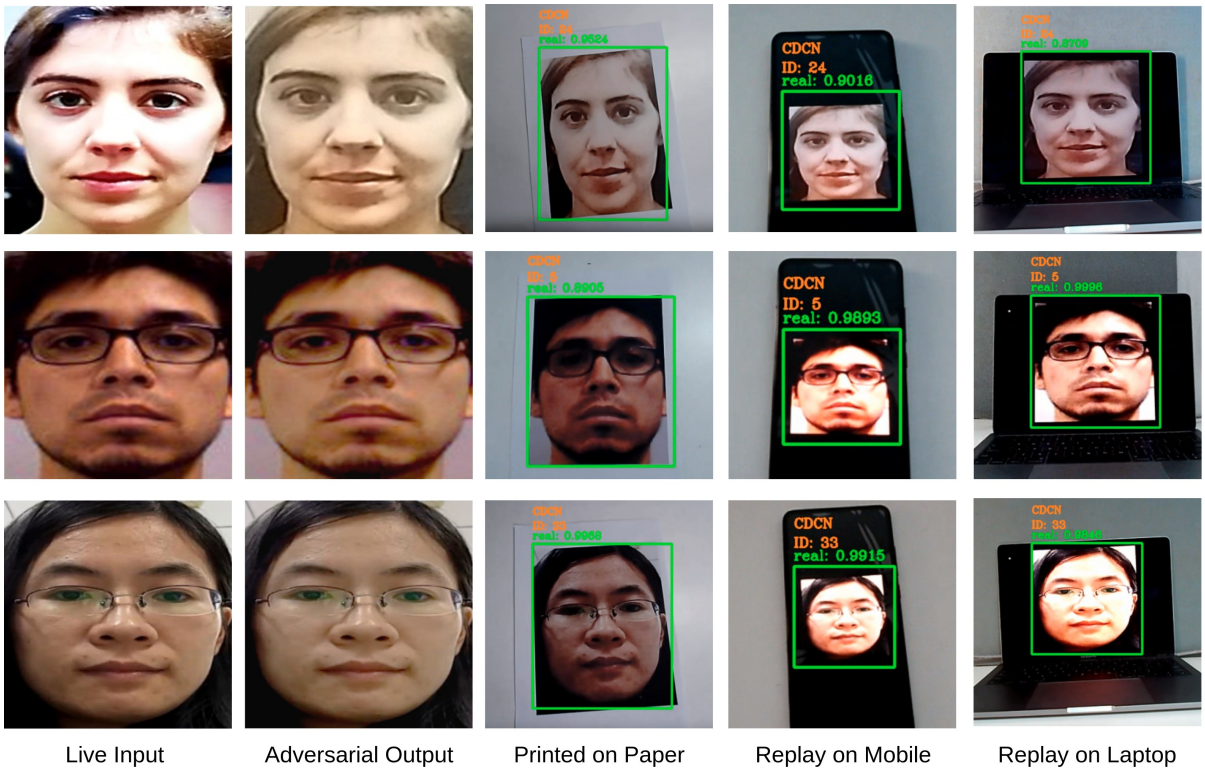


Figure 1.4: Physical Adversarial attacks on Face Presentation Attack Detection Sytems

into these systems (Fig 1.5) 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-artapproaches use auxiliary 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.

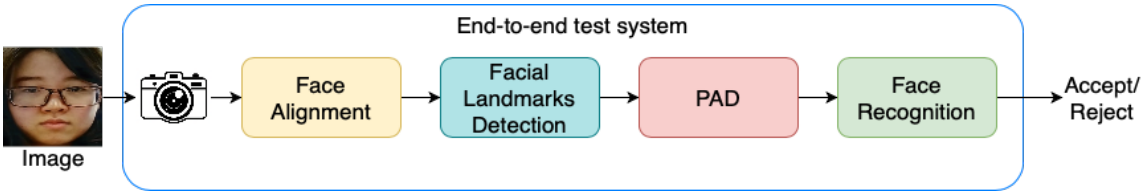


Figure 1.5: Face recognition pipeline

Attacks on PADs can be divided into two categories: presentation attacks and digital adversarial attacks. Presentation attacks are created using printed photographs or replayed videos. For generating presentation attacks, the hacker needs to actively participate by wearing a mask or replaying a photograph/video of the genuine individual, which may be conspicuous in scenarios where human operators are involved. Adversarial faces, on the other hand, do not require active participation during verification.

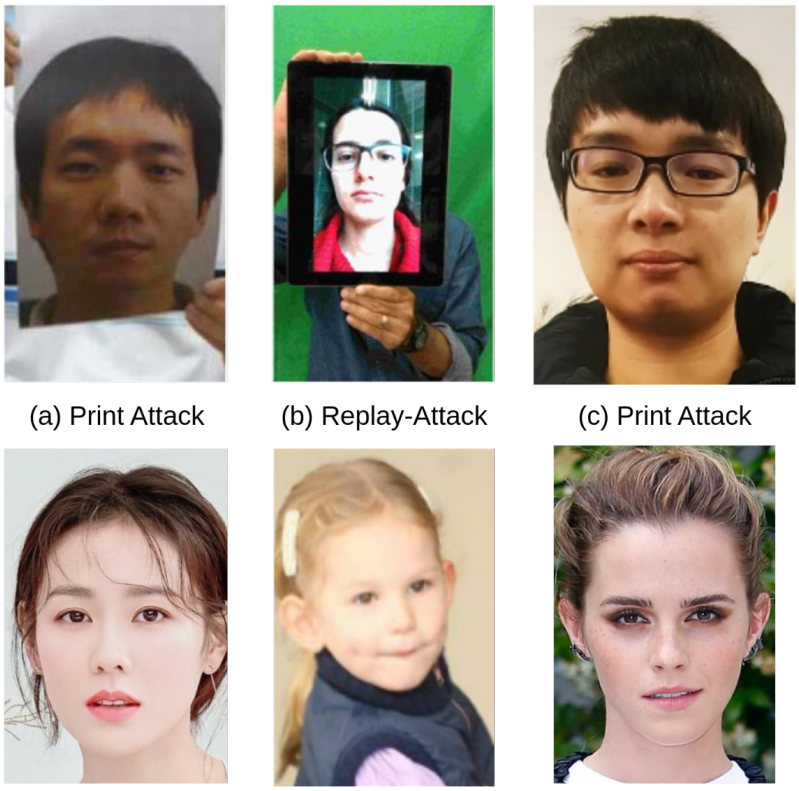


Figure 1.6: Two types of face spoofing attacks: The first row shows the physical presentation attacks, (a) a printed photograph, (b) replaying the targeted person's video on a screen, and (c) a print attack image of the target's face. The second row shows digital adversarial attacks, created by adding slight modifications to the captured image. To a human observer, face presentation attacks (a-c) are more conspicuous than adversarial faces.

The use of deep learning has significantly improved the accuracy of Presentation Attack Detectors. However, these systems are known to be vulnerable to adversarial examples [40, 19, 34, 13], which are constructed to fool models by adding perturbations to live face images. Adversarial attacks have been explored extensively and have exposed some of the vulnerabilities of deep learning-based PAD models. This is a concern, especially in the safety-sensitive fields, as these attacks require a small intensity

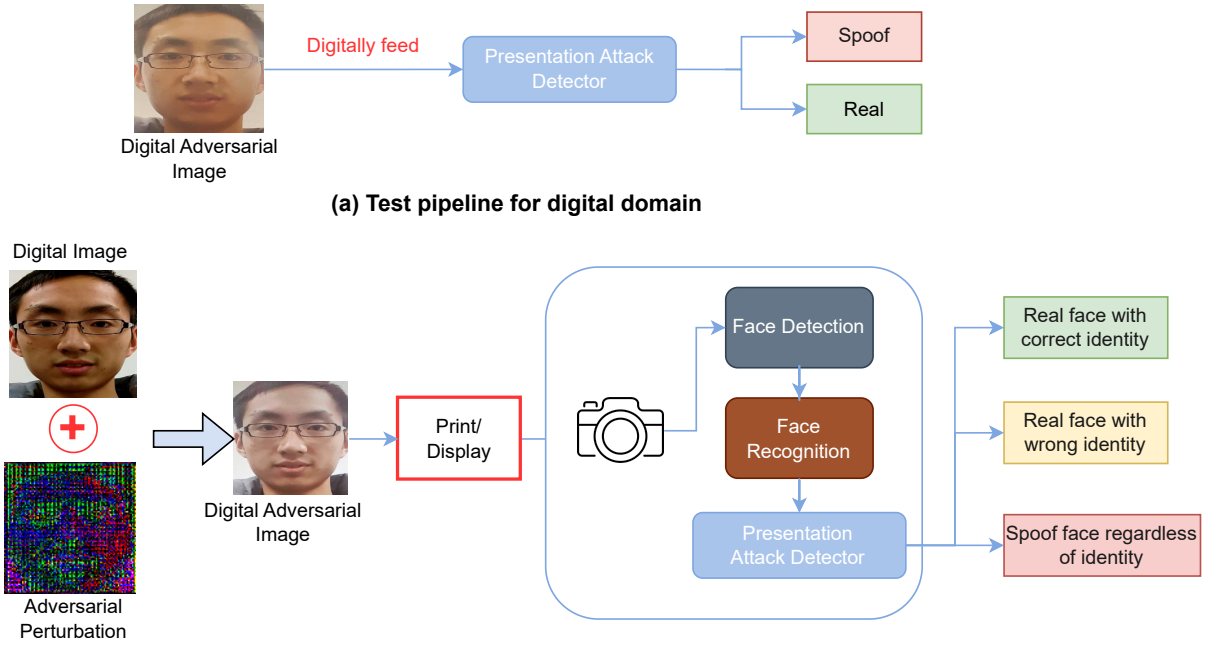


Figure 1.7: Experimental pipelines to evaluate the performance of the adversarial attacks. (a) shows the pipeline used when we attack a PAD in the digital domain, and (b) shows our testing pipeline in a physical domain. The digital image has to undergo two transformations and has to be effective after distortions are introduced in these processes.

of adversarial perturbation and are usually imperceptible to the human eye. Adversarial images can lead recognition models to falsely reject a genuine subject or falsely match an impostor, known as an obfuscation attack and impersonation attack, respectively. Some adversarial attacks such as FGSM and PGD [19, 32] are very potent as they generate adversarial examples by back-propagating through the target model. These methods can easily mislead the PAD model, resulting in missed presentation attacks. However, these attacks need access to the internals of the capture system to modify the captured image.

Adversarial image attacks can further be of two forms: untargeted and targeted. In untargeted adversarial attacks, we cannot control the output label of the adversarial image. Whereas, in targeted adversarial attacks, we can control the output label of the image.

Fig 1.6 shows the difference between the two attacks that we have mentioned so far. As seen in the fig, physical adversarial attacks require significant adversarial perturbations as they must survive the print and capture process. The physical and environmental conditions, such as the surface nature and deformations of the paper, screen characteristics, and illumination during the physical capture process, make the problem much more challenging. Compared to digital-world attacks, attacks in the physical world are more challenging because they require robust perturbations to overcome various constraints

and disturbances. Therefore, a series of operations need to be taken to ensure generated adversarial perturbations obtain physical-world robustness.

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 Fig 1.7.

1.8 Contributions

The contributions of the thesis can be summarized as follows:

1. We explore a new physical adversarial attack strategy that can fool a PAD. The system detects and recognizes the facial region as a face while also preserving the subject's identity.

2. AdvPrint, A gradient-based white-box attack strategy that can perform an adversarial attack and learns to generate visually realistic adversarial face images that simulate a physical print and video-replay attack.

3. 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.

4. We propose AdvGen, a generative adversarial network trained to generate perturbations that are robust to distortions introduced to an image during physical transformations. It is a whitebox adversarial attack generator.

5. 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.

6. 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].

1.9 Thesis Organization

The rest of the thesis is organized as follows:

- **Chapter 2** provides background on face presentation attack detection methods, generative adversarial networks and adversarial attacks on face recognition system.
- **Chapter 3** introduces a whitebox physical adversarial attacks using a method called *AdvPrints*. It uses contrastive methods and a conventional adversarial attack strategy to generate a image which is an adversary for the PAD systems.
- **Chapter 4** discusses a black-box physical adversarial attack on face PAD systems. A generative adversarial network is trained to generate an adversarial perturbation which shows superior attack success rate on state-of-the-artface PAD systems.
- **Chapter 5** summarizes this thesis and presents ideas for future work.

Chapter 2

Literature Survey

2.1 Taxonomy of Face Presentation Attack Detection

Face recognition systems are primarily designed to distinguish between genuine users and are not inherently equipped to determine whether the biometric sample presented to the sensor is real or fake. In this context, a presentation attack detection method is broadly defined as any technique that automatically distinguishes between authentic biometric characteristics presented to the sensor and artificially generated Presentation Attack Instruments (PAIs).

There are four main approaches to conducting presentation attack detection [40]: (i) with dedicated hardware to detect an evidence of liveness, which is not always possible to deploy, (ii) with a challenge-response method where a presentation attack can be detected by requesting the user to interact with the system in a specific way, (iii) employing recognition algorithms intrinsically resilient against attacks, and (iv) with software that uses already available sensors to detect any pattern characteristic of live traits.

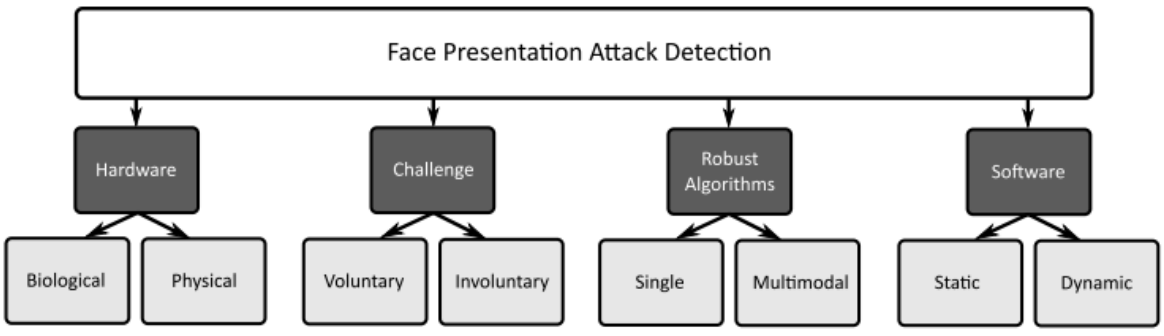


Figure 2.1: Taxonomy of Face Presentation Attack Detection methods.

- **Hardware-based:** Presentation Attack Detection (PAD) methods that rely on dedicated hardware leverage specialized sensors such as Near Infrared (NIR) cameras, thermal sensors, Light Field Cameras (LFC), multispectral sensors, and 3D cameras. These sensors exploit the unique properties

of different types of dedicated hardware to more effectively measure and distinguish the biological and physical characteristics of genuine faces from Presentation Attack Instruments (PAIs). For instance, thermal cameras can process temperature information, and 3D acquisition sensors can estimate the 3D volume of artifacts. Although these approaches tend to achieve high presentation detection rates, they are not widely adopted because the required hardware is often expensive and not readily available.

- **Challenge-Response:** In Challenge-Response PAD methods, users are presented with a challenge, such as completing a predefined task or responding to a stimulus. The user's voluntary or involuntary response to this challenge is then analyzed to determine if the access attempt is from a legitimate user or an attacker. For instance, some methods study involuntary eye responses to visual stimuli, while others ask users to perform specific eye movements or say predefined words. While challenge-response methods can be effective against many presentation attacks, they typically require more time and cooperation from users, which may not always be feasible or desirable.

- **Robust Algorithms:** Existing face recognition systems can be designed or trained to learn how to differentiate between legitimate faces and PAIs, making them inherently robust to certain types of presentation attacks. However, developing face recognition algorithms that are intrinsically robust to presentation attacks is a complex task. The most common approach involves relying on multimodal biometrics, which combines information from other biometric modalities to enhance security.

Despite the advantages of hardware-based and challenge-response methods, the ease of deploying software-based PAD methods has led to significant research focusing on software-based approaches. In the following section of this chapter, we will delve into software-based PAD, exploring its different categories and components."

Fig 2.1 provides an organizational framework for PAD methods based on this proposed taxonomy.

2.2 Software-based face PAD

Software-based PAD methods are often convenient because they offer the flexibility to enhance existing systems without the need for additional hardware. They also enable real-time authentication without requiring extra user interaction. Whether they fall into the hand-crafted or deep learning category, software-based PAD methods can be broadly categorized based on their consideration of temporal information into two main groups: static and dynamic analysis.

2.2.1 Static Analysis

This subsection refers to the development of techniques that analyze static features like the facial texture to discover unnatural characteristics that may be related to presentation attacks. The key idea of

the texture-based approach is to learn and detect the structure of facial micro-textures that characterise real faces but not fake ones. Micro-texture analysis has been effectively used in detecting photo attacks from single face images: extraction of texture descriptions such as Local Binary Patterns (LBP) [14] or Gray-Level Co-occurrence Matrices (GLCM) followed by a learning stage to perform discrimination between textures

2.2.1.1 Handcrafted Feature based Face PAD

Based on distinct feature properties, handcrafted feature-based face Presentation Attack Detection (PAD) approaches are categorized into five primary cues: structural material, image quality, texture, micro motion, and physiological signals. These handcrafted features are typically combined with binary classification models like Support Vector Machines (SVM) or Multi-Layer Perceptrons (MLP). The aim is to differentiate between genuine faces and presentation attacks effectively.

Approaches Based on Structural Materials: In real-world scenarios, presentation attacks (PAs) often involve physical presentation attack instruments (PAIs) like paper, glass screens, and resin masks. These PAIs have distinct material properties that differ from human facial skin. These differences can be described as meaningful cues for detecting spoofing attempts, such as variations in structural depth and specular reflection. To obtain information about the 3D structure or material composition of the face, one direct approach is to use binocular or depth cameras, as they can capture depth information. However, in practical applications, single RGB cameras are more commonly used. Consequently, many face Presentation Attack Detection (PAD) research efforts focus on estimating 3D and material cues using monocular RGB cameras.

On one hand, considering that 2D PAs on paper and screens typically lack depth information, Wang et al. [68] proposed a method to recover the sparse 3D shape of face images, enabling the detection of various 2D attacks. On the other hand, the differences in illumination and reflection between human facial skin and 2D PAs are leveraged as significant spoof cues. Kim et al. [16], for instance, utilized illumination diffusion cues based on the observation that illumination from 2D surfaces of 2D attacks diffuses more slowly and exhibits a more uniform intensity distribution than that from 3D surfaces. Additionally, Wen et al. [71] introduced a technique to calculate statistical features based on the percentage of specular reflection components in face images to detect screen replay attacks. While methods based on structural material cues hold theoretical promise for detecting 2D PAs, estimating depth and material information from a monocular RGB camera poses challenges due to ill-conditioned problems, and these methods tend to have high computational complexity.

Image Quality-Based Approaches: In the realm of Presentation Attack Detection (PAD), the quality of the presented face images serves as a crucial indicator. Spoofed faces often exhibit degraded image quality owing to color distortions and instrument artifacts stemming from specific physical Presentation

Attack Instruments (PAIs). Researchers have harnessed this degradation in image quality as a significant cue for effective face PAD.

For instance, Galbally et al. [21] employed a battery of 25 image quality assessment (IQA) metrics, comprising 21 full-reference and 4 non-reference metrics, to facilitate face liveness detection. In a similar vein, Wen et al. [71] adopted three distinct IQA features, namely blurriness, color moment, and color difference, to enhance face PAD. These IQA features prove effective in capturing the intrinsic distortions present in spoofed images.

Image quality-based methods excel in detecting screen-replayed faces, low-quality printed faces, and roughly constructed 3D mask-based spoofed faces. However, they do exhibit limitations when confronted with high-quality printed faces and sophisticated, high-fidelity 3D mask-based attacks, which can result in elevated false acceptance rates for these methods.

Texture-based approaches: Texture-based approaches in Presentation Attack Detection (PAD) take advantage of specific characteristics of Presentation Attack Instruments (PAIs). Typically, PAIs result in spoofed faces with coarser and smoother textures compared to genuine faces captured directly by cameras, which retain finer-grained local texture details. To address this, various texture-based methods have been developed for face PAD.

These methods often utilize classical local texture descriptors such as Local Binary Pattern (LBP) and Histogram of Oriented Gradients (HOG) to capture intricate texture features from facial images. Notably, some researchers have observed that texture features in the HSV color space exhibit greater invariance across different environments. For instance, Boulkenafet et al. propose extracting LBP-based color texture features from the HSV space, which proves efficient and versatile.

However, it's crucial to note that texture-based methods require high-resolution input to effectively discern subtle texture distinctions between genuine and spoofed faces. In scenarios where image quality is suboptimal, these methods may yield a high false rejection rate. Furthermore, the diversity of image acquisition conditions and spoofing instruments can lead to varying extracted texture patterns, limiting the generalizability of these approaches under complex real-world circumstances.

2.2.1.2 Dynamic Analysis

Micro Motion based Approaches: Approaches based on capturing micro motions to detect liveness involve observing short-term facial dynamics, such as expressions and head movements, as well as dynamic textures that distinguish live from spoof samples. For instance, Tirunagari et al. [63] proposed a method that temporally amplifies facial motion and extracts dynamic features, including the Histogram of Oriented Optical Flow (HOOF) and Local Binary Pattern histograms from Three Orthogonal Planes (LBP-TOP) for face PAD. However, motion magnification often introduces external noise, affecting the robustness of subsequent feature representation.

In an alternative approach, Siddiqui et al. [59] employ dynamic mode decomposition to select the most reliable dynamic mode for facial motion feature extraction, avoiding some of the noise issues associated

with motion magnification. Nonetheless, micro motion-based methods may not be effective against certain attacks, such as wrapped or shaking paper attacks and video replay attacks, due to interference from undesired dynamics. These methods assume that there is a clear distinction in non-rigid motion between genuine and presentation attacks. However, explicitly describing and representing such micro motions can be challenging.

Remote Photoplethysmograph based Approaches: Physiological signal is another important living body signal, and it is also an intrinsic cue for distinguishing live faces from artificial materials. In recent years, remote photoplethysmograph (rPPG) technology [79] has developed quickly, which aims at measuring blood pulse flow by modeling the subtle skin color changes caused by the heartbeat. Due to the low transmittance characteristics of artificial materials, rPPG signals from the live faces are usually periodic, but more noisy on the PAs such as 3D mask and printed paper. Therefore, rPPG signals are suitable for face liveness detection. Li et al. [37] analyze the live/spoof rPPG cues via calculating the statistics of the rPPG frequency responses. Different from the method of spectrum analysis using long-term observation of rPPG signals in the frequency domain, Liu et al. [41] propose to leverage the temporal similarity of facial rPPG signals for fast 3D mask attack detection, which can be within one second by analyzing the time-domain waveform of the rPPG signal. However, rPPG cues are sensitive to the head motion, light condition, and video quality. Another disadvantage is that the replayed video attack on electronic screen might still contain weak periodic rPPG signals.

2.3 Deep Learning based Face PAD

The prevalence of deep learning in computer vision has extended to the realm of face Presentation Attack Detection (PAD). In this context, traditional deep learning approaches are worth mentioning. Initially, let's explore conventional deep learning methods that employ cross-entropy and pixel-wise supervision. Subsequently, we will delve into domain-generalized deep learning techniques.

2.3.1 Traditional Deep Learning Approaches with Cross-Entropy Supervision

Traditional deep learning approaches for face Presentation Attack Detection (PAD) have often employed cross-entropy (CE) loss for supervision. This treatment is rooted in the notion that face PAD is essentially a classification task, which can be binary (bonafide vs. presentation attack) or multi-class (bonafide, print, replay, mask, etc.). In these approaches, deep models ϕ are utilized to extract features F from the input face X, followed by classification heads that make binary predictions Y. These predictions are then supervised using binary cross-entropy (BCE) loss.

In the context of BCE loss, Y_{gt} represents the ground truth, where $Y_{gt} = 0$ denotes presentation attacks (PAs), and $Y_{gt} = 1$ signifies bonafide samples. The pioneering work by Yang et al. [73] introduced an end-to-end deep face PAD method employing shallow convolutional neural networks (CNN) for bonafide/PA

feature representation. CNN, through stacked convolutional layers, captures critical semantic cues indicative of spoofing, such as the hand-hold contour of printed paper. However, training CNN models from scratch for face PAD poses challenges due to limited data and coarse supervision signals from BCE loss. To address these issues, recent research endeavors [12, 24] often fine-tune pretrained ImageNet models (e.g., ResNet18 and vision transformers) with BCE loss for face PAD. Transferring well-tuned model parameters, originally designed for large-scale generic object classification tasks, to face PAD data proves relatively efficient. Alternatively, some approaches modify BCE loss into a multi-class CE version to provide CNNs with more fine-grained and discriminative supervision signals. For instance, Xu et al. [72] rephrase face PAD as a fine-grained classification problem, supervising deep models with multi-class CE loss (e.g., bonafide, print, and replay). This approach explicitly represents intrinsic properties of bonafide samples and particular attack types. However, models supervised with multi-class CE loss still grapple with convergence issues due to class imbalance, and the global constraints in these supervision signals may lead to overfitting by neglecting crucial local spoof patterns.

2.3.2 Traditional Deep Learning Approaches with Pixel-wise Supervision

Compared to genuine facial images, presentation attacks (PAs) often exhibit distinct physical characteristics in local responses. For instance, 2D PAs like printed paper or electronic screens lack local geometric facial depth information, whereas genuine faces possess such depth information. Recognizing this, recent research efforts [3, 84, 70] have introduced pixel-wise pseudo depth labels (depicted in the fourth column of Fig. 4) to guide deep learning models. These labels prompt the models to predict genuine depth for bona fide samples while generating zero maps for PAs. To enhance accurate facial depth estimation using multi-level features, Atoum and colleagues [3] introduced the "DepthNet," a multi-scale fully convolutional network. Supervised with pseudo depth labels, the DepthNet generates comprehensive depth maps for genuine faces and coarse zero maps for 2D PAs, serving as interpretable decision evidence. To improve the fine-grained intrinsic feature representation, Yu et al. [84] devised a novel deep operator known as "central difference convolution" (CDC). CDC can replace conventional convolution in DepthNet without additional learnable parameters, forming the CDCN architecture. This CDC operator can be mathematically expressed as:

$$y(p_0) = \underbrace{\theta \cdot \sum_{p_n \in R} w(p_n) \cdot (x(p_0 + p_n) - x(p_0))}_{\text{Central Difference Convolution}} + \underbrace{(1-\theta) \cdot \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n)}_{\text{Vanilla Convolution}} \tag{2.1}$$

where x, y and w denote the input features, output features, and learnable convolutional weights, respectively. p_0 denotes the current location on both input and output feature maps while p_n enumerates the locations in neighbor region R. The hyperparameter θ in the range [0, 1] balances the contribution between intensity-level and gradient-level information. CDCN, known for its outstanding representation capacity for both low-level details and high-level semantic cues, is favored in pixel-wise supervision frameworks and widely adopted in the deep face Presentation Attack Detection (PAD) community.

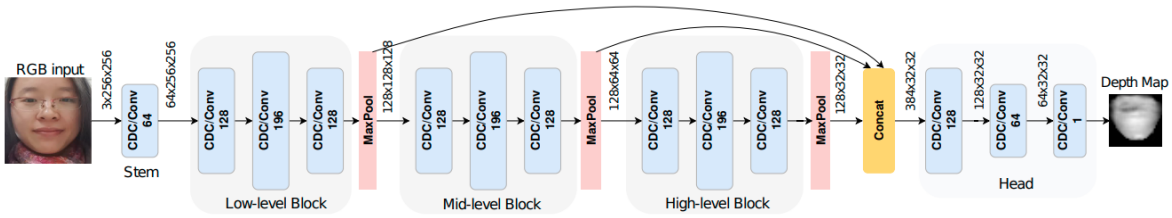


Figure 2.2: The multi-scale architecture of DepthNet [46] with vanilla convolutions ('Conv' for short) and CDCN [84] with CDC. Inside the blue block are the convolutional filters with 3x3 kernel size and their feature dimensionalities.

Considering the resource-intensive generation of pseudo depth maps and their limited relevance for 3D face PAs with realistic depth, binary mask labels (depicted in the second column of Fig. 4) offer a more practical alternative. Binary supervision entails providing deep embedding features with binary labels in each spatial position corresponding to the bona fide/PA distributions in original patches (e.g., 16×16). With binary mask supervision, models can identify PAs in the corresponding patches, irrespective of the attack type, and the results are spatially interpretable. Other auxiliary pixel-wise supervisory techniques, such as pseudo reflection maps (depicted in the third column of Fig. 4) and 3D point cloud maps (depicted in the last column of Fig. 4), are also employed. The former provides cues related to physical material reflection, while the latter encompasses denser 3D geometric cues. To learn more intrinsic material-related features, multi-head supervision was developed in [76] to simultaneously supervise PAD models with multiple pixel-wise labels, including pseudo depth, binary mask, and pseudo reflection. The corresponding pixel-wise loss functions can be formulated as:

$$
\begin{aligned}
L_{depth} &= \frac{1}{HW} \sum_{i \in H, j \in W} ||D_{(i,j)} - D_{gt(i,j)}||_2^2, \\
L_{reflection} &= \frac{1}{HWC} \sum_{i \in H, j \in W, c \in C} ||R_{(i,j,c)} - R_{gt(i,j,c)}||_2^2, \\
L_{binarymask} &= \frac{1}{HW} \sum_{i \in H, j \in W} -(B_{gt(i,j)} \log(B_{(i,j)}) + (1 - B_{gt(i,j)}) \log(1 - B_{(i,j)}))
\end{aligned}
\tag{2.2}
$$

where D_{gt}, R_{gt} and B_{gt} denote ground truth depth map, reflection map and binary mask map respectively and H, W and C represent the height, width, and channels of the maps. In summary, pixel-wise auxiliary supervision contributes to developing physically meaningful and explainable representations. However, it's worth noting that pseudo-auxiliary labels are often generated coarsely without human annotations, leading to occasional inaccuracies and noise in cases of partial attacks. For example, the binary mask for FunnyEye glasses attacks should ideally cover the eye regions rather than the entire face.

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

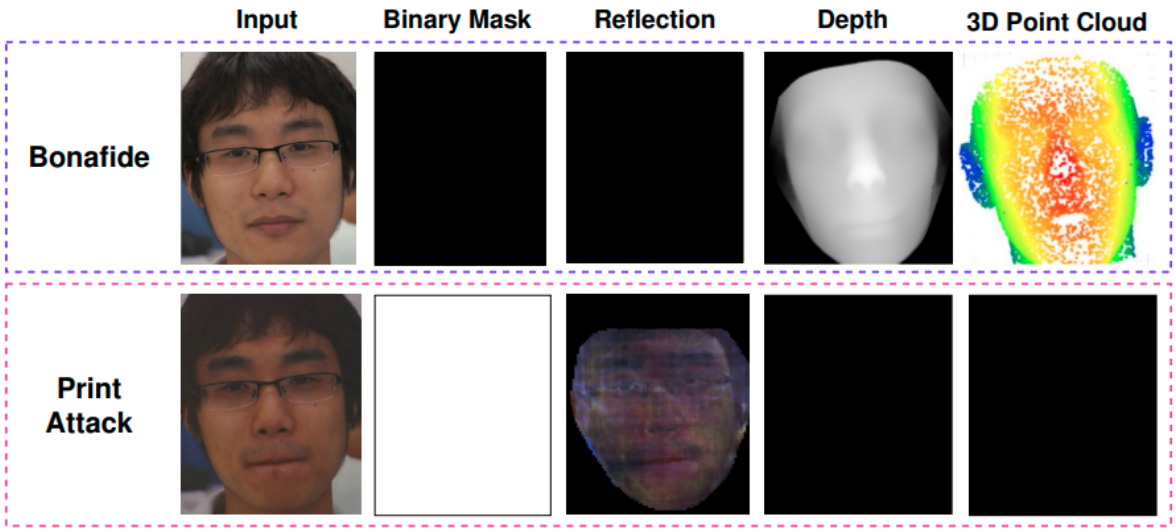


Figure 2.3: Visualization of pixel-wise supervision signals [77] including binary mask label [23], pseudo reflection maps [76], pseudo depth labels [84] and 3D point cloud maps [38] for face PAD.

[19, 8, 32, 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 [60, 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, 61], and representation learning [36, 37, 33]. Previous studies with GAN have shown that it is possible to generate high-resolution images up to 1024×1024 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 [32], which perturbs every pixel in the image and image is generated by gradient-based methods. LatentHSJA [35] 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 images of the target subject for training and inference.

Chapter 3

Whitebox Adversarial Attacks

In this chapter, we adopt a white-box approach to craft adversarial faces with specific, targeted outputs, leveraging the principles of contrastive learning. This methodology allows us to meticulously manipulate facial data to create images that can effectively deceive facial recognition systems. Our focus here is on developing a white-box attack strategy, which proves highly effective in generating these adversarial faces. This white-box approach involves training a neural network in a manner that enables it to generate perturbed instances of facial images, all while having access to the internal workings of a single, known face recognition system. The beauty of this method is that it imparts a deep understanding of the network, enabling it to create adversarial faces that exploit the vulnerabilities and weaknesses of the chosen face recognition system.

3.1 Introduction

White-box adversarial attacks pose a significant threat to face antispoofing systems. In a white-box attack scenario, the attacker has access to detailed knowledge about the underlying architecture, algorithms, and parameters of the antispoofing system. Armed with this information, they can craft highly targeted and effective attacks that exploit vulnerabilities in the system's design. These attacks often involve making subtle modifications to the input data, such as facial images or videos, in a way that is imperceptible to the human eye but can easily deceive the antispoofing system.

White-box adversarial attacks on face antispoofing systems can take various forms. Attackers may leverage optimization techniques to generate adversarial examples—modified inputs designed to fool the system. These adversarial examples can be carefully crafted to evade detection and appear as genuine faces, allowing unauthorized access or fraudulent activities. Moreover, the knowledge gained about the system's inner workings can aid attackers in identifying weak points or vulnerabilities that can be exploited to bypass the antispoofing mechanisms. As face recognition and antispoofing technologies continue to evolve, so does the sophistication of white-box attacks, making it crucial for developers and security experts to assess and fortify these systems against such threats continuously.

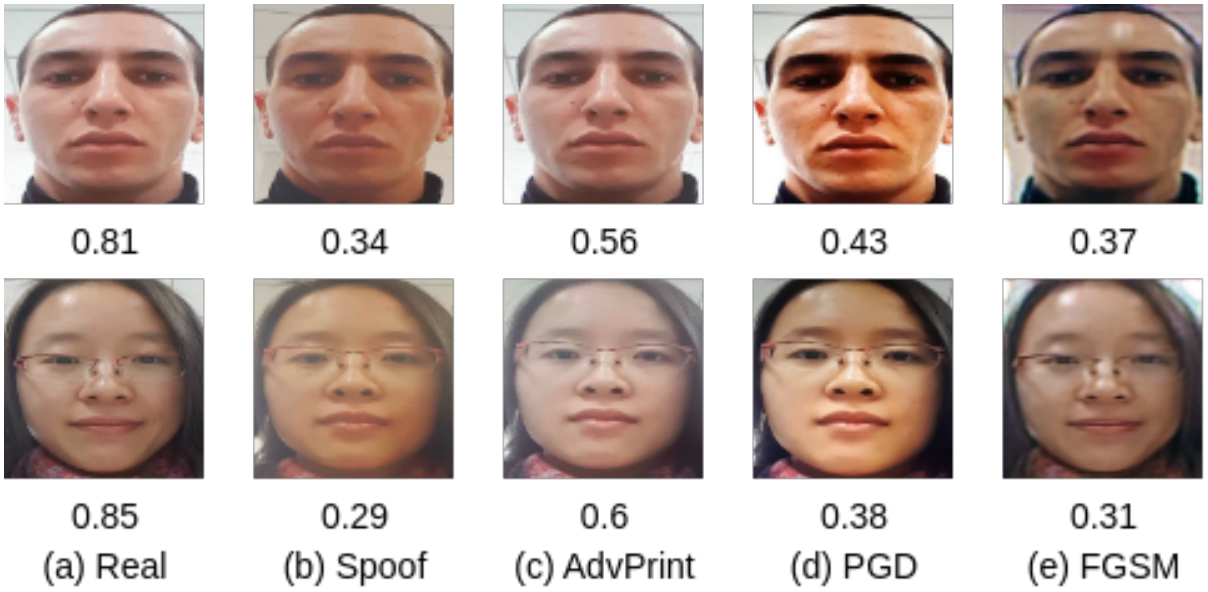


Figure 3.1: Input: real face image and corresponding synthesized adversarial images. (a) Two sample real-face images taken from OULU-NPU dataset [5] (b) the same subject's spoof image, adversarial images generated from (c) our proposed method, Adversarial Prints (d) PGD [32] (e) FGSM [19]. PAD scores are given below the images. A score above 0.4 (threshold @ 0.1 $\%$ False Accept Rate) indicates that the images belong to the same person and will be classified as real.

We propose "Adversarial Prints," a pioneering leap in the domain of physical adversarial spoof image generation. It exploits a Conditional Generative Adversarial Network (GAN) to automatically create adversarial face images specifically designed to deceive state-of-the-art spoof classifiers. What distinguishes our method is its incorporation of both major attack types, print attacks and replay attacks, within a single framework—a novel approach in the field. This comprehensive coverage ensures that our generated images effectively emulate the complexities of real-world presentation attacks.

Through rigorous ablation experiments, we have convincingly showcased the superiority of our generated images when compared to the existing state-of-the-art spoof classifiers. These images not only outperform previous attempts but also exhibit a remarkable degree of authenticity when assessed by real-world evaluation metrics. The crux of our methodology lies in its ability to replicate the mechanics of physical presentation attacks, such as print attacks while operating in the digital domain. When a live image is processed through our GAN, it effectively simulates the intricacies of a physical printing process, resulting in an adversarial image. This image possesses the characteristics of a printed representation, yet it defies conventional spoof classifiers by consistently being classified as genuine. Importantly, our method also maintains the integrity of the original individual's facial identity, ensuring that the adversarial image remains indistinguishable from the authentic face.

Once the training phase is complete, the resulting adversarial attacks can be seamlessly transferred to target any black-box face recognition system. This means that the knowledge gained during the training process, which is specific to one system, can be effectively applied to deceive other face recognition systems with unknown architectures or internal workings. This adaptability is a key strength of our approach, as it allows for the widespread applicability of our adversarial faces across various recognition systems. Our method achieves this by combining the power of the white box approach, the strategic insights from contrastive learning, and the versatility of the semi-whitebox attack.

The ultimate objective of Adversarial Prints is to bridge the gap between physical and digital adversarial attacks. By infusing the properties of physical presentation attacks into the digital domain, our approach not only enhances the effectiveness of spoofing techniques but also underlines the critical need for robust countermeasures and further advancements in facial recognition security. As adversaries become increasingly sophisticated, our research represents a significant stride towards fortifying facial recognition systems and protecting the privacy and security of individuals in the digital age.

The contributions of the chapter can be summarized as follows:

1. PAD-GAN, A network capable of simulating real-world physical processes such as printing or replay while preserving the identity of the input.

2. AdvPrint, A gradient-based attack strategy that can learn to perform an adversarial attack to generate visually realistic adversarial face images that synthesize a physical print and video-replay attack.

3. We explore a new physical adversarial attack strategy that can fool a PAD. The system detects and recognizes the facial region as a face while also preserving the identity of the subject.

4. We also propose to release an experimental dataset for experiment purposes synthesized using AdvPrint.

3.2 Method

The goal is to synthesize a print or replay face image that deceives both a PAD and a face recognition system by retaining the identity of the target, thus illegally gaining access to the system. AdvPrint consists of three components i) a simulator network that emulates printing and replaying input images, ii) a similarity-matching module that computes a similarity score between PAD representations of the simulated and real image, and iii) a gradient-based adversarial generation module that is supervised by the similarity-matching module.

Formulation of our Physical attack generation using an adversarially attacked Conditional Generative Adversarial Network in Section 3.1. First, we train PAD-GAN, a Conditional GAN [61] which learns

real to spoof translation. We describe itś architecture in Section 3.2. Next, we use FGSM [19] to perform an adversarial attack on the trained PAD-GAN. The attack generation strategy is explained in section 3.3.

3.2.1 Problem Formulation

In this section, we formulate the problem of adversarial attack generation on Face PAD Systems. Presentation attacks like printing/replay attacks include some form of a physical process like printing or displaying the real capture on a device which adds noise to the real capture. PAD relies on identifying these noise patterns introduced to perform classification. Equation 3.1 shows physical attack as a function of real input $\mathcal{I}_r \in \mathbb{R}^d$ and degradation function $\phi(\mathcal{I}_r)$ due to printing or displaying it on a screen.

$$\mathcal{I}_s = \mathcal{I}_r + \phi(\mathcal{I}_r) \tag{3.1}$$

Conventional adversarial attacks directly attack the PAD module to deceive its decisions but in case of physical attack, the intermediate physical medium need to be handled skillfully. Our Goal is to simulate the intermediate physical process, $\mathcal{F}_n(\mathcal{I})$ using neural networks and perform an attack the simulation to generate an adversarial image $\mathcal{I}_{adv}$. This constitutes a semi-whitebox attack on the PAD module.

$$\mathcal{I}_{adv} = \mathcal{I}_r + \mathcal{F}_n(\mathcal{I}) \tag{3.2}$$

We propose an adversarial attack on the simulation network guided by gradients of a PAD , $\nabla_{\mathcal{I}_r}\mathcal{L}(\mathcal{I}_r, w)$ and identity preserving loss, as an optimization problem with $\mathcal{J}(\mathcal{I}_r, \mathcal{I}_{adv})$ as a loss function

$$\begin{aligned} \min_{r} \quad & \nabla_{\mathcal{I}_r}\mathcal{L}(\mathcal{I}_r, w) \cdot r + \mathcal{L}(\mathcal{I}_r, w) \\ & \text{s.t.} \quad \|r\|_\infty \end{aligned} \tag{3.3}$$

of which solution is given by:

$$\mathcal{I}_{adv} = \mathcal{I}_r + \epsilon \cdot sign(\nabla_{I_r}\mathcal{L}(\mathcal{I}_r, \mathcal{I}_{adv})) \tag{3.4}$$

Here $\mathcal{L}(.)$ is a loss function which gives loss(obtained by computing similarity) between image embeddings.

3.2.2 PAD-GAN Architecture

We train an architecture derived from CycleGAN to learn the simulation from Real to spoof. The architecture is shown in fig:short-a.

Generators: The network consists of 2 based generators $\mathcal{G}_{rs}$ 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 corresponding discriminators $\mathcal{D}_r$ and $\mathcal{D}_s$ are supposed to distinguish between the captured examples and the generated samples by the generators. Discriminators are PatchGAN discriminators and project the

input to a patch-based matrix where each value in the matrix corresponds to the score of the particular patch's discriminative score. The network is trained using 2 types of losses:

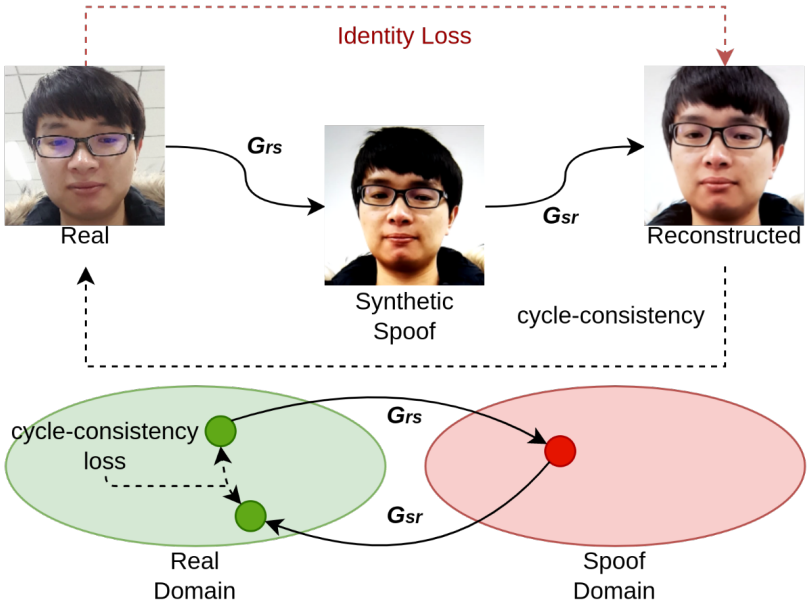


Figure 3.2: In the absence of constraints, the adversarial loss does not produce good images. A domain-appropriate output is enforced, but not recognizably identical inputs and outputs. The cycle consistency loss addresses this issue. By successively feeding the image through both generators, you should get back something similar to what you put in when you convert an image from one domain to the other. Here, $\mathcal{G}_{rs}$ is learning to synthesize replay and printed spoofs, which are fed through Discriminator $\mathcal{D}_r$ to check whether they are synthetic spoofs or captured spoofs. $\mathcal{G}_{sr}$ is learning to reconstruct real images, which are fed through Discriminator $\mathcal{D}_s$ to check whether they are reconstructed images or captured real images.

1. **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,

$$\begin{aligned}
\mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r) &= \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)} log[\mathcal{D}_s(\mathcal{I}_s)] + \\
&\quad \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)} log[1 - \mathcal{D}_s(\mathcal{G}_{rs}(\mathcal{I}_r))] \\
\mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r) &= \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)} log[\mathcal{D}_s(\mathcal{I}_s)] + \\
&\quad \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)} log[1 - \mathcal{D}_s(\mathcal{G}_{rs}(\mathcal{I}_r))] \\
\mathcal{L}_{adv} &= \mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r) + \\
&\quad \mathcal{L}_{adv}(\mathcal{G}_{rs}, \mathcal{D}_s, \mathcal{I}_r, \mathcal{D}_r)
\end{aligned} \tag{3.5}$$

2. **Cycle Consistency Loss**: Only adversarial loss leaves the learning unconstrained. Hence the Cycle Consistency Loss 4.3 is added as a regularization term to the generator's objectives. This loss is defined as,

$$\begin{aligned}
\mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) &= \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)}[\|\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)) - \mathcal{I}_r\|_1] \\
\mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s) &= \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)}[\|\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)) - \mathcal{I}_s\|_1] \\
\mathcal{L}_{cycle} &= \mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) + \mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s)
\end{aligned} \tag{3.6}$$

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

3. **Identity Regularizer:** The generated image should preserve the identity as in the input face. This would be a critical component in the adversarial attack generation. We add an identity-preserving regularization term. The CycleGAN, at every iteration, tries to preserve identity by minimizing the cosine similarity between face embeddings of the generated image and the input image that is defined as,

$$\begin{aligned}
\mathcal{L}_{identity}(\mathcal{G}_{rs}, \mathcal{G}_{sr}, \mathcal{I}_r, \mathcal{I}_s) &= \mathcal{F}[\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)), \mathcal{I}_r] \\
&\quad + \mathcal{F}[\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)), \mathcal{I}_s]
\end{aligned} \tag{3.7}$$

Finally, PAD-GAN is trained using the following objective,

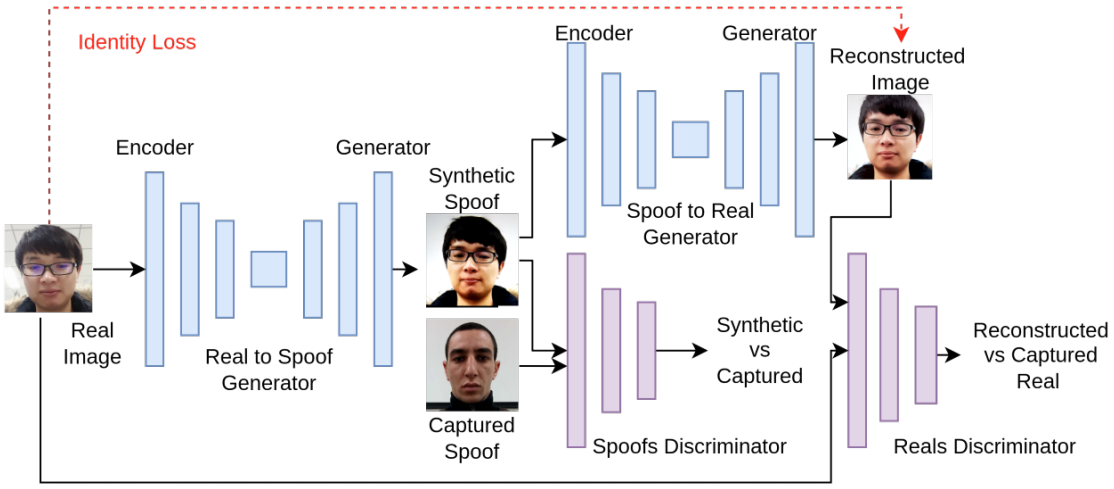
$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_{cycle} \times \mathcal{L}_{cyclic} + \lambda_{identity} \times \mathcal{L}_{identity} \tag{3.8}$$

3.2.3 Adversarial Attack Framework

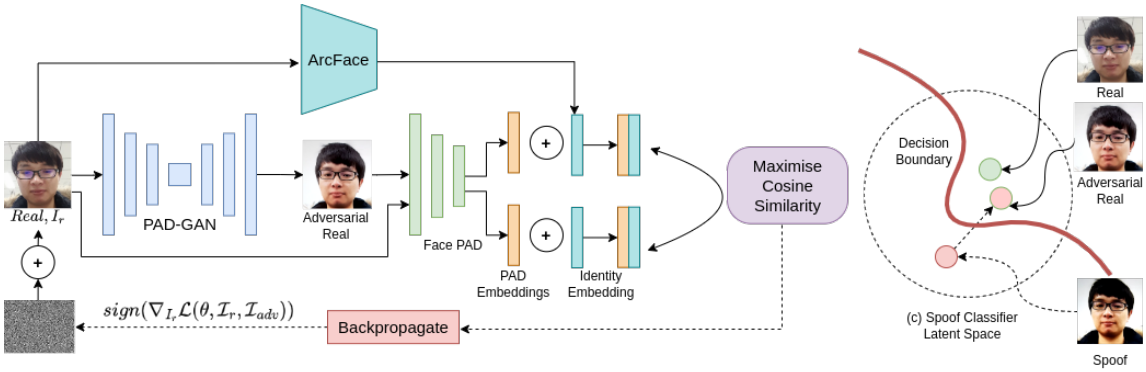
Perturbing the input image to generate an adversarial variant is a common strategy in standard attacks like FGSM [19]. We adopt an FGSM-based attack on the pretrained(Print Synthesiser(PAD-GAN)) which is guided by a self-supervised loss. FGSM is an efficient one-step attack which is formally defined as

$$\mathcal{I}_{adv} = \mathcal{I}_r + \epsilon \cdot sign(\nabla_{I_r} \mathcal{L}(\mathcal{I}_r, \mathcal{I}_{adv})) \tag{3.9}$$

where ϵ is the max perturbation, $L(\cdot)$ is the loss function and I_{adv} is the adversarial image generated by perturbing input image I_r.



((a)) Training of AdvPrint



((b)) Generating adversarial attacks

Figure 3.3: Architecture of AdvPrint comprises of two generators $\mathcal{G}_{rs}$ and $\mathcal{G}_{sr}$, two discriminators $\mathcal{D}_r$ and $\mathcal{D}_s$, and a face PAD. Synthesizing adversarial face images using AdvPrint consists of two stages: (a) Training of AdvPrint using CycleGAN, which when given a real image learns to generate a captured spoof. Identity loss is introduced as an identity regularizer to preserve the subject's identity (b) After training, this network synthesizes a printing attack and replay attack process. Here, we backpropagate the gradient of the loss which is calculated by maximizing the cosine similarities between PAD embeddings and identity embeddings (generated from a face recognition model, ArcFace [11]). When we input a real image through this network, we get an adversarial print image which, when passed through a face PAD would be classified as a real image. (c) shows how a generated adversarial image moves from the spoof space to the real space.

We want to perform an attack over the PAD-GAN to generate a spoof image that i) can deceive the PAD hence the PAD feature representations should lie close in the PAD latent space and ii) recognized correctly by a face detector, hence identity should be preserved. To achieve that, the loss in eq:fgsm is a negative cosine similarity between the generated and real image's embeddings.

The embedding used to compute the similarities comprises of 2 components i) Feature representation of the images from a PAD classifier V^{pad} is used to retain presentation attack features and ii) to the presentation attack features, identity features $V^{identity}$ are added using feature representations from a pretrained face recognition module. Both these representations are concatenated to form the final feature representation.

$$\begin{aligned} \mathcal{V}_r &= \mathcal{V}_r^{pad} || \mathcal{V}_r^{identity} \\ \mathcal{V}_{adv} &= \mathcal{V}_{adv}^{pad} || \mathcal{V}_{adv}^{identity} \end{aligned} \tag{3.10}$$

where $|\cdot|$ denotes the concatenation of the vectors. Cosine similarity is computed over these embeddings and the negative of this is used as the loss which is back-propagated.

$$\mathcal{J} = -similarity(\mathcal{I}_{adv}, \mathcal{I}_r) = \frac{\mathcal{I}_r \cdot \mathcal{I}_{adv}}{\|\mathcal{I}_r\| \cdot \|\mathcal{I}_{adv}\|} \tag{3.11}$$

The parameters of both the PAD model and face detector module are frozen and stacked one over another and, the gradients are back-propagated to the input real image. This perturbs the input image such that when it is printed using the PAD-GAN generates an adversarial spoof that resembles a real image.

3.3 Experiments

In this section, we first introduce the datasets used and the experimental setup. Then we evaluate the performance of our framework on PADs with adversarial training.

3.3.1 Dataset for Experiments

OULU-NPU is used as the base dataset to generate the adversarial images. With 55 subjects in the OULU-NPU dataset, we propose to release 10 frames per subject replicating print and replay attacks and generating adversarial variants of it. This can serve as a tool to experiment with different methods to defend against such attacks. We plan to extend the generation to other spoof types(mask, makeup, etc.). Sample images are provided as a reference in Fig 4.4. We have released a dataset for experimentation comprising adversarial images generated using our proposed method.

3.3.1.1

Evaluation protocols for the evaluation of the generalization capability of the face PAD methods, four protocols are used in OULU-NPU.

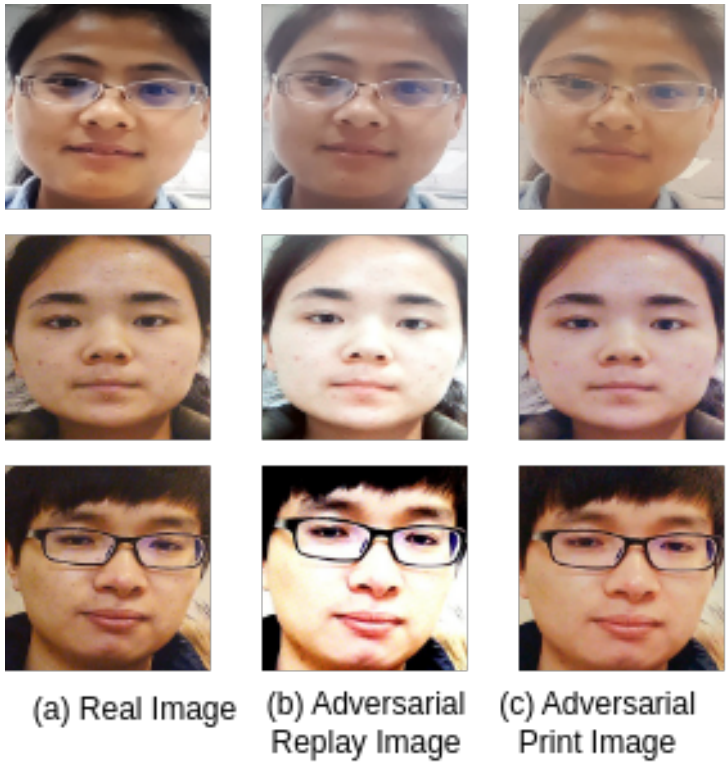


Figure 3.4: Sample images from the proposed dataset synthesized by AdvPrint. (a) represents a real image that is fed into AdvPrint, which generates two types of adversarial images: (b) adversarial replay image, which when printed and captured by a camera results in (c) adversarial print image.

- **Protocol I**:

  The first protocol is designed to evaluate the generalization of the face PAD methods under previously unseen environmental conditions, namely illumination and background scene. As the database is recorded in three sessions with different illumination condition and location, the train, development and evaluation sets are constructed using video recordings taken in different sessions.

- **Protocol II**:

  The second protocol is designed to evaluate the effect of attacks created with different printers or displays on the performance of the face PAD methods as they may suffer from new kinds of artifacts. The effect of attack variation is assessed by introducing a previously unseen print and video-replay attack in the test set.

- **Protocol III**:

  One of the critical issues in face PAD and image classification in general is sensor interoperability. To study the effect of the input camera variation, a Leave One Camera Out (LOCO) protocol is

used. In each iteration, the real and the attack videos recorded with five smartphones are used to train and tune the algorithms, and the generalization of the models is assessed using the videos recorded with the remaining one.

- **Protocol IV**:

  In the last and most challenging protocol, all above three factors are considered simultaneously and generalization of face PAD methods are evaluated across previously unseen environmental conditions, attacks and input sensors.

3.3.2 Experimental Setup

Figure 3.5: Experimental setup to perform physical attack on the deployed face presentation attack detection system.

In our experimental setup, we meticulously address the challenges posed by print and replay attack scenarios. For the printing phase, we employ an HP Smart Tank 580 printer, ensuring the authenticity of our simulated print attacks. To display and scrutinize the results, we utilize two distinct mediums—a MacBook Pro equipped with Intel Iris Plus Graphics 640 (1536 MB) and a Redmi K20 Pro featuring a

Super AMOLED display with HDR10 support. This diverse range of display devices allows us to assess the versatility of our adversarial attack method. During data collection, all images are captured within a proximity range spanning from 20cm to 40cm, replicating real-world usage scenarios and ensuring the relevance of our findings.

To rigorously validate the efficacy of our developed adversarial attack methodology, we integrate it into a user-friendly streamlit application. This application serves as a practical platform for real-time evaluation. It ingests live image feeds and promptly returns not only the predicted identity of the individual but also a binary classification of 'spoof' or 'live,' accompanied by a confidence score quantifying the prediction's certainty.

Our evaluation encompasses a meticulously curated test set comprising 300 images sourced from the OULU-NPU dataset, which features a diverse array of identities. Specifically, we sample 20 distinct identities from the OULU-NPU dataset. These sampled images are handpicked with utmost care to encompass the widest possible spectrum of variations in terms of facial expressions, lighting conditions, and poses. This comprehensive approach ensures that our evaluation is robust, accounting for many real-world scenarios and challenges.

3.3.3 Experimental Settings

While training AdvPrint, we are using the ADAM optimizer with β_1 = 0.5 and β_2 = 0.9, for the entire network. We are training our network on the OULU-NPU [5] dataset. Each mini-batch consists of 1 face image. We train AdvPrintfor 100 epochs with a fixed learning rate of 0.0002. We also use identity loss with parameters λ_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 AdvPrintwith ϵ = 0.1. All experiments are conducted using PyTorch.

3.3.4 Evaluation Metrics

By comparing our network against state-of-the-art baselines, we measure adversarial attacks' effectiveness through their attack success rate (ASR) as well as structural similarity (SSIM) [46].

The attack success rate (ASR) is computed as

$$ASR = \frac{\text{No. of attacks classified as real}}{\text{Total number of attacks}} \times 100\% \tag{3.12}$$

The Structural Similarity Index (SSIM) metric extracts 3 key features from an image: luminance, contrast, and structure. Here, Luminance is a comparison of μ_x and μ_y, Contrast is a comparison of σ_x and σ_y, and Structure is calculated by dividing the input signal with its standard deviation so that the result has unit standard deviation which allows for a more robust comparison. SSIM is calculated between adversarial image and real image:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{3.13}$$

Print Attack	AdvPrint	PGD	FGSM
Attack Success Rate %			
Protocol 1	**74.43**	23.70	19.96
Protocol 2	**57.24**	19.34	15.71
Protocol 3	**44.53**	18.25	12.30
Protocol 4	**22.98**	8.74	5.48
Structural Similarity	**0.73**	0.49	0.24

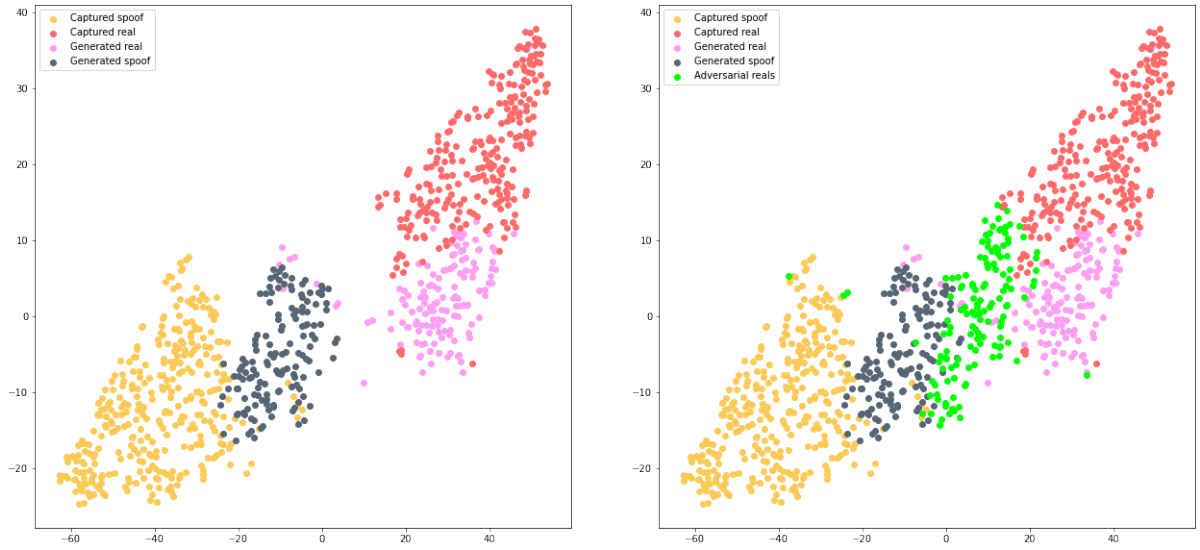
Table 3.1: Attack success rates and structural similarities between various protocols in the OULU-NPU dataset. The Protocols signify that PAD-GAN is trained on the corresponding train and test set in the protocol. The other 2 attacks are directly performed on the PAD-GAN and printed without handling the physical printing process

Here, x and y are the two images that are compared, μ_x, μ_y, σ_x, and σ_y are the means and variances of x and y, respectively. σ_{xy} is the covariance of x and y. Parameters $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ are constants to ensure stability when the denominator becomes zero, k_1, k_2 are constants, and $L = \left(2^{(no.\ of\ bits\ per\ pixel\ )} - 1\right)$ is the dynamic range for pixel values.

3.3.5 Comparison Studies

Replay Attack	AdvPrint	PGD	FGSM
Attack Success Rate %			
Protocol 1	**71.67**	20.44	17.58
Protocol 2	**53.22**	18.57	16.20
Protocol 3	**36.53**	16.61	11.74
Protocol 4	**19.98**	6.89	4.93
Structural Similarity	**0.68**	0.35	0.18

Table 3.2: Attack success rates and structural similarities between various protocols in the OULU-NPU dataset. The Protocols signify that PAD-GAN is trained on the corresponding train and test set in the protocol. The other 2 attacks are directly performed on the PAD-GAN and replayed without handling the physical printing process.



((a)) ((b))

Figure 3.6: t-SNE visualization of the datasets. (a) the t-SNE plot of feature representation from the PAD module for generated images by PAD-GAN for a mini-batch of images from the corresponding sets. (b) t-SNE plot of feature representation from the PAD module for adversarial images generated by PAD-GAN

Table 3.1 and 3.2 show quantitative evaluations of our proposed method on different protocols in the OULU-NPU dataset. We have mostly compared our methodś performance against the standard attach methods FGSM and PGD directly on PAD-GAN before generating the spoofs and then printing those adversarial images and presenting them to the PAD . The results clearly demonstrate i) simply generating an adversarial attack without taking into consideration the physical process doesn't prove to be a successful attack strategy ii) Our method performs fairly even on a difficult Protocol 3 with the Leave One Camera Out(LOCO) constraint. The method is able to learn to generate fairly on unseen presentation attacks and, iii) compared to replay attack, presentation attack is easier to simulate as replay involves complex ways to capture and present the image.

Table 3.3 demonstrates quantitative results across protocols. All the data from one of the protocols (in case of 2) and multiple (in case of multiple protocols) are used as the train set and the test set is curated from the other set. Such a kind of evaluation is commonly followed in standard PAD works [55, 30, 59, 24] to test the generalizability of the proposed approach.

Our method outperforms standard attack strategies on all possible combinations of protocols. It gives an idea of the generalizable capability of the proposed approach. Preserving both PAD features and identity features in the generation and attack strategies helps the attack to be quite robust even in unseen conditions.

Print Attack	AdvPrint	PGD	FGSM
Attack Success Rate %			
Protocol 1 & 2	**64.76**	21.54	18.74
Protocol 2 & 3	**61.83**	17.47	13.09
Protocol 3 & 4	**46.87**	12.21	8.45
Protocol 1 & 4	**51.61**	13.70	10.20
Protocol 1, 2, 3	**56.29**	16.81	14.59
Protocol 1, 2, 3, 4	**41.43**	6.22	4.86
Structural Similarity	**0.76**	0.52	0.34

Table 3.3: Attack success rates and structural similarities in cross protocols evaluation in the OULU-NPU dataset. The Protocols signify that PAD-GAN is trained on one of the corresponding protocol's entire data and test and validation set are curated from the other protocol. The other 2 attacks are directly performed on the PAD-GAN and printed without handling the physical printing process

3.3.6 Ablation Studies

We undertake a series of ablation studies with the primary objective of illustrating the critical significance of incorporating identity embedding within the similarity computation framework, as depicted in Fig 3.7. The implications of this inclusion are profound, directly influencing the nature of perturbations introduced into the generated images. It is important to highlight that these perturbations are not arbitrary but rather meticulously designed to manipulate the image in a manner that aids in evading detection by face recognition systems.

The role of identity embedding within this context cannot be overstated. Without incorporating identity information, the generated images experience a significant loss of critical identity-related details. This loss is highly detrimental, as it adversely affects the recognition performance of the adversarial attacks. In simpler terms, when identity information is omitted from the computation, the generated adversarial images deviate substantially from the original identity, leading to a misalignment between the generated image and the expected identity. Consequently, recognition scores go down, and the adversarial attack's effectiveness is severely compromised. This highlights the pivotal role that identity embedding plays in the overall success of our approach, emphasizing the need for a holistic and comprehensive consideration of identity information in the adversarial image generation process.

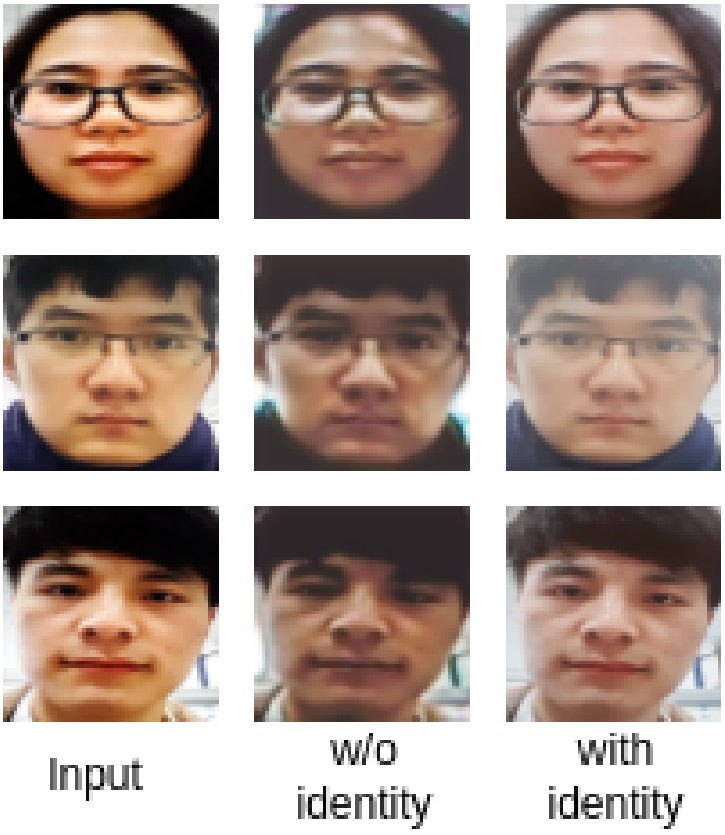


Figure 3.7: Variants of AdvPrint trained without identity loss and with identity loss. Images trained without identity loss affects the recognition score.

3.4 Conclusion

Attacking Presentation Attack Detection (PAD) systems in the physical world, like using a printed photo, is often more challenging than fooling them digitally. To tackle this challenge, we've developed a unique method called AdvPrint. This method allows us to create special deceptive images that mimic the appearance of printed photos and replayed videos. AdvPrint is highly effective. It enables us to generate these tricky images that can successfully bypass the anti-spoofing measures and trick the face recognition system. What's more, our method ensures that the identity of the person in the image remains intact. In simpler terms, we've found a way to make photos that look real but can outsmart the system, all while keeping the person's identity unchanged.

Chapter 4

Blackbox Adversarial Attacks

In this chapter, we shift our focus to a black-box adversarial attack strategy, departing from the white-box approach previously discussed. Our aim now is to craft adversarial faces with specific, targeted outputs while operating under the constraints of limited knowledge about the internal workings of the target face recognition system. To achieve this, we harness the power of Conditional Generative Adversarial Networks (GANs) and Generative Adversarial Networks (GANs). These sophisticated deep learning architectures enable us to generate adversarial images that can effectively deceive facial recognition systems.

4.1 Introduction

Face recognition technology has become a part of our daily lives, offering unparalleled convenience and security. However, this widespread adoption has also exposed these systems to new challenges, particularly the need to safeguard them from adversarial attacks. Among these threats, black-box adversarial attacks stand out as a significant menace to the reliability of face-presentation attack detection systems. These attacks are notorious for their capacity to subvert the effectiveness of facial antispoofing models, all while operating with limited knowledge of the model's internal workings. This paper ventures into the intriguing domain of black-box adversarial attacks, with a specific focus on their impact on face antispoofing models. We introduce an inventive approach that harnesses Conditional Generative Adversarial Networks (GANs) in combination with a Spoof Decomposition Network and Expectation over-transform techniques to tackle this challenge head-on.

In an era where facial recognition technology is becoming increasingly prevalent, the convenience and security it offers are undeniable. However, this ubiquity has given rise to a pressing concern: how can we shield these systems from adversarial attacks? Among the various threats, black-box adversarial attacks pose a particularly daunting challenge. These attacks are distinguished by their ability to subvert the defenses of facial antispoofing models, all without an intricate understanding of how these models work internally. This paper takes a deep dive into the captivating world of black-box adversarial attacks, with a specific emphasis on their impact on face antispoofing models. Within these pages, we introduce an

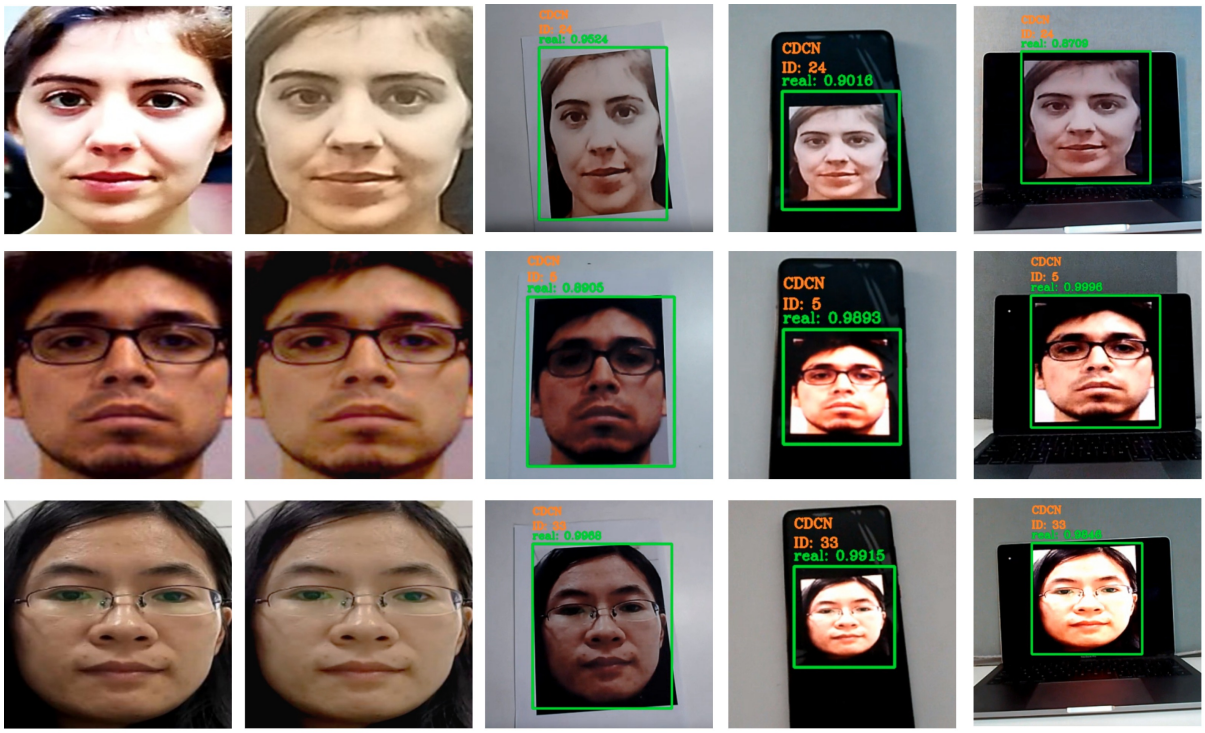


Figure 4.1: Example live images and corresponding adversarial images generated by AdvGen. First Column: live images from presentation attack datasets, second column: the corresponding adversarial images generated by AdvGen, third column: the predicted class along with the confidence score and recognized 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.

innovative approach that harnesses the power of Conditional Generative Adversarial Networks (GANs), working in tandem with a Spoof Decomposition Network and Expectation over Transform techniques. Together, these elements form a robust strategy to address the complexities of this problem.

Black-box adversarial attacks pose intricate challenges when they target face-presentation attack detection systems. In these attacks, the assailant operates with only partial knowledge of the inner workings of the target model. This scenario mirrors real-world situations where adversaries attempt to trick facial recognition systems without comprehensively understanding the system's architecture. Crafting adversarial inputs that can successfully deceive the system in such circumstances becomes a daunting task. While white-box attacks, which rely on full knowledge of the model, have been extensively studied, the realm of black-box attacks demands innovative strategies. These strategies must be able

to bypass facial antispoofing systems stealthily and adapt to their defenses. To tackle this captivating challenge, we present AdvGen, a pioneering method that harnesses the capabilities of Conditional Generative Adversarial Networks (GANs) and sophisticated loss functions to generate adversarial inputs. Remarkably, our approach accomplishes this while operating within the constraints of limited knowledge about the internal workings of the target model.

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 AdvPrint, 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]. (Fig 4.1).

4.2 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.

4.2.1 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.

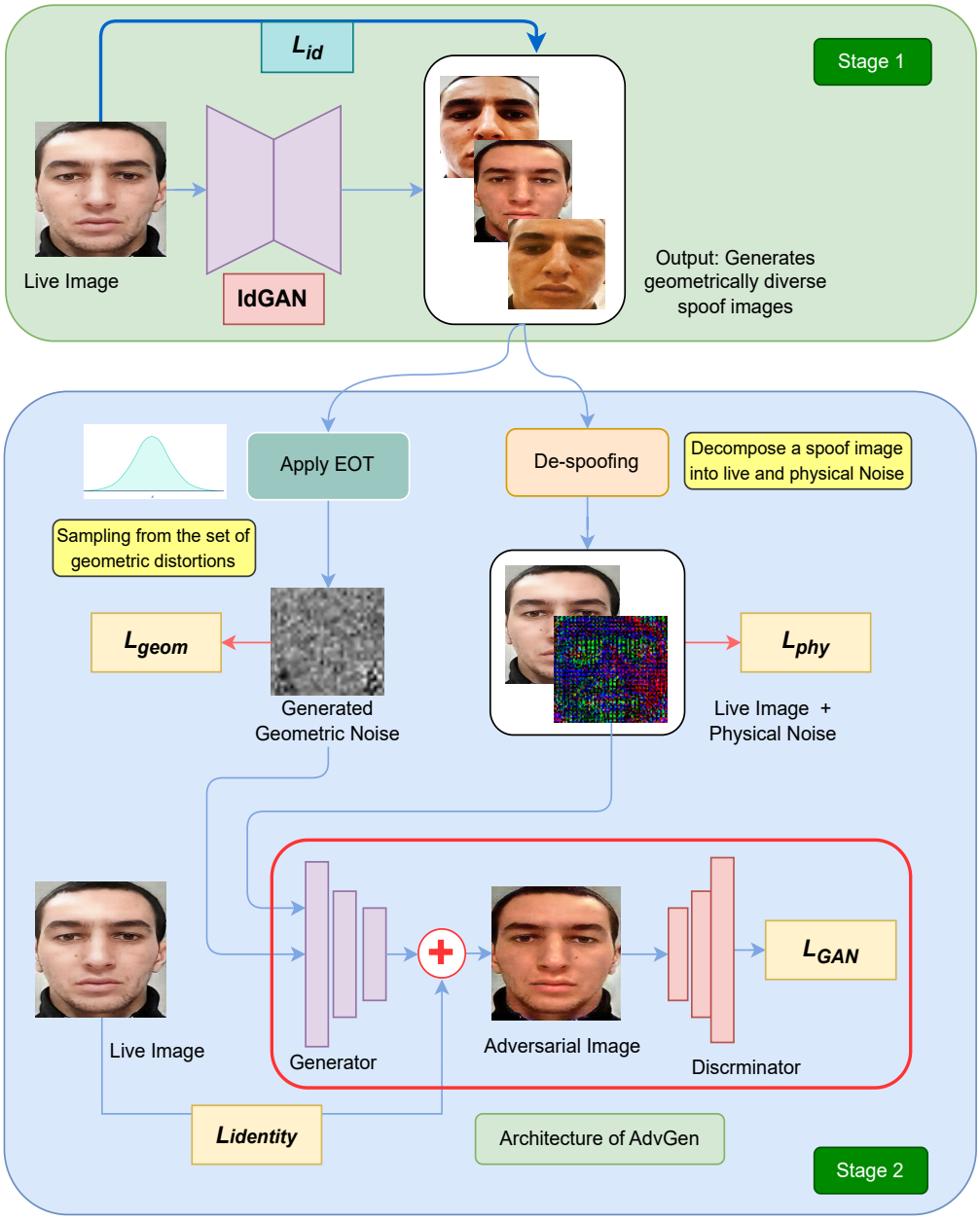


Figure 4.2: Synthesizing adversarial face images using AdvGen consists of two stages: **Stage 1:** Training of *IdGAN* which, given a live image, learns to generate geometrically diverse spoof images. These generated images produced by *IdGAN* simulate printing and replay. *Identity loss* is introduced 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.

The process of generating an adversarial perturbation δ involves solving the following optimization problem:

$$\begin{aligned} \arg\min_{\delta} \mathcal{L}(f(\mathcal{I}+\delta), l_{target}), \\ \text{subject to } \parallel \delta \parallel_p < \epsilon \end{aligned} \tag{4.1}$$

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

$$\arg\min_{\delta} \mathcal{L}(f(\mathcal{I}+\delta), l_{target}) + \lambda \parallel \delta \parallel_p \tag{4.2}$$

where λ is a hyper-parameter that controls the regularization of the distortion $\parallel \delta \parallel_p$.

In a physical domain setting, we denote a spoof image as $\mathcal{I}_s$. The spoof detection network is not fed directly with $\mathcal{I}_{adv} = \mathcal{I}_s + \delta^*$ (δ^* is the optimal digital perturbation obtained by using Eq. 4.2 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 ρ^*.

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 ρ can be obtained by optimizing the average loss over $\mathcal{T}$,

$$\arg\min_{\rho} \mathbb{E}_{t\sim\mathcal{T}}[\mathcal{J}(f_s(t(\mathcal{I})+\rho), l_{target})] + \lambda \parallel \rho \parallel_p \tag{4.3}$$

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 ρ obtained using Eq. 4.3.

4.2.2 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 $\mathcal{G}_{rs}$ 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

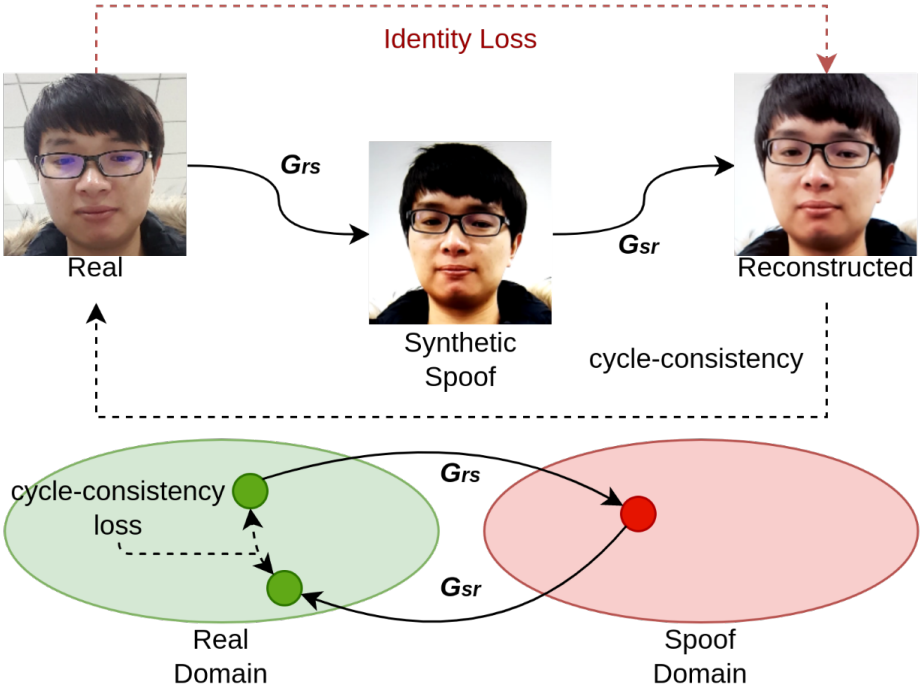


Figure 4.3: Loss terms used to train **IdGAN**. along with conventional $\mathcal{L}_{adv}$ and $\mathcal{L}_{cycle}$, we introduce a $\mathcal{L}_{id}$ to preserve identity in the generated image, which is a crucial step for the stage 2.

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,

$$\begin{aligned}\mathcal{L}_{id}(\mathcal{G}_{rs},\mathcal{G}_{sr},\mathcal{I}_r,\mathcal{I}_s) = \mathbb{E}_x[1-\mathcal{F}[\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)),\mathcal{I}_r]] \\ + \mathbb{E}_x[1-\mathcal{F}[\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)),\mathcal{I}_s]]\end{aligned} \tag{4.4}$$

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,

$$\begin{aligned}\mathcal{L}_{adv}(\mathcal{G}_{rs},\mathcal{D}_s,\mathcal{I}_r,\mathcal{D}_r) = \mathbb{E}_{\mathcal{I}_s\sim p_{data}(\mathcal{I}_s)}log[\mathcal{D}_s(\mathcal{I}_s)]+ \\ \mathbb{E}_{\mathcal{I}_r\sim p_{data}(\mathcal{I}_r)}log[1-\mathcal{D}_s(\mathcal{G}_{rs}(\mathcal{I}_r))] \\ \mathcal{L}_{adv}(\mathcal{G}_{sr},\mathcal{D}_r,\mathcal{I}_s,\mathcal{D}_s) = \mathbb{E}_{\mathcal{I}_r\sim p_{data}(\mathcal{I}_r)}log[\mathcal{D}_r(\mathcal{I}_r)]+ \\ \mathbb{E}_{\mathcal{I}_s\sim p_{data}(\mathcal{I}_s)}log[1-\mathcal{D}_r(\mathcal{G}_{sr}(\mathcal{I}_s))] \\ \mathcal{L}_{adv} = \mathcal{L}_{adv}(\mathcal{G}_{rs},\mathcal{D}_s,\mathcal{I}_r,\mathcal{D}_r)+ \\ \mathcal{L}_{adv}(\mathcal{G}_{sr},\mathcal{D}_r,\mathcal{I}_s,\mathcal{D}_s)\end{aligned} \tag{4.5}$$

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 Fig 4.3. This loss is defined as,

$$\begin{aligned}\mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) &= \mathbb{E}_{\mathcal{I}_r \sim p_{data}(\mathcal{I}_r)}[\|\mathcal{G}_{sr}(\mathcal{G}_{rs}(\mathcal{I}_r)) - \mathcal{I}_r\|_1] \\ \mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s) &= \mathbb{E}_{\mathcal{I}_s \sim p_{data}(\mathcal{I}_s)}[\|\mathcal{G}_{rs}(\mathcal{G}_{sr}(\mathcal{I}_s)) - \mathcal{I}_s\|_1] \\ \mathcal{L}_{cycle} &= \mathcal{L}_{cyc}(\mathcal{G}_{rs}, \mathcal{I}_r) + \mathcal{L}_{cyc}(\mathcal{G}_{sr}, \mathcal{I}_s)\end{aligned} \tag{4.6}$$

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

Finally, IdGAN is trained using the following objective,

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_{cycle} \times \mathcal{L}_{cycle} + \lambda_{id} \times \mathcal{L}_{id} \tag{4.7}$$

4.2.3 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. Fig 4.2 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:

$$\mathcal{L}_{phy} = \mathbb{E}_x[\max(\epsilon_1, \|\mathcal{P}hy(x)\|_2)] \tag{4.8}$$

ϵ_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

$$\mathcal{L}_{geom} = \mathbb{E}_x[\max(\epsilon_2, \|\mathcal{G}eom\|_2)] \tag{4.9}$$

ϵ_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,

$$\mathcal{L}_{identity} = \mathbb{E}_x[1 - \mathcal{F}(x, x + \mathcal{G}(x))] \tag{4.10}$$

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:

$$\mathcal{L}_{GAN} = \mathbb{E}_x[\log \mathcal{D}(x)] + \mathbb{E}_x[\log(1 - \mathcal{D}(x + \mathcal{G}(x))] \tag{4.11}$$

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

$$\begin{aligned} \mathcal{L} = \lambda_{phy} \times \mathcal{L}_{phy} + \lambda_{geom} \times \mathcal{L}_{geom} + \\ \lambda_{identity} \times \mathcal{L}_{identity} + \lambda_{GAN} \times \mathcal{L}_{GAN} \end{aligned} \tag{4.12}$$

4.3 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:

4.3.1 Datasets and Baselines

We train AdvGen on OULU-NPU [5] and test on SiW [57], MSU-MFSD [48], Replay-Attack [9] and OULU-NPU [5][1] datasets. **OULU-NPU [5]** face presentation attack detection database contains 4,950

[1]We train on training and validations sets of Protocol 1 of OULU-NPU and test on the corresponding test set

Attack Success Rate on OULU-NPU(%) and SSIM after attack					
	BIM [28]	EOT [2]	RP_2 [16]	D2P [21]	**Ours**
CDCN [55]	41.19	55.82	63.12	68.37	**81.02**
CDCNpp [58]	37.47	51.61	59.39	64.26	**78.22**
C-CDN [54]	38.38	51.58	60.83	65.49	**79.34**
DC-CDN [54]	39.95	53.83	61.36	66.03	**80.55**
SSAN-M [47]	40.06	52.02	61.40	65.27	**80.42**
SSAN-R [47]	34.54	49.83	57.03	61.79	**75.15**
DBMNet [22]	38.78	52.69	59.89	62.74	**79.63**
STDN [31]	40.92	53.93	61.67	63.29	**80.98**
Meta-FAS [7]	35.38	47.67	57.25	59.53	**76.19**
De-Spoofing [24]	46.44	58.43	65.41	68.66	**84.67**
SSIM in [0,1]	***0.64***	***0.38***	***00.32***	***0.45***	***0.98***

Table 4.1: Comparison of attack success rates on different models and ours using four different datasets.

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-artphysical 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 [32], 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 4.1.

4.3.2 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

$$ASR = \frac{\text{No. of attacks classified as real}}{\text{Total number of attacks}} \times 100\% \tag{4.13}$$

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]:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{4.14}$$

Here, x and y are the two images that are compared, μ_x, μ_y, σ_x, and σ_y are the means and variances of x and y, respectively. σ_{xy} is the covariance of x and y. Parameters $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ are constants to ensure stability when the denominator becomes zero, k_1, k_2 are constants, and $L = \left(2^{(no.\ of\ bits\ per\ pixel\ )} - 1\right)$ is the dynamic range for pixel values.

4.3.3 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.

4.3.4 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. We also use identity loss with parameters λ_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 ϵ = 0.1. All experiments are conducted using PyTorch.

Figure 4.4: Experimental setup to perform physical attack on the deployed face presentation attack detection system.

4.4 Results and Analysis

4.4.1 Effectiveness in 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 4.2. 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.

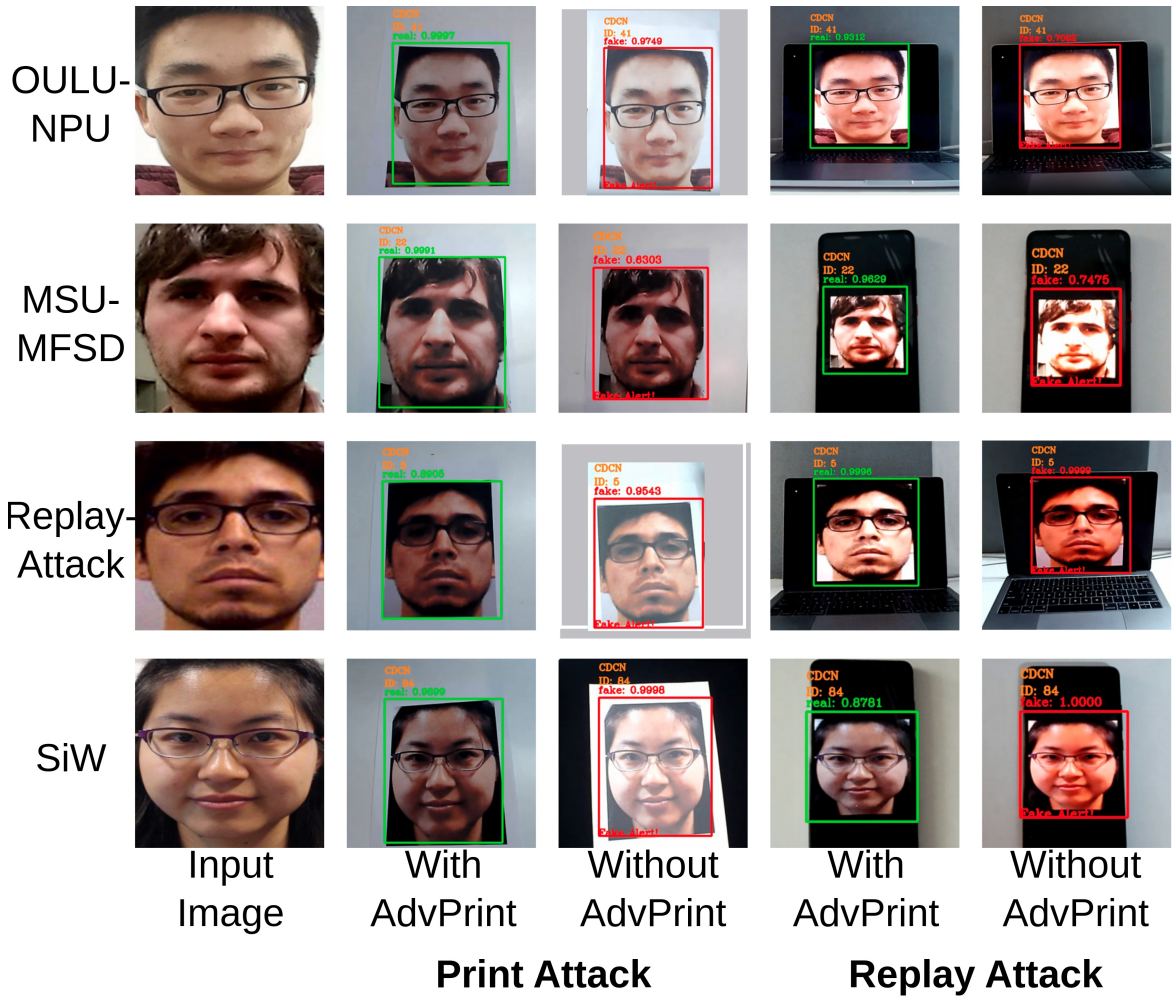


Figure 4.5: Experimental pipelines to evaluate the performance of the attacks. (a) shows the pipeline used when we attack a PAD in the digital domain, and (b) shows our testing pipeline in a physical world setting.

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 [32]	98.63	36.42
AdvGen	**100**	**81.02**

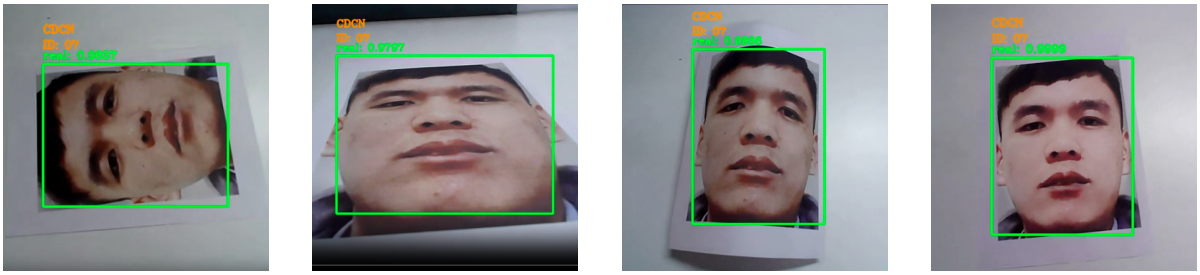
Table 4.2: Performance of state-of-the-artadversarial attack methods in the digital and physical domain.

4.4.2 Comparison Studies

In Table 4.1, 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.

4.4.3 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. Fig 4.6 demonstrates the effectiveness of our methods through various geometric distortions.



(a) Rotation (b) View Point (c) Physical Distortion (d) Brightness

Figure 4.6: 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.

4.4.4 Ablation Study

(a) Input Image (b) without L_{GAN} (c) without L_{phy} (d) without L_{geom} (e) without $L_{identity}$ (f) with all

Figure 4.7: Variants of AdvGen trained without GAN loss, physical perturbation hinge loss, geometric distortion hinge loss, and identity loss, respectively.

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 Fig 4.7. 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 geometric 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.

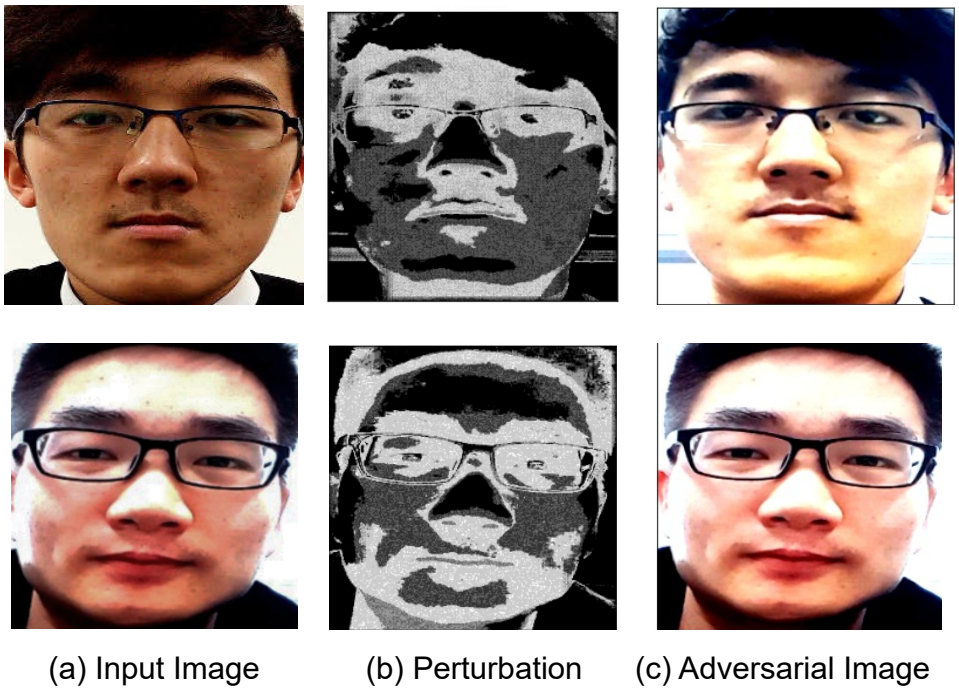


Figure 4.8: Visualization of the generated perturbation. (a) shows the input image, which can be live or spoof, (b) the locations of the input face resulting in perturbation we get from AdvGen , and (c) shows the final adversarial image.

4.5 Explainability

The primary objective of our perturbation method is to craft a perturbation mask that delicately preserves the identity of the subject while effectively making the spoofing attempt appear genuine. To achieve this, we focus on a profound understanding of the facial regions responsible for an individual's unique identity. These key identity-bearing regions, including the eyebrows, eyeballs, and nose, remain largely untouched by our perturbation process. By safeguarding these regions, we ensure that the subject's identity remains intact, even as we manipulate other areas to create a convincing presentation attack.

Our approach extends beyond preserving identity; it also strategically targets the areas of the face where face presentation attack detectors tend to concentrate their scrutiny. These detectors often analyze low-frequency regions such as the forehead and cheeks to identify patterns indicative of spoofing attempts. Therefore, our method introduces noise deliberately to these specific facial regions, enhancing the effectiveness of our attack. The elegance of our approach lies in its ability to identify and perturb these critical areas of the face, rendering our attack both successful and deceptively authentic.

This research serves as a pivotal step in unraveling the vulnerabilities of state-of-the-art face presentation attack detection models. By shedding light on the significance of low-frequency regions on the face in the context of face antispoofing, we pave the way for more targeted and robust face presentation

attack detection models. Future research can place specific emphasis on fortifying these regions against attacks, ultimately advancing the reliability and security of facial recognition systems. Moreover, our work underscores the importance of explainability in these models, providing valuable insights into how they function and how they can be enhanced to address the evolving landscape of face presentation attacks.

4.6 Conclusion

We develop a physical attack strategy aimed at a Convolutional Neural Network (CNN)-based face authentication system that attack a robust anti-spoofing module. What sets our study apart is the formidable challenge we tackle: attacking an anti-spoofing face authentication system in the physical domain. This endeavor presents unique and intricate difficulties differentiating it from attacking systems in other application scenarios. Our pioneering framework, which we've aptly named AdvGen , represents a significant leap in adversarial attack research. Its primary function is to generate adversarial images that impeccably mimic the intricacies of a printing and replay procedure.

Through rigorous experimentation and a series of compelling findings, we provide compelling evidence that AdvGen is more than just a theoretical concept. It's a practical solution with real-world implications. Our method excels at producing synthetic adversarial prints that possess the remarkable capability to outsmart Presentation Attack Detectors (PADs) and deceive a face recognition system. What's truly remarkable is that these adversarial prints, crafted with precision by AdvGen , accomplish this feat while maintaining the unmistakable identity of the individual in question. In essence, we're not merely introducing a new concept; we're demonstrating its tangible effectiveness in the complex landscape of facial recognition security.

Chapter 5

Future Work and 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 AdvPrint, can produce adversarial images that mimic a printing and replay procedure. Through experimentation, we have demonstrated that AdvPrintcan 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.

Focusing on the print and reply attack scenario, we proposed AdvPrint, 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.

Related Publications

- **AdvGen: Physical Adversarial Attack on Face Presentation Attack Detection Systems**, Sai Amrit Patnaik, Shivali Chansoriya, Anil K. Jain, Anoop Namboodiri. **In IJCB:** International Joint Conference on Biometrics 2023, Ljubljana, Slovenia.
- **Adversarial Prints: A Face Presentation Attack using Adversarial Images**, Sai Amrit Patnaik, Shivali Chansoriya, Anoop Namboodiri. **Under review**.

Bibliography

[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] 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] 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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

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

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[27] H. Kim. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020.

[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.

[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.

[30] Y. Liu, A. Jourabloo, and X. Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[31] 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.

[32] 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.

[33] 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.

[34] 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.

[35] D. Na, S. Ji, and J. Kim. Unrestricted black-box adversarial attack using gan with limited queries. *arXiv preprint arXiv:2208.11613*, 2022.

[36] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[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.

[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.

[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.

[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.

[41] F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

[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.

[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.

[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.

[45] X. Wang, K. He, C. Song, L. Wang, and J. E. Hopcroft. At-gan: An adversarial generator model for non-constrained adversarial examples. *arXiv preprint arXiv:1904.07793*, 2019.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[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.

[59] J. Zhao, Y. Cheng, Y. Xu, L. Xiong, J. Li, F. Zhao, K. Jayashree, S. Pranata, S. Shen, J. Xing, S. Yan, and J. Feng. Towards pose invariant face recognition in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

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

[61] 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.