International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02136-1

FD-GAN: Generalizable and Robust Forgery Detection via Generative Adversarial Networks

Nanqing Xu[1] · Weiwei Feng[1] · Tianzhu Zhang[1,2] · Yongdong Zhang[1]

Received: 23 September 2023 / Accepted: 30 May 2024

Abstract
Generalization across various forgeries and robustness against corruption are pressing challenges of forgery detection. Although previous works boost generalization with the help of data augmentations, they rarely consider the robustness against corruption. To tackle these two issues of generalization and robustness simultaneously, in this paper, we propose a novel forgery detection generative adversarial network (FD-GAN), which consists of two generators (a blend-based generator and a transfer-based generator) and a discriminator. Concretely, the blend-based generator and the transfer-based generator can adaptively create challenging synthetic images with more flexible strategies to improve generalization. Besides, the discriminator is designed to judge whether the input is synthetic and predicts the manipulated regions with a collaboration of spatial and frequency branches. And the frequency branch utilizes Low-rank Estimation algorithms to filter out adversarial corruption in the input for robustness. Furthermore, to present a deeper understanding of FD-GAN, we apply theoretical analysis on forgery detection, which provides some guidelines on data augmentations for improving generalization and mathematical support for robustness. Extensive experiments demonstrate that FD-GAN exhibits better generalization and robustness. For example, FD-GAN outperforms 14 existing methods on 3 benchmarks in generalization evaluation, and it separately improves the performance against 6 kinds of adversarial attacks and 7 types of distortions by 16.2% and 2.3% on average in robustness evaluation.

Keywords Face forgery · Forgery detection · Generative adversarial networks

1 Introduction

With the rapid development of deep learning, generative models like Variational Auto-Encoders (VAEs) (Kingma & Welling, 2013), Generative Adversarial Networks (GANs) (Goodfellow et al., 2014a), and Diffusion Models (Ho et al., 2020) have been present in many fields and made significant progress. Along with these generative methods, face forgery has become a popular topic in recent research, such as FaceSwap. However, by synthesizing realistic faces to fool human beings, face forgery techniques expose risks and may be used for nefarious purposes, such as fake news and financial fraud. Therefore, to eliminate these potential threats, forgery detection has become a significant research direction, where plenty of efforts (Li et al., 2021b; Luo et al., 2021; Chen et al., 2022; Liu et al., 2021; Qian et al., 2020) are spurred to face forgery detection.

Current forgery detectors can achieve excellent performance when the training and testing forgeries are from the same datasets and deepfake techniques, dubbed as "in-dataset" settings since they focus on method-specific

Communicated by Segio Escalera.

Nanqing Xu, Weiwei Feng have contributed equally to this work.

✉ Tianzhu Zhang
tzzhang@ustc.edu.cn

Nanqing Xu
xnq@mail.ustc.edu.cn

Weiwei Feng
fengww@mail.ustc.edu.cn

Yongdong Zhang
zhyd73@ustc.edu.cn

[1] School of Information Science and Technology, University of Science and Technology of China, JinZhai Road Baohe District, Hefei 230026, Anhui, China

[2] Deep Space Exploration Lab, Hefei, Anhui, China

Published online: 26 June 2024

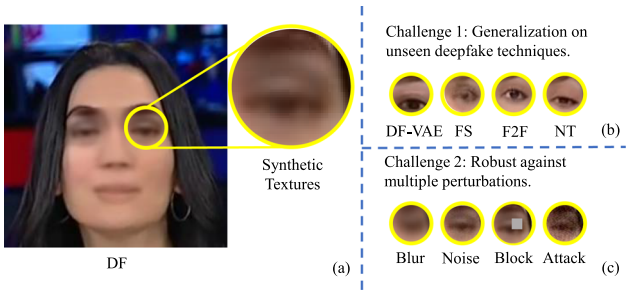

Fig. 1 Challenges in forgery detection. **a** Current forgery detectors usually make predictions depending on the method-specific synthetic textures (e.g., Deepfakes (DF)). **b** Unseen deepfake techniques hold quite different synthetic textures (e.g., DF-VAE (Jiang et al., 2020a), Face2Face (F2F) (Thies et al., 2016), FaceSwap (FS), and Neural-Textures (NT) (Thies et al., 2019)), which causes the challenge of generalization across datasets. **c** Perturbations poison these textures (e.g., blur, noise, block, and adversarial attacks (Szegedy et al., 2014; Goodfellow et al., 2014b; Li et al., 2021a)), which leads to the challenge of robustness against corruption

synthetic textures (Fig. 1a) [1]. Although these detectors can achieve good performance under "in-dataset" settings, for practical usage, there are still two inevitable challenges for face forgery detection. **(1)** The first challenge is the generalization of forgery detectors across various datasets ("cross-dataset" settings), where the testing forgeries are created by unseen advanced deepfake methods. Due to the gap in synthetic textures between the training and testing data generated by various deepfake methods, as shown in Fig. 1b, existing forgery detectors usually suffer from poor detection performance. **(2)** The other challenge is the robustness of forgery detectors. Since there are many uncertainties in the real world, natural media data is often disturbed by common corruption like blur, compression, and designed adversarial attacks (Szegedy et al., 2014; Goodfellow et al., 2014b; Li et al., 2021a; Feng et al., 2021, 2023b, a), as shown in Fig. 1c. These perturbations may poison some discriminative synthetic textures and mislead forgery detectors to incorrect predictions, thereby reducing their performance.

For the aforementioned two challenges, plenty of works have made impressive progress in boosting the generalization for cross-dataset detection, while few works focus on improving the robustness. **(1)** To improve the generalization across datasets, in earlier years, it is proposed to apply frequency artifacts (Durall et al., 2020; Liu et al., 2021; Qian et al., 2020) and spatial information (Wang et al., 2020; Afchar et al., 2018; Nguyen et al., 2019b). Recently, researchers have found data augmentations can lead to better generalization improvement, and many works focus on taking advantage of data augmentations to enhance the generalization further. Specifically, they usually synthesize various data by empirically designed augmentations, like generating blended images from two pristine images. Although these augmented-based methods can improve the detector's generalization, to a certain extent, they also have some defects. Firstly, their designed augmentations mainly depend on intuitive thoughts, which only include limited fixed synthetic strategies. Although SLADD (Chen et al., 2022) tries to construct various samples dynamically by adversarial learning, its manipulated region selection is still hand-crafted, and the synthetic strategy remains blend-based. Secondly, these augmented-based methods lack theoretical analysis for the effectiveness of data augmentations, and they do not take into account the use of frequency information to further improve the generalization like earlier works (Durall et al., 2020; Liu et al., 2021; Qian et al., 2020). **(2)** For robustness, among these attempts to improve the generalization, only limited works (Haliassos et al., 2022, 2021) consider keeping detectors robust against corruption, especially more threatening adversarial perturbations. Thus, an investigation on simultaneously improving the generalization and robustness of forgery detectors with theoretical analysis is noteworthy.

Inspired by the aforementioned discussion, in this paper, we propose a forgery detection generative adversarial network (FD-GAN) with two generators (i.e., a blend-based generator and a transfer-based generator) for adaptive data augmentations and a discriminator (i.e., the forgery detector), which can simultaneously boost both the generalization and robustness. **Specifically, the blend-based generator adaptively calculates manipulated regions (*i.e.*, the forgery masks) for blending. And the transfer-based generator mixes the synthetic style in the fake reference image and the semantics in the real source image to make augmentations.** Moreover, to further improve the generalization, we design the discriminator to judge whether the input is synthetic and predict manipulated regions (i.e., the forgery prototypes) with a collaboration of the spatial and frequency branches, like earlier works (Durall et al., 2020; Liu et al., 2021; Qian et al., 2020). Concretely, the spatial branch aims to compute spatial features and predict forgery prototypes, and the frequency branch works for mining helpful and generalizable frequency cues. Further, to boost the robustness of the detector, we propose a Low-rank Module in the frequency branch, which utilizes the Low-rank Estimation algorithm (Zhuo et al., 2021; Li et al., 2018b; Zhang et al., 2019) to filter out adversarial corruption. Therefore, our method can explore a large variety of augmented forgeries from the adaptive generator with spatial and robust frequency cues to improve generalization and robustness progressively. Besides, we also provide a theoretical analysis about the generalization and the robustness against adversarial attacks to guarantee the effectiveness of the proposed method. And

[1] Figure 1a and b display that different forgery techniques may produce forgeries with different textures, and existing forgery detection methods mainly focus on method-specific synthetic textures to improving performance on the specific forgery.

extensive experiments illustrate that our method can achieve state-of-the-art performance on forgery detection in both in-dataset and cross-dataset settings. In addition, the robustness of our method against adversarial attacks can also be demonstrated by experiments. Meanwhile, experimental results in Sect. 4.4 also verify the robustness of our method against common corruption (e.g., blur, noise, and *etc.*).

The contributions of our paper are as follows:

- We propose a forgery detection generative adversarial network (FD-GAN) with two generators (i.e., a blend-based generator and a transfer-based generator) for adaptive data augmentations and a discriminator (i.e., the forgery detector), which can not only improve the generalization across datasets but also boost the robustness against corruption and adversarial attacks.
- We design the discriminator to identify whether the input is real and indicate the manipulated regions with spatial and frequency branches. Besides, an LRM in the frequency branch based on Low-rank Estimation removes adversarial corruption to keep our model robust.
- Our method achieves superior performance on face forgery detection than current state-of-the-art methods in both in-dataset and cross-dataset settings. Moreover, our method shows strong robustness against common corruption and adversarial attacks.

2 Related Work

Face Forgery Techniques. In the past decades, face forgery techniques have rapidly developed. Early attempts (Dale et al., 2011; Garrido et al., 2014, 2015; Thies et al., 2015) on face forgery usually reconstruct 3D models for both source and target faces and generate synthetic videos. For example, Face2Face (Thies et al., 2016) is a classical real-time face forgery technique with 3D model reconstruction and image-based rendering. Some methods, like FaceSwap, even simply utilize only image processing to create synthetic faces. With the development of deep learning, many face forgery techniques [(e.g., Deep Video Portraits (Kim et al., 2018) and Neural Textures (Thies et al., 2019)] apply neural networks in their pipeline for facial reenactment. Recently, Generative models like Variational Auto-Encoders (VAEs) (Kingma & Welling, 2013), Generative Adversarial Networks (GANs) (Goodfellow et al., 2014a), and Diffusion Models (Ho et al., 2020) have revealed excellent performance in AI-content generation (AIGC) and become common choices for face forgery generation. Since these techniques can produce high-quality synthetic facial images and videos, their potential threats should be seriously considered.

Face Forgery Detection. Recent research has seen different attempts on face forgery detection (Wang et al., 2020; Rossler et al., 2019; Yang et al., 2019; Li et al., 2018a, 2020a; Chen et al., 2022; Shiohara & Yamasaki, 2022; Li et al., 2021b; Qian et al., 2020; Liu et al., 2021; Luo et al., 2021; Xu & Feng, 2023). Earlier works (Wang et al., 2020; Rossler et al., 2019) apply common CNNs like ResNet (He et al., 2016) and Xception (Chollet, 2017) to treat face forgery detection as a binary classification problem. Later, some approaches (Yang et al., 2019; Li et al., 2018a) pay attention to anomalies frequently in clumsy face forgery, while others (Cozzolino et al., 2021; Agarwa et al., 2019) utilize auxiliary identity information. Concerns about the generalization of forgery detectors arise along with the rapid development of deepfake techniques, and numerous methods have been proposed to solve this problem, such as applying data augmentations (Li et al., 2020a; Chen et al., 2022; Shiohara & Yamasaki, 2022), mining frequency cues (Li et al., 2021b; Qian et al., 2020; Liu et al., 2021; Luo et al., 2021; Miao et al., 2023, 2022), assisting with extra tasks (Chen et al., 2022; Nguyen et al., 2019a), using attention mechanisms (Zhao et al., 2021), and focusing on self-consistency (Huh et al., 2018; Dong et al., 2022). In this paper, we utilize a forgery detector with a spatial branch and a frequency branch to combine their advantages, rarely considered in previous works. Furthermore, in order to explore more training samples, an adversarial data augmentation strategy is also employed.

Adversarial Robustness. In the real world, neural networks can encounter incidental adversity like common corruption and intentional adversity created by adversarial attackers. Both can mislead models into wrong predictions. Some face forgery detection methods (Haliassos et al., 2021, 2022) have made progress in defending common corruption. However, adversarial attacks are usually considered more severe since they can target models with a crisis, fooling a model with invisible perturbations to human beings.

In addition to those universal adversarial attacks (Szegedy et al., 2014; Goodfellow et al., 2014b; Li et al., 2021a; Feng et al., 2024; Xu et al., 2024), some works (Li et al., 2021a; Jia et al., 2022) explore attacks targeted for face forgery detection. Li et al. (2021a) use pre-trained StyleGAN (Karras et al., 2019a) with gradients to generate high-quality adversarial examples, and Carlini and Farid (2020) apply black-box attacks on forgery detection to evaluate their robustness. Several frequency-based attacks (Jia et al., 2022; Luo et al., 2022) are proposed to evade frequency-based detectors and keep adversarial perturbations imperceptible. Recently, backdoor attacks (Sun et al., 2023), attribute variation-based attacks (Meng et al., 2023) and audio-based attacks (Panariello et al., 2023) have been applied to face forgery detection. On the contrary, only a few works (Hussain et al., 2021; Neekhar et al., 2021) try to prevent detectors from adversarial attacks. In this paper, we consider removing adversarial corruption with an elaborate module based on Low-rank Estimation.

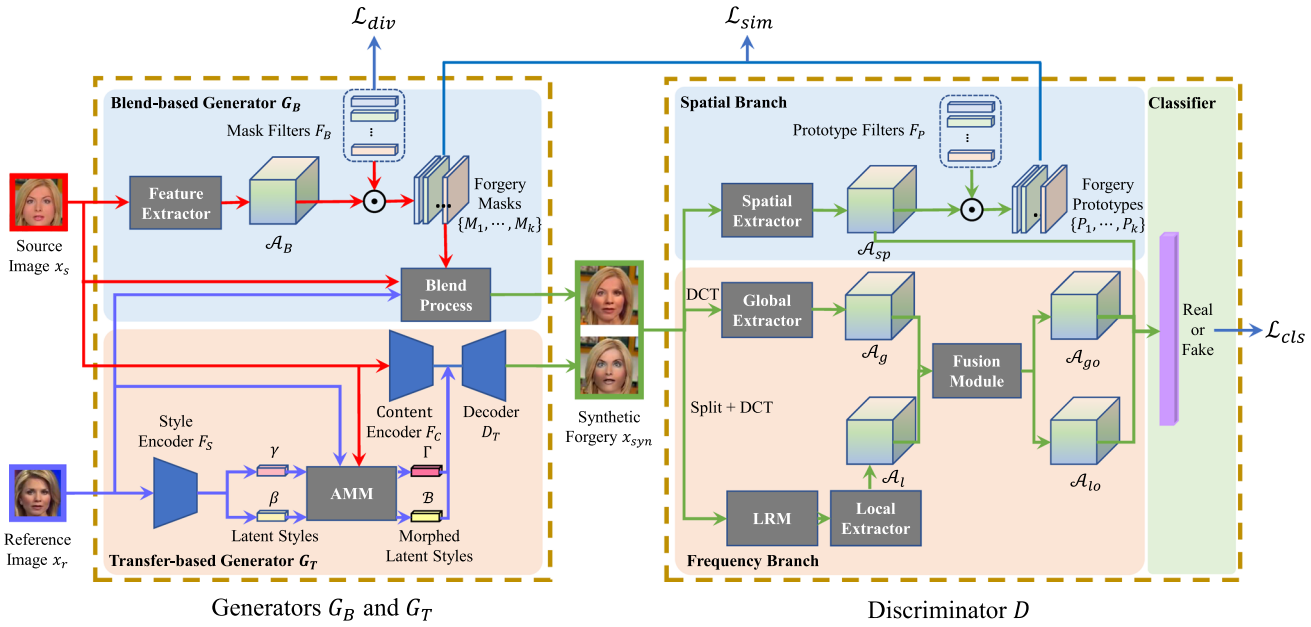

Fig. 2 The pipeline of our method. (1) The blend-based generator G_B and the transfer-based generator G_T take source and reference images to derive synthetic samples. (2) The discriminator D utilizes synthetic samples to get prediction (real or fake) and manipulated regions (i.e., the forgery prototypes $\{P_1, \cdots, P_k\}$) with a spatial branch and a frequency branch. Details are available in Sect. 3

3 Method

In this section, we propose a forgery detection generative adversarial network (FD-GAN), as shown in Fig. 2. Our FD-GAN consists of two generators and a discriminator: (1) The blend-based and transfer-based generators are responsible for adversarial data augmentations to boost generalization, illustrated in Sect. 3.1. (2) The discriminator (i.e., the detector) is applied for forgery detection with robust features, described in Sect. 3.2. Furthermore, Sect. 3.3 formulates the training objectives and Sect. 3.4 provides a detailed discussion with theoretical support.

3.1 Generators

Both the blend-based generator $G_B(\cdot;\theta_B)$ and transfer-based generator $G_T(\cdot;\theta_T)$ aim to improve generalization with adversarial data augmentations, where θ_B and θ_T denote their parameters. They take an original source image $x_s \in \mathcal{R}^{3\times H_0\times W_0}$ and a manipulated reference image $x_r \in \mathcal{R}^{3\times H_0\times W_0}$ as inputs, and output synthetic samples $x_{syn} \in \mathcal{R}^{3\times H_0\times W_0}$.

Blend-based Generator G_B. The blend-based generator extracts the feature map $\mathcal{A}_B \in \mathcal{R}^{C\times H\times W}$ from inputs, where C, H, and W represent the number of channels, the height and the width of the feature map. To produce high-quality data augmentations with dynamic manipulated regions, we suppose the manipulated regions can be divided into k local parts.

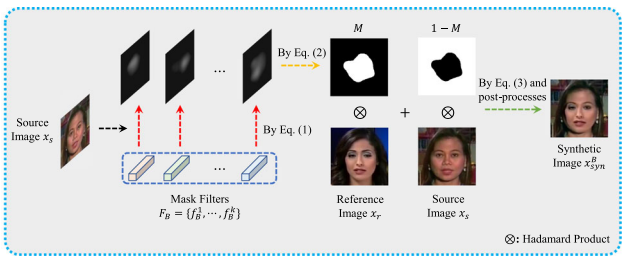

Fig. 3 Details of the Blend Process. The source image x_s is processed with k mask filters $F_B = \{f_B^1, \cdots, f_B^k\}$ by Eq. (1) to get the corresponding forgery masks $\{M_1, \cdots, M_k\}$. Since each M_i covers the corresponding part of manipulated regions, these forgery masks are fused by Eq. (2) to obtain the blending mask M. With the blending mask M, the blend-based generator merges the reference image x_r and the source image x_s into a synthetic image x^B_{syn}

Thus, we design k mask filters (MFs) $F_B = \{f_B^1, \cdots, f_B^k\}$, where each $f_B^i, \forall i = 1, \cdots, k$ is responsible for locating one specific part. Concretely, each MF $f_B^i \in \mathcal{R}^{1\times 1\times C}$ is parameterized by a 1×1 convolution kernel weight. These MFs are utilized to convolve with the feature map $\mathcal{A}_B$ and get the corresponding forgery masks $\{M_1 \cdots, M_k\}$:

$$M_i = \sigma(f_B^i \odot \mathcal{A}_B), \tag{1}$$

where $\odot$ is the convolution operation and $\sigma(\cdot)$ represents the sigmoid function. Each M_i covers the corresponding part of manipulated regions, and we fuse $\{M_1, \cdots, M_k\}$ as the

blending mask M:

$$M = Clip(\frac{1}{k}\sum_{i=1}^{k} Clip(M_i, th), 0), \tag{2}$$

where $Clip(X, th)$ treats values in X greater than th as 1 and others as 0. And the blending can be formulated as:

$$x_{syn}^{B} = M \otimes x_r + (1 - M) \otimes x_s, \tag{3}$$

where $\otimes$ is the Hadamard product, and x_{syn}^{B} is the synthetic result of blending. Details of the above steps are also displayed in Fig. 3. Note that pre-processes like face alignment, color transfer, and blur should be applied before blending to avoid significant artifacts in the results.

Transfer-based Generator G_T. The main idea of the transfer-based generator comes from a simple heuristic principle: Face forgeries can be viewed as a combination of original contents and synthetic styles. As a result, we can separate the styles of the synthetic reference samples x_r and generate a new synthetic sample x_{syn}^{T} with such synthetic styles and the content from the original source image x_s. Compared with the blend-based generator, the transfer-based generator can prevent boundary artifacts from synthetic samples, which enables it to explore more challenging samples for augmentation. It contains a style encoder F_S, a content encoder F_C, and a decoder D_T.

The style encoder F_S uses the encoder-bottleneck architecture (Choi et al., 2018) to extract the synthetic style from the reference image x_r and the output feature map $\mathcal{A}_S \in \mathcal{R}^{C^\star \times H^\star \times W^\star}$ is fed into two 1×1 convolution layers for latent styles (γ, β). The content feature map $\mathcal{A}_C \in \mathcal{R}^{C^\star \times H^\star \times W^\star}$ is produced from x_s by the content encoder F_C, similar to the style encoder. Then we consider these styles should be transferred between the similar relative parts on the face. For instance, synthetic textures on the eyes should be transferred to the corresponding eye regions of the source image x_s. Consequently, we introduce Attentive Makeup Morphing (AMM) module (Jiang et al., 2020b) to morph the latent styles (γ, β) for synthetic sample generation. AMM calculates an attentive matrix $A \in \mathcal{R}^{H^\star W^\star \times H^\star W^\star}$ by $\mathcal{A}_S$, $\mathcal{A}_C$, and facial landmarks, where $A_{i,j}$ suggests the attentive value between the i-th pixel in the source image x_s and the j-th pixel in the reference image x_r.[2] As a result, we get:

$$\gamma' = \sum_j A_{i,j}\gamma_j, \beta' = \sum_j A_{i,j}\beta_j. \tag{4}$$

γ' and β' are duplicated and expanded along the channel dimension to produce $(\Gamma, \mathcal{B})$. Finally, we generate synthetic samples x_{syn}^{T} with the morphed latent styles $(\Gamma, \mathcal{B})$, the content encoder F_C, and the decoder D_T (Choi et al., 2018), calculated by:

$$x_{syn}^{T} = D_T(\Gamma \otimes \mathcal{A}_C + \mathcal{B}). \tag{5}$$

[2] The detailed calculation of the attentive matrix A is available in Jiang et al. (2020b).

3.2 Discriminator

To discover universal synthetic artifacts, our discriminator $D(\cdot; \theta_D)$ has a frequency branch and a spatial branch, focusing on frequency and spatial features separately.

Spatial Branch. With reference to Rossler et al. (2019), we adopt Xception to capture the synthetic spatial textures. As shown in Fig. 2, the spatial extractor gets output feature maps $\mathcal{A}_{sp} \in \mathcal{R}^{C' \times H' \times W'}$ from the synthetic samples x_{syn}. Considering the blend-based generator creates forgeries with a series of forgery masks, we expect to encourage our spatial branch to locate the manipulated regions for better generalization.

Thus, k **forgery prototypes** $\{P_1, \cdots, P_k\}$ are proposed, corresponding to the forgery masks $\{M_1, \cdots, M_k\}$. They are created from k prototype filters $\{f_P^1, \cdots, f_P^k\}$, similar to those mask filters $\{f_B^1, \cdots, f_B^k\}$:

$$P_i = \sigma(f_P^i \odot \mathcal{A}_{sp}),\ f_P^i \in \mathcal{R}^{1 \times 1 \times C'},\ \forall i = 1, \cdots, k. \tag{6}$$

Frequency Branch. The frequency branch explores frequency artifacts from both a global and local perspective. It consists mainly of the following modules:

- **The Global Extractor** obtains the global feature map $\mathcal{A}_g$ from the synthetic sample x_{syn} transformed by Discrete Cosine Transform (DCT) to deal with global textures.
- **The Low-rank Module (LRM)** is applied to remove adversarial perturbations in blocks for robustness by Low-rank Estimation algorithms, and its explanation is shown in Sect. 3.4. Its input should be split into blocks and then transformed by DCT to reduce the computation, and its output should be reverted to the same size as the input.
- **The Local Extractor** pays more attention to local frequency textures. It derives the local feature map $\mathcal{A}_l$ from the output of LRM.
- **The Fusion Module** enables the collaboration between the global and local information with a convolution layer $Conv(\cdot)$. Specifically, it is formulated as:

$$\begin{aligned} M_c &= Conv(\mathcal{A}_g + \mathcal{A}_l), \\ \mathcal{A}_{lo} &= \mathcal{A}_l + M_c \otimes \mathcal{A}_g, \\ \mathcal{A}_{go} &= \mathcal{A}_g + M_c \otimes \mathcal{A}_l, \end{aligned} \tag{7}$$

where $\mathcal{A}_{go}$ and $\mathcal{A}_{lo}$ are the final global and local feature maps.

The frequency feature maps $\mathcal{A}_{go}$, $\mathcal{A}_{lo}$, and the spatial feature map $\mathcal{A}_{sp}$ are concatenated to get the final feature map, utilized in the prediction on forgery detection.

3.3 Training Objectives

Classification Loss $\mathcal{L}_{cls}$. Following previous works (Qian et al., 2020; Shiohara & Yamasaki, 2022), we use the binary cross-entropy loss to compute $\mathcal{L}_{cls}$.

$$\mathcal{L}_{cls} = -\frac{1}{N}\sum_{i=0}^{N-1}\{y_i \log f(x_i) + (1-y_i)\log(1-f(x_i))\}, \quad (8)$$

where y_i is the corresponding ground truth label, and $f(x_i)$ indicates the probability predicted by our proposed model of the input sample x_i.

Forgery Similarity Loss $\mathcal{L}_{sim}$. The forgery masks and forgery prototypes are fed into a fully connected (FC) layer to adaptively get the importance w_i of each forgery mask M_i and the corresponding forgery prototype P_i:

$$w_i = FC(concat(M_i, P_i)), \forall i = 1, \cdots, k. \quad (9)$$

Note the transfer-based synthetic sample x_{syn}^T can be viewed as the entire face synthesis, so we fix its w_i as $\frac{1}{k}$ and M_i as $\mathbf{1}$. And the forgery similarity loss $\mathcal{L}_{sim}$ can be denoted as:

$$\mathcal{L}_{sim} = \sum_{i=1}^{k} w_i ||M_i - P_i||_1. \quad (10)$$

Diversity Loss $\mathcal{L}_{div}$. It is likely that all forgery masks cluster in the same region and generate specific synthetic samples. We propose a diversity loss inspired by Liu et al. (2019), written as:

$$\mathcal{L}_{div} = \sum_{i=1}^{k}\sum_{j=1}^{k} cos(f_B^i, f_B^j), \quad (11)$$

where $cos(\cdot,\cdot)$ denotes the cosine similarity.

Finally, the optimization process can be formulated as:

$$\begin{aligned} &\min_{\theta_D}\max_{\theta_B,\theta_T} \mathcal{L}(\theta_B,\theta_T,\theta_D), \\ &s.t.\, \mathcal{L}(\theta_B,\theta_T,\theta_D) = \mathcal{L}_{cls} + \eta\mathcal{L}_{sim} + \lambda\mathcal{L}_{div}, \end{aligned} \quad (12)$$

where η and λ are hyper-parameters.

3.4 Discussion

In this part, we discuss our design on data augmentation and adversarial robustness in detail, suggesting our FD-GAN's effectiveness with theoretical analyses.

Data Augmentation. Several previous works (Shiohara & Yamasaki, 2022; Li et al., 2020a) utilize data augmentation to improve their generalization, while most depend on intuitive ideas. SLADD (Chen et al., 2022) shows the superior performance of generative models like GANs Goodfellow et al. (2014a) on augmentation in forgery detection. In this part, we will give helpful hints about data augmentation with detailed theoretical analysis in our Appendix A. They expound on how to make data augmentation effective and why generative models like GANs work well in forgery detection.

We first provide some basic settings for the following analysis. For convenience, we set real samples as the target since forgery detection is a binary classification problem. Suppose real samples in the training set are $X = \{X_1, \cdots, X_k\}$, where T_d is the size of X. They should be independent and identically distributed because they are all real samples, following a probability density function (PDF) p_d. Similarly, synthetic samples $Y = \{y_1, \cdots, y_{T_r}\}$ follow another PDF p_r and the size of Y is T_r. It is common sense that p_r depends on the data augmentation strategies and the source data. Here we mainly concern with the influence of various strategies rather than the source data since the latter is not the focus of our paper.

Our theoretical analysis suggests the following:

- Suggestion 1: Make the number of synthetic samples T_r as large as possible, which is universal among most various augmentation strategies in forgery detection.
- Suggestion 2: Generative models like GANs can effectively create diverse synthetic samples to complement the original training set X in forgery detection.

It is convenient for us to generalize our discovery across various forgery detector designs because our theoretical analysis mainly applies only to Logistic Regression. Based on Suggestion 2, we have developed our detector (FD-GAN) by a GAN-based approach, incorporating both a blend-based generator and a transfer-based generator to generate diverse synthetic samples for better generalization, demonstrated by the results in Sect. 4.3. Besides, our Appendix A also supports our Suggestion 1.

Adversarial Robustness. In general, adversarial examples can be regarded as a combination of natural semantic information and adversarial perturbations. A natural thought arises that we can prevent forgery detectors from adversarial attacks if the adversarial perturbations are filtered out. However, this process is usually lossy to the inputs' quality and may degrade the performance of detectors. To tackle this problem, Low-rank Estimation (Zhuo et al., 2021; Li et al., 2018b; Zhang et al., 2019) provides a well-established theory and useful algorithms for recovering data matrices from noise observations when original data matrices have some ideal properties (e.g., sparse singular values). As stated in Awasthi et al. (2020), natural images often hold sparse

singular values, and forgeries hold similar properties [3]. It implies the validity of using Low-rank Estimation to remove adversarial perturbations while keeping the natural semantic information. Since DCT usually makes matrices sparse with their rank unchanged, we design the LRM in the frequency branch. Concretely, it aims to solve the following problem:

$$\min_{\hat{R}} rank(X), \quad s.t., \forall i, j, \ \hat{R}_{ij} \approx X_{ij}, \tag{13}$$

where X, $\hat{R}$, and R represent the input with adversarial noises, the estimator, and the true data matrix, respectively. Moreover, the recovery of the true data matrix with the low-rank property can be theoretically guaranteed (Please refer to our Appendix A). It suggests our LRM can retain the semantic information and remove the annoying adversarial perturbations. Similar to other adversarial defense techniques based on pre-processing like (Dziugaite et al., 2016; Xu et al., 2017; Ding et al., 2019), LRM may slightly lower FD-GAN's performance on normal images but improve the adversarial robustness significantly, proved by our experiments in Sect. 4.5.

4 Experiment

4.1 Experimental Setup

Inputs. We use RetinaFace (Deng et al., 2020) for face extraction and DLIB (Sagonas et al., 2016) for facial landmark detection. All faces are aligned and resized in the training and testing datasets.

Discriminator. We adopt Xception (Rossler et al., 2019) as the backbone of Spatial Extractor, Global Extractor, and Local Extractor. Besides, LRM applies USVT (Chatterjee, 2012) for Low-rank Estimation.

textbiGenerator. We modify Xception (Chollet, 2017) as the backbone for the blend-based generator, which is initialized by pre-trained Xception on ImageNet (Deng et al., 2009). The transfer-based generator's architecture follows (Choi et al., 2018), as stated in our paper.

Optimization. The hyper-parameters in the final loss function are $\eta = 0.1$ and $\lambda = 0.02$. Besides, we use the Adam optimizer (Kingma & Ba, 2014) for both the generator and the discriminator with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batch size is fixed to 32, and the learning rates of the discriminator and generator are set to 1×10^{-4} and 3×10^{-5}, respectively.

Training Datasets. Based on recent deepfake detection methods (Li et al., 2021b; Wang & Deng, 2021; Luo et al., 2021; Chen et al., 2022; Liu et al., 2021; Qian et al., 2020; Li et al., 2020a), we train our model mainly on Faceforencis++ (FF++) dataset (Rossler et al., 2019), which consists of 1K real videos and 4K synthetic videos. Deepfakes (DF), FaceSwap (FS), Face2Face (F2F) (Thies et al., 2016), and Neural-Textures (NT) (Thies et al., 2019) are applied to generate synthetic videos with various compression levels, including RAW, High Quality (HQ), and Low Quality (LQ). We adopt the HQ version in our experiments by default unless otherwise specified.

Testing Datasets. To evaluate the generalizability of our method, we perform experiments on the following datasets: (1) CelebDF-v2 (CDF) (Li et al., 2020b) contains 518 test videos created by the improved deepfake technology. (2) Deepfake Detection Challenge Preview Dataset (DFDC) (Dolhansky et al., 2020) includes over 1K real videos and over 4K synthetic videos manipulated by multiple methods. (3) Deepfake Detection Dataset (DFD) contributes over 300 real videos and over 3K fake videos to support deepfake detection efforts. (4) DeeperForensics (DFo) (Jiang et al., 2020a) mainly consists of forged videos created by DF-VAE (Jiang et al., 2020a). As for the adversarial robustness evaluation, we follow (Jia et al., 2022) to choose 560 (140×4) frames from synthetic videos in FF++ test dataset.

Baselines. We mainly compare our methods with various augment-based methods (Face X-ray (Li et al., 2020a), SBI (Shiohara & Yamasaki, 2022), and SLADD (Chen et al., 2022)) and frequency-based forgery detection methods (F3Net (Qian et al., 2020), SPSL (Liu et al., 2021), and FDFL (Li et al., 2021b)). Some state-of-the-art methods are also selected for comparisons, such as Two Branch (Masi et al., 2020), MADD (Zhao et al., 2021), FTCN (Zheng et al., 2021), RealForensics (Haliassos et al., 2022), LipForensics (Haliassos et al., 2021), and ICT (Dong et al., 2022). Several popular baselines are considered, like Xception (Rossler et al., 2019), MesoNet (Afchar et al., 2018), Patch-based (Chai et al., 2020), CNN-GRU (Sabir et al., 2019), CNN-aug (Wang et al., 2020), Capsule (Nguyen et al., 2019b), Multi-task (Nguyen et al., 2019a), and DSP-FWA (Li & Lyu, 2018). Limited by computational cost and few official code implementations, some results are unavailable and represented by "-". In the adversarial robustness evaluation, we adopt spatial attacks (FGSM (Goodfellow et al., 2014b), PGD (Madry et al., 2017), MIM (Dong et al., 2018), DIM (Xie et al., 2019), and TIM (Dong et al., 2019)) and frequency attacks (FreqAttack (Jia et al., 2022) and SSAH (Luo et al., 2022)). The hyper-parameters follow the defaults in Jia et al. (2022).

Evaluation Metrics. Following the previous works (Li et al., 2021b; Wang & Deng, 2021; Luo et al., 2021; Chen et al., 2022; Liu et al., 2021; Qian et al., 2020; Li et al., 2020a), we mainly report the accuracy (ACC) and the Area Under the receiver operating characteristic Curve (AUC) for the evaluation on forgery detection. Besides, we choose the Attack Success Rate (ASR) for adversarial robustness evaluation based on Jia et al. (2022).

[3] The detailed proof is available in our Appendix A.

Table 1 In-dataset evaluation results. Quantitative results (ACC (%) and AUC (%)) on FF++ are displayed with Raw, HQ and LQ versions, respectively. The bold results are best

Methods	RAW		HQ		LQ	
	ACC	AUC	ACC	AUC	ACC	AUC
CNN-GRU Sabir et al. (2019)	98.60	**99.90**	97.00	99.30	90.10	92.20
LipForensics Haliassos et al. (2021)	98.90	**99.90**	**98.80**	99.70	**94.20**	98.10
LRNet Sun et al. (2021)	–	**99.90**	–	97.30	–	95.70
MADD Zhao et al. (2021)	–	–	96.37	98.97	86.95	87.26
MesoNet Afchar et al. (2018)	95.23	–	83.10	–	70.47	–
Patch-based Chai et al. (2020)	99.30	**99.90**	92.60	97.20	79.10	78.30
Two Branch Masi et al. (2020)	–	–	–	86.59	–	98.70
Xception Rossler et al. (2019)	99.26	99.20	95.73	96.30	86.86	89.30
F3Net Qian et al. (2020)	**99.95**	99.80	97.52	98.10	90.43	93.30
FDFL Li et al. (2021b)	99.43	99.70	96.69	99.30	89.00	92.40
SPSL Liu et al. (2021)	–	–	91.50	95.32	81.57	82.82
Face X-ray Li et al. (2020a)	–	99.10	–	87.35	–	61.60
Ours	99.83	**99.90**	98.12	**99.75**	91.60	**98.77**

Table 2 Cross-dataset evaluation results. Quantitative results (AUC (%)) on CDF, DFDC, DFD, and DFo are displayed

Method	Test Set AUC (%)			
	CDF	DFDC	DFD	DFo
Capsule Nguyen et al. (2019b)	63.7	–	69.7	68.4
CNN-aug Wang et al. (2020)	75.6	72.1	60.1	74.4
CNN-GRU Sabir et al. (2019)	69.8	68.9	–	74.1
DSP-FWA Li and Lyu (2018)	69.5	67.3	91.0	50.2
ICT Dong et al. (2022)	85.7	–	84.1	93.6
LipForensics Haliassos et al. (2021)	82.4	73.5	–	97.6
MesoInc4 Afchar et al. (2018)	53.6	–	59.1	51.4
Multi-task Nguyen et al. (2019a)	75.7	68.1	65.2	77.7
Patch-based Chai et al. (2020)	69.6	65.6	49.9	81.8
RealForensics Haliassos et al. (2022)	**86.9**	75.9	–	99.3
Xception Rossler et al. (2019)	73.7	70.9	95.6	84.5
SPSL Liu et al. (2021)	76.9	66.1	–	–
Face X-ray Li et al. (2020a)	79.5	65.5	94.1	86.8
SLADD Chen et al. (2022)	79.7	–	–	–
Ours	84.2	**77.2**	**97.1**	**99.6**

The bold results are best

Table 3 Robustness against adversarial attacks. ASR (%) of adversarial attacks on various face forgery detection methods is shown

Method	Attack						
	FGSM	PGD	DIM	TIM	FreqAttack	SSAH	Average
CNN-aug Wang et al. (2020)	65.9	66.1	72.0	74.9	67.2	79.8	72.0
Xception Rossler et al. (2019)	48.9	61.6	94.6	86.4	70.5	71.0	72.2
F3Net Qian et al. (2020)	**24.8**	80.9	86.9	82.6	82.5	76.8	72.4
SBI Shiohara and Yamasaki (2022)	92.9	92.6	95.0	96.0	90.2	94.0	93.4
Ours	40.7	**58.9**	**64.7**	**65.4**	**52.6**	**52.7**	**55.8**

The best results are in bold

Table 4 Robustness against common corruption. Average AUC scores (%) of methods for each corruption in Jiang et al. (2020a) across five intensity levels are shown

Method	Clean	Saturation	Contrast	Block	Noise	Blur	Pixel	Compress	Average
CNN-aug Wang et al. (2020)	99.8	99.3	99.1	95.2	54.7	76.5	91.2	72.5	84.1
CNN-GRU Sabir et al. (2019)	99.9	99.0	98.8	97.9	47.9	71.5	86.5	74.5	82.3
FTCN Zheng et al. (2021)	99.4	99.4	96.7	97.1	53.1	95.8	98.2	86.4	89.5
LipForensics Haliassos et al. (2021)	99.9	99.9	**99.6**	87.4	73.8	96.1	95.6	95.6	92.6
Patch-based Chai et al. (2020)	99.9	84.3	74.2	99.2	50.0	54.4	56.7	53.4	67.5
RealForensics Haliassos et al. (2022)	99.8	99.8	**99.6**	98.9	79.7	95.3	98.4	97.6	95.6
Xception Rossler et al. (2019)	99.8	99.3	98.6	**99.7**	53.8	60.2	74.2	62.1	78.3
F3Net Qian et al. (2020)	100.0	**100.0**	99.3	98.0	32.7	93.2	97.1	96.7	88.1
FDFL Li et al. (2021b)	99.8	99.8	99.4	91.5	67.3	95.1	89.2	96.2	91.2
SPSL Liu et al. (2021)	99.7	98.8	88.0	96.0	73.1	92.0	95.4	95.0	91.2
Face X-ray Li et al. (2020a)	99.8	97.6	88.5	99.1	49.8	63.8	88.6	55.2	77.5
SBI Shiohara and Yamasaki (2022)	99.6	98.9	97.6	98.7	42.2	73.4	89.2	85.9	83.7
Ours	99.8	99.5	**99.6**	99.0	**92.3**	**96.7**	**98.7**	**99.7**	**97.9**

The best results are in bold

4.2 In-Dataset Evaluation

In this part, we first compare our method with baselines on different face manipulation methods in FF++, including the RAW, HQ, and LQ versions. The results are shown in Table 1. Obviously, our method outperforms other baselines in AUC with the LQ dataset, and the performances on the RAW and HQ datasets are comparable (close to 100%). The performance improvement mainly benefits from the spatial and frequency information extracted by our method. In summary, these experiments show great success with previous methods on in-dataset evaluations, and we will illustrate the transferability of our method in the following.

4.3 Cross-Dataset Evaluation

In this section, we train our method on FF++ with multiple forgeries while evaluating it on other benchmarks, such as CDF, DFDC, DFo, and DFD. Since the synthetic samples are generated with unseen techniques in these benchmarks, this cross-dataset setting is more challenging than the in-dataset setting. Table 2 shows the AUC comparison with baseline methods for face forgery detection. Our method still achieves the state-of-the-art AUC in most cases, although it is on par with RealForensics (Haliassos et al., 2022) and ICT (Dong et al., 2022) on CDF, which obtain extra temporal information. These results illustrate the advantage of our proposed method on generalization under different datasets, which mainly benefits from the appreciable data augmentation and the spatial-frequency components. Detailed analysis of our method is available in Sect. 4.5 to understand the components responsible for excellent performance.

Table 5 Framework ablation. AUC (%) on CDF and DFD, and ASR (%) on selected adversarial images after training on FF++ are shown. "Frequency Branch" and "Spatial Branch" represent the corresponding part of our model. "w/o LRM" and "w/o Prototype" represent the performance without the LRM and forgery prototypes, respectively. All modules are defined in Sect. 3.2

Frequency Branch	Spatial Branch	w/o LRM	w/o Prototype	CDF	DFD	ASR
✓	✓			84.2	97.1	87.6
✓	✓	✓		**84.6**	**97.6**	55.8
✓	✓		✓	80.5	92.0	–
✓				84.1	93.1	–
	✓			78.0	94.2	–

The best results are in bold

4.4 Robustness Evaluation

In general, ideal detectors should be robust against common and adversarial corruption in addition to great generalization on various datasets. Thus, we evaluate our model with some baselines against adversarial attacks and common corruption to assess their robustness.

Robustness against Adversarial Attacks. Table 3 reports the ASR against FF++ and illustrates that our method outperforms other forgery detection methods by a large margin. For instance, PGD gets 66.1% ASR against CNN-aug (Wang et al., 2020) while only achieving 58.9% ASR against our method. Besides, CNN-aug (Wang et al., 2020) augmented with blur and compression resists more robust compared with other baselines, dropping a hint that some augmentations may help to keep models from adversarial corruption.

Table 6 Data augmentation ablation. AUC (%) on CDF and DFD after training on FF++ is shown. "Blend", "Blend (fixed)", and "Transfer" represent augmentations from the blend-based generator G_B, another fixed blending strategy Li et al. (2020a), and the transfer-based generator G_T, respectively

Blend	Blend (Fixed)	Transfer	CDF	DFD
✓		✓	**84.2**	**97.1**
	✓	✓	83.7	95.3
✓			82.7	94.3
	✓		81.8	94.1
		✓	79.9	93.7

The best results are in bold

Robustness against Common Corruption. Following (Haliassos et al., 2021) and (Haliassos et al., 2022), we train our FD-GAN on FF++. The distortions are the same as those in Jiang et al. (2020a), including changes in saturation and contrast, Gaussian blur and noise, compression on both video and image levels, and local block-wise distortion. Five different intensity levels are applied for each type, and the average AUC across all intensity levels is shown in Table 4. As expected, our method outperforms baselines in most cases, even compared to the popular RealForensics (Haliassos et al., 2022) and LipForensics (Haliassos et al., 2021). Besides, we report AUC for each intensity level separately on some common corruption with several models. In Fig. 5, we can observe that our FD-GAN performs best in almost all cases, especially against severe corruption. The superior performance of our FD-GAN against common corruption is credited to our generators (adaptive data augmentation) and the LRM (removing noises).

4.5 Ablation Studies

Ablation studies in this section try to determine how the factors contribute to our method's performance.

Framework Ablations. In Table 5, we ablate the components of our method and check the generalization of the detector on CDF and DFD (trained on FF++). First, training our detector with only the frequency or the spatial branch leads to a significant drop in performance (about 3.3% on average). Second, the LRM slightly degrades the performance of our detector (up to 0.5% drop in performance) while it results in much bettter robustness (leading to over 30% improvement in ASR). More results of LRM on adversarial robustness are available in the following. Finally, we observe remarkable improvements (about 4.4% on average) with the forgery prototypes.

Data Augmentations. Evaluation results are shown in Table 6. We obverse that with only one kind of augmentation, the performance of our model degrades, especially only with the transfer-based augmentation (by about 3.85% on average).

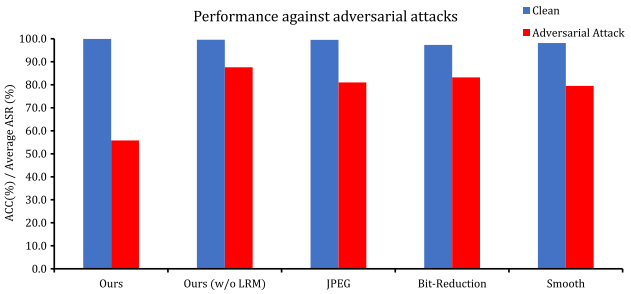

Fig. 4 Defense ablation. ACC (%) on clean samples and average ASR (%) on various adversarial attacks (the lower, the better) are shown. "w/o LRM" means our method's performance without the LRM. And "JPEG", "Bit-Reduction", and "Smooth" represent the performance of our model with the corresponding defense

Table 7 AUC (%) on CDF and DFD after training on FF++. "Blend" and "Transfer" denote the blend-based and transfer-based generators. "w AMM" and "w/o AMM" indicate the transfer-based generator is with or without AMM

Blend	Transfer w AMM	w/o AMM	CDF	DFD
✓	✓		**84.2**	**97.1**
✓		✓	83.0	94.5
	✓		79.9	93.7
		✓	78.3	92.6

The best results are in bold

The main reason is that limited choices of synthetic samples are likely to suppress the generalization of our model. Moreover, the fixed blend-based augmentation seems less effective than our blend-based generator since the fixed augmentations cannot be adaptive to the evolving forgery detector in training.

Defense Techniques. Here, we use other defense methods based on image processing techniques, including JPEG (Dziugaite et al., 2016), Bit-depth Reduction (Xu et al., 2017), and Smooth (Ding et al., 2019). As a baseline, we also provide the performance against perturbations of our detector without the LRM. In Fig. 4, each defense method can benefit our detector in robustness, but our LRM gains superior improvement compared with others. Besides, all adversarial defenses listed in Fig. 4 lead to a decrease in performance on clean samples and LRM shows the least performance penalty compared with other methods, which suggests its effectiveness.

Effect of Attentive Makeup Morphing. We perform experiments on Attentive Makeup Morphing (AMM), shown in Table 7. The performance drop without AMM suggests that AMM helps the transfer-based generator for better augmentation.

Sensitivity Analysis on Hyper-parameters of the Final Loss Function in Eq. (12). Sensitivity experiments on η and λ

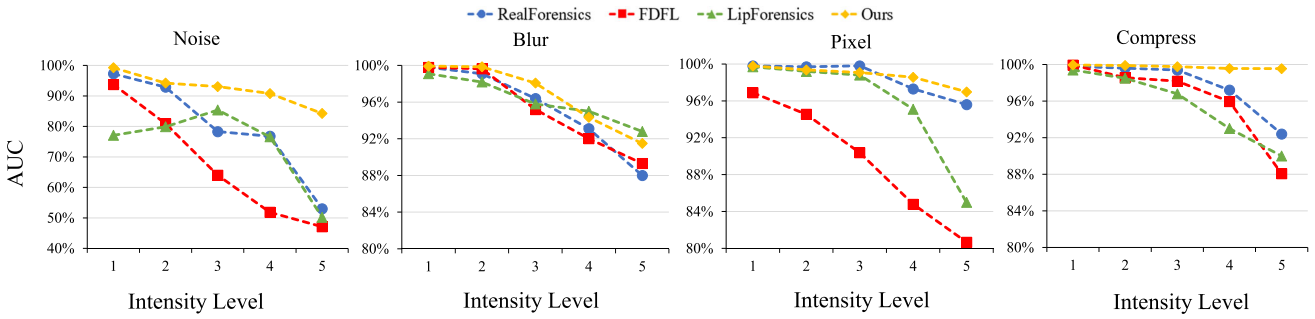

Fig. 5 AUC under common corruption in various intensity levels. The results of RealForensics and LipForensics are from their original paper

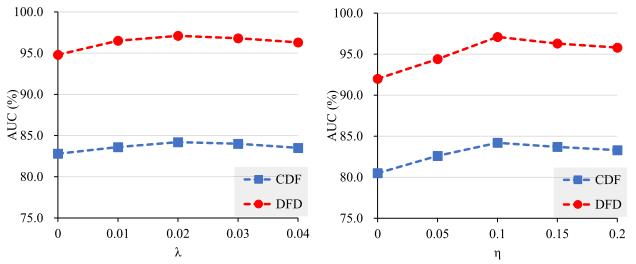

Fig. 6 Sensitivity analysis on hyper-parameters in loss function. We test detectors on CDF and DFD after training on FF++

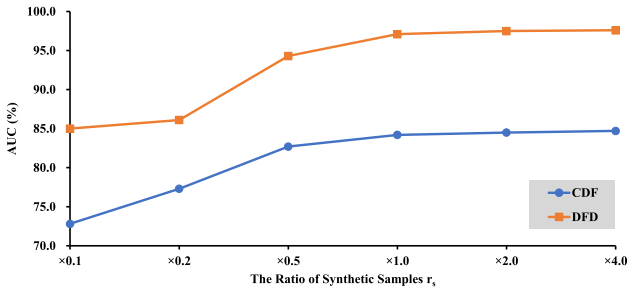

Fig. 7 Performance with various sizes of synthetic samples. AUC (%) of our FD-GAN on CDF (Li et al., 2020b) and DFD is shown when the number of synthetic samples varies. Note that the ratio of original samples to synthetic samples is fixed. "×0.5" means the applied training dataset contains only half of the synthetic samples compared with the normal training process ("×1.0"), and so on

are displayed in Fig. 6. The results verify the stability of our FD-GAN since both coefficients are robust in a large range.

Effect of the number of synthetic samples. To verify our **Suggestion 1**, we construct some experiments. In Fig. 7, we obverse that the performance of our method increases with a larger size of data augmentations. However, the performance shows a modest increase when the number of synthetic samples is very large. It suggests the limited benefit of much larger data augmentation.

Effect of the number of forgery prototypes. The forgery masks and prototypes are designed to locate the manipulated regions, leading to flexible data augmentation strategies and better generalization across datasets. In this part, we focus

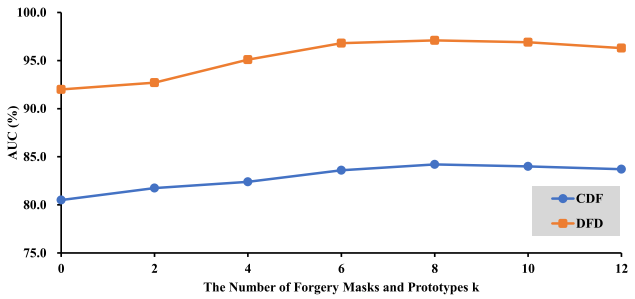

Fig. 8 Performance with various sizes of forgery masks and prototypes. AUC (%) of our FD-GAN on CDF (Li et al., 2020b) and DFD is shown when the number of forgery masks and prototypes k varies

on the effect of the number of forgery masks and prototypes (i.e., k). As displayed in Fig. 8, more masks and prototypes ($k \leq 8$) help to improve the generalization of our model, while too many masks and prototypes ($k \geq 8$) may result in confusion among various parts of manipulated regions. Therefore, we select $k = 8$ as our default setting.

Robustness against more threatening attacks. Table 8 displays the results of our FD-GAN and several baselines against more threatening attacks. Although the performance of our FD-GAN against these threatening attacks degrades by a large margin, it still outperforms other forgery detectors, suggesting its robustness.

"Fakeness" score of FD-GAN's synthetic samples. Following CNN-aug Wang et al. (2020), we list the performance of forgery detectors ("Blur + JPEG (0.5)" (Wang et al., 2020) and "Blur + JPEG (0.1)" Wang et al. (2020)) on synthetic samples generated by our FD-GAN and baselines. The results in Table 9 illustrate that our generator's synthetic samples can fool detectors better than others in all cases. It indicates our detector enjoys more challenging samples for better generalization.

4.6 Qualitative Results

Saliency map visualization. To gain better insights into both generalization and robustness, we highlight the behavior of our forgery prototypes with some baselines. Figure 9 visu-

Table 8 Attack success rate (ASR) of several adversarial attacks against forgery detectors

Method	Attack C&W (%)	BPDA (%)	EOT (%)
CNN-aug Wang et al. (2020)	86.6	76.7	81.3
F3Net Qian et al. (2020)	93.5	84.1	90.0
SBI Shiohara and Yamasaki (2022)	98.1	87.5	93.9
Ours	78.4	70.8	73.2

Table 9 The accuracy of CNN-aug Wang et al. (2020) on synthetic samples created by various generative models

	SAN Dai et al. (2019)	Deepfake Rossler et al. (2019)	BigGAN Brock et al. (2018)	CRN Chen and Koltun (2017)	StyleGAN2 Karras et al. (2019b)	StyleGAN Karras et al. (2019a)	Ours
Blur + JPEG (0.5) Wang et al. (2020)	50.0	51.1	59.0	87.6	68.4	73.4	**41.8**
Blur + JPEG (0.1) Wang et al. (2020)	50.5	53.5	70.2	86.3	84.4	87.1	**43.6**

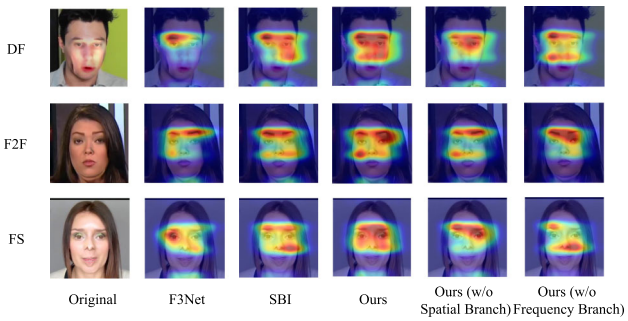

Fig. 9 Saliency map visualization. The baselines (F3Net (Qian et al., 2020) and SBI (Shiohara & Yamasaki, 2022)) capture method-specific artifacts while failing to detect the complete manipulated regions. Our FD-GAN's attention covers most of the manipulated regions. However, without the help of the frequency branch ("w/o Frequency Branch") or the spatial branch ("w/o Spatial Branch"), our method tends to locate method-specific artifacts like the baselines

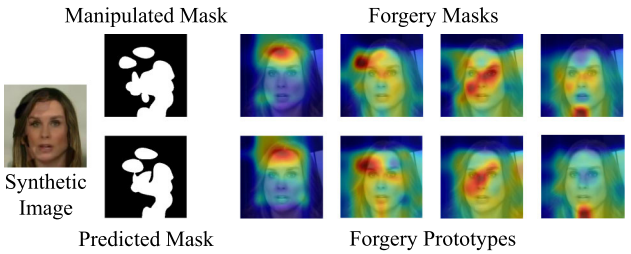

Fig. 10 Manipulated regions, predicted masks, part of forgery masks and the corresponding forgery prototypes

alizes several examples with Grad-CAM (Selvaraju et al., 2019). Clearly, our FD-GAN has more complete coverage of manipulated regions in most cases since our model enjoys a large variety of synthetic samples. However, the baselines focus on a limited choice of training samples.

Manipulated Regions and Predicted Masks. Visualization of manipulated regions and predicted masks are shown in Fig. 10. The predicted masks cover most manipulated regions. Moreover, the forgery prototypes describe the corresponding forgery masks accurately. It demonstrates that the discriminator can not only judge whether the input is synthetic, but also predict the manipulated regions.

Synthetic samples. Synthetic samples created by our FD-GAN are shown in Fig. 11. We can see that our synthetic samples are high-quality, although our paper does not focus on face forgery generation. And the manipulated regions vary with the corresponding source and reference images.

Forgery prototypes. In Figure 12, we can observe the explicit semantic correspondences between the same forgery prototypes. It proves the efficiency of our forgery prototypes. After training on plenty of synthetic samples, each forgery prototype can capture a specific semantic pattern so that the prediction is likely to be more robust to some corruption and capture universal synthetic artifacts.

5 Conclusion

In this paper, we propose a forgery detection generative adversarial network (FD-GAN) with two generators (a blend-based and a transfer-based generator) and a discriminator (i.e., detector), which can generalize well in unseen scenarios and keep robust against adversarial and common corruption. Specifically, the two generators can adaptively create challenging synthetic images with more flexible strategies to improve generalization. Besides, we design the discrim-

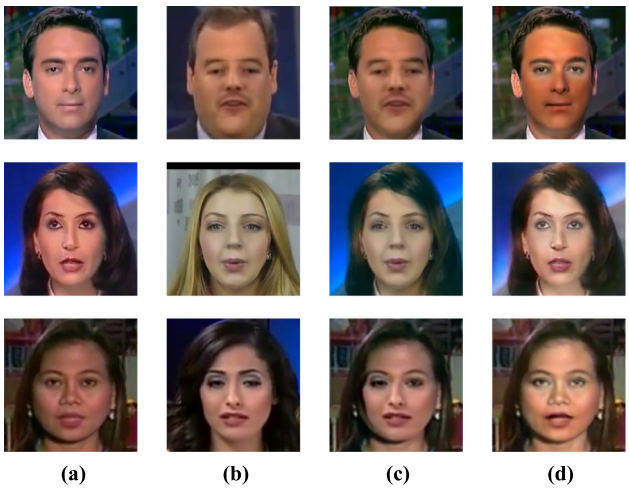

Fig. 11 Synthetic samples. **a**, **b**, **c**, and **d** represent the source images, reference images, blend-based synthetic samples, and transfer-based synthetic samples, respectively

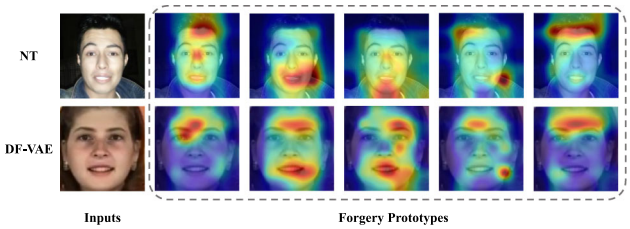

Fig. 12 Visualization of our forgery prototypes. We take five prototypes of forgeries created by NT (Thies et al., 2019) and DF-VAE (Jiang et al., 2020a) as examples. Obviously, we can see that each forgery prototype focuses on a specific manipulated part

inator to judge whether the input is synthetic and predicts the manipulated regions with a collaboration of spatial and frequency branches. Further, we propose a Low-rank Module in the frequency branch to remove adversarial corruption in the input for robustness improvement. And we also provide some guidelines on data augmentations for improving generalization and mathematical support for robustness. In experiments, FD-GAN exhibits superior generalization and robustness than the state-of-the-art methods.

Appendix A Mathematical Analysis

A.1 Analysis

Data augmentation. Recall the basic settings in our manuscript: Forgery detection can be viewed as a binary classification problem, and we set real samples as the target for convenience. A set of real samples $X = \{x_1, \cdots, x_{T_d}\}$ and another set of synthetic samples $Y = \{y_1, \cdots, y_{T_r}\}$ are also defined, where T_d and T_r represent the size of X and Y, respectively. Samples are independent. Meanwhile, x_i follows a probability density function (PDF) p_d while y_i follows another PDF p_r. Suggestions in our manuscript are listed as follows:

- Suggestion 1: Make the number of synthetic samples T_r as large as possible, which is universal among most various augmentation strategies in forgery detection.
- Suggestion 2: Generative models like GANs can effectively create diverse synthetic samples to complement the original training set X in forgery detection.

We consider the distribution of real samples in different datasets depends on a parameterized family of functions $p_u(\cdot;\theta)$ by θ. And there exist some $\theta^\star$ to satisfy $p_d = p_u(\cdot;\theta^\star)$. To solve this binary classification problem, we hope to approximate $\theta^\star$ as $\hat{\theta}$. Ideally, $\hat{\theta}$ should yield the following properties:

$$\int p_u(u;\hat{\theta})du = 1, \text{ (Normalized)} \tag{A1}$$

$$p_u(\cdot;\hat{\theta}) \geq 0. \text{ (Non-negative)}. \tag{A2}$$

However, the approximated $\hat{\theta}$ faces the gap between the train set and the test set, and Eq. (A1) may not hold across various datasets $p_u(\cdot;\theta)$. It is exactly the challenge of generalization in forgery detection.

To highlight this challenge, we denote distributions as $p'_u(\cdot;\alpha)$, and α represents the characteristics of datasets. For each $p'_u(\cdot;\alpha)$, we can use the normalization function $C(\alpha)$:

$$C(\alpha) = \int p'_u(u;\alpha)du. \tag{A3}$$

Obviously, $C(\alpha)$ can convert any distribution $p'_u(\cdot;\alpha)$ into a normalized one $p'_u(\cdot;\alpha)/C(\alpha)$. However, $C(\alpha)$ cannot be calculated directly in most cases. Previous works (Chen et al., 2020c; He et al., 2020; Oord Avd et al., 2018; Gutmann & Hyvärinen, 2012; Chen et al., 2020, a) provide a solution with the help of p_r since we can obtain p_d if p_r and the ratio p_d/p_r are known. In other words, we can infer the properties of real samples from the properties of synthetic samples and the differences between real and synthetic samples. Let α as a part of θ for simplification, i.e., $\theta \to (\theta, \alpha)$, and then we can get the following theorem (Gutmann & Hyvärinen, 2012):

Theorem 1 *By logistic regression, the objective function can be written as:*

$$J_T(\theta) = \frac{1}{T_d}\sum_{i=1}^{T_d}\ln(h(x_i;\theta)) + \gamma\frac{1}{T_r}\sum_{i=1}^{T_r}\ln(1-h(y_i;\theta)), \tag{A4}$$

with

$$h(u;\theta) = (1 + \tau \exp(-\ln \frac{p_d(u;\theta)}{p_r(u)})^{-1}, \tag{A5}$$

where $Y = \{y_1, \cdots, y_{T_r}\}$ *is the set of training synthetic samples,* T_r *is the size of* Y*, and* $\gamma = T_r/T_d$*.*

Equation (A4) is also known as the binary cross-entropy loss function, commonly used in forgery detection. Furthermore, we have several corollaries:

Corollary 1 *$\hat{\theta}_T$ is the value of θ, which maximizes Eq. (A4). It converges in probability to θ^* in proper conditions.*

Corollary 2 *$\sqrt{T_d}(\hat{\theta}_T - \theta^*) \to N(0, \Sigma)$ when $T_d + T_r \to \infty$. Σ is the covariance matrix.*

Corollary 1 illustrates that applying Eq. (A4) in forgery detection can lead to an ideal forgery detector, also proved by many empirical results (Zheng et al., 2021; Li et al., 2020a; Qian et al., 2020; Haliassos et al., 2021; Wang & Deng, 2021). Moreover, Corollary 2 indicates that MSE error $E\{\hat{\theta}^T - \theta^\star\} = tr(\Sigma)/T_d$ can be independent of augmentation strategies when the size of training samples becomes very large (our **Suggestion 1**).

One more observation is that $\mathcal{J}$ attains a maximum at $p_r = p_d$ so that we may achieve better performance when p_r is closer to p_d. Associated with the original GAN's (Goodfellow et al., 2014a) Proposition 2 (The distribution p_r produced by generator G can converge to p_d under suitable conditions), adversarial augmentation can work well in synthetic sample generation (our **Suggestion 2**).

Adversarial robustness. In this part, we continue the description in our manuscript and show the following results:

- **Sparse singular values of synthetic forgeries.** Synthetic samples are usually created by generative models (e.g., GANs and VAEs), where down-sampling and up-sampling are widely applied.
  However, up-sampling can lead to sparse singular values in synthetic forgeries, proved by Theorem 2:

Theorem 2 *Suppose the synthetic image as $X_s \in R^{H\times W}$ by up-sampling from $X_{s-ori} \in R^{\frac{H}{2}\times\frac{W}{2}}$, and we have:*

$$rank(X_s) \le \min(\frac{H}{2}, \frac{W}{2}). \tag{A6}$$

Considering multiple up-sampling operations applied in generative models, the rank of forgery images is limited to a very small number. Since low-rank matrices always have sparse singular values, we can see that both real and synthetic images hold sparsity in their singular values (i.e., the low-rank property). It implies the validity of using Low-rank Estimation algorithms for data recovery.

- **Low-rank Estimation can remove (adversarial) noises from the disturbed inputs.** First, we review the formulation of the Low-rank Estimation problem defined in our manuscript:

$$\min_{\hat{R}} rank(\hat{R}), \quad s.t., \forall i, j, \ \hat{R}_{ij} \approx R_{ij}, \tag{A7}$$

  where X, $\hat{R}$, and R represent the input with adversarial noises, the estimator, and the true data matrix, respectively. Besides, the next theorem describes why Low-rank Estimation can remove adversarial noises (Chatterjee, 2012):

Theorem 3 *Suppose the rank of $R \in \mathcal{R}^{n\times n}$ as r. C_0 and c are constants, depending on hyper-parameters in algorithms. $C(\epsilon)$ is a function of ϵ, which also relies on hyper-parameters. Then we have:*

$$MSE(\hat{R}) \le C_0 r_{min} + C(\epsilon)e^{-cnp}, \tag{A8}$$

where $r_{min} = \min(\sqrt{\frac{r}{mp}}, 1)$, and $MSE(\hat{R})$ is formulated as:

$$MSE(\hat{R}) = E\{\frac{1}{n^2}\sum_{n}^{i=1}\sum_{n}^{j=1}(R_{ij} - \hat{R}_{ij})^2\}. \tag{A9}$$

Detailed proof of Theorem 3 is available in USVT (Chatterjee, 2012). Theorem 3 tells us that $MSE(\hat{R})$ is strictly limited if r is small, leading to the recovery of majority entries and the erasure of adversarial noises. Besides, we can see that Theorem 3 is not limited to adversarial perturbations, suggesting its effectiveness in filtering out common noises like gaussian noises. Table 4 in our manuscript shows similar results.

A.2 Proofs

A.2.1 Proof for Theorem 1

Proof. We first come to the idea that no dataset bias exists. Let $U = \{u_1, \cdots, u_{T_d+T_r}\}$ be the union of $X = \{x_1, \cdots, x_{T_d}\}$ (the training set of real samples) and Y (the training set of synthetic samples). And each sample u_t is assigned a binary class label C_t: $C_t = 1$ if $u_t \in X$ and $C_t = 0$ if $u_t \in Y$. Obviously, the prior probabilities are:

$$P(C = 1) = \frac{T_d}{T_d + T_r}, \quad P(C = 0) = \frac{T_r}{T_d + T_r}. \tag{A10}$$

And the posterior probabilities are:

$$P(C = 1|u;\theta) = \frac{p_m}{p_m + \gamma p_r}, \ P(C=0|u;\theta) = \frac{\gamma p_r}{p_m + p_r}, \tag{A11}$$

where $p_m = p(u|C=1;\theta)$, $p_r = p(u|C=0)$ are the conditional probability densities with logistic regression. Using $h(u;\theta)$ introduced in our paper, we can easily get the conditional log-likelihood:

$$\begin{aligned} l(\theta) &= \sum_{t=1}^{T_d+T_r} C_t \ln P(C_t=1|u_t;\theta) \\ &\quad + (1-C_t)\ln P(C_t=0|u_t;\theta) \\ &= \sum_{t=1}^{T_d} \ln[h(x_t;\theta)] + \sum_{t=1}^{T_r} \ln[1-h(y_t;\theta)]. \end{aligned} \tag{A12}$$

Moreover, Eq. (A12) is also known as the **binary cross-entropy** function.

To treat the generalization problem, we can see:

$$\ln p_m(\cdot;\theta) = \ln p'(\cdot;\alpha) + c, \tag{A13}$$

where $\theta \leftarrow (\alpha, c)$. The parameter $c = C(\alpha)$ scales $p'(\cdot;\alpha)$ so that the normalized property can be fulfilled, as mentioned in our paper. Therefore, we look for $\theta \leftarrow (\alpha, c)$ instead of $\theta \leftarrow (\alpha, \theta)$ for convenience. With the above settings, the loss function can be formulated as:

$$\begin{aligned} J_T(\theta) &= \frac{1}{T_d}\left\{\sum_{t=1}^{T_d}\ln[h(x_t;\theta)] + \sum_{t=1}^{T_r}\ln[1-h(y_t;\theta)]\right\} \\ &= \frac{1}{T_d}\sum_{t=1}^{T_d}\ln[h(x_t;\theta)] + \frac{\gamma}{T_d}\sum_{t=1}^{T_r}\ln[1-h(y_t;\theta)]. \end{aligned} \tag{A14}$$

The weak law of large numbers shows that $J_T(\theta) \to J(\theta)$ in probability when $T_d + T_r \to \infty$:

$$J(\theta) = E\{\ln[h(x;\theta)]\} + \gamma E\{\ln[1-h(y;\theta)]\}.\square \tag{A15}$$

A.2.2 Proof for Corollary 1

Proof. First, we show the proper condition as followings:

Assumption 1 When $p_d \neq 0$, $p_r \neq 0$.

Assumption 2 $\sup_\theta |J_T(\theta) - J(\theta)| \to 0$ in probability.

Assumption 3 The matrix I_γ is positive definite, where

$$\begin{aligned} I_\gamma &= \int g(u)g(u)^T P_\gamma(u) p_d(u)\mathrm{d}u, \\ P_\gamma(u) &= \frac{\gamma p_r(u)}{p_d(u)+\gamma p_r(u)},\ g(u) = \nabla_\theta \ln p_m(u;\theta)|_{\theta^*}. \end{aligned} \tag{A16}$$

Proof.
Assumption 1 mainly contributes to the existence of $h(u;\theta)$. To prove Corollary 1, we have to show that given $\forall\epsilon > 0$, $P(||\hat{\theta}_T - \theta^*|| > \epsilon) \to 0$ when $T_d + T_r \to \infty$. With the definition of $J(\theta)$ in Eq. (A15), we have:

$$\begin{aligned} J(\theta+\epsilon\phi) = &\int \ln[h(u;\theta+\epsilon\phi)]p_d(u)du + \\ &\gamma\int \ln[1-h(u;\theta+\epsilon\phi)]p_r(u)du,\ \forall\epsilon>0, \end{aligned} \tag{A17}$$

where $\theta, \phi \in R^m$. For simplification, we define $r_\gamma(x) = \frac{1}{1+\gamma\exp(x)}$ and $G(u;\theta) = \ln p_m(u;\theta) - \ln p_r(u)$ so that $h(u;\theta) = r_\gamma(G(u;\theta))$ ($\gamma = \tau$ for normalization). we define auxiliary variables a_1 and a_2 as:

$$\begin{aligned} a_1 &= \phi^\top \nabla G(u;\theta), \\ a_2 &= \frac{1}{2}\phi^\top H_G(u;\theta)\phi, \end{aligned} \tag{A18}$$

where H_G is the Hessian matrix of $G(u;\theta)$. And we obtain

$$\ln r_\gamma(G(u;\theta+\epsilon\gamma)) = \ln r_\gamma\left(G(u;\theta) + \epsilon a_1 + \epsilon^2 a_2 + O\left(\epsilon^3\right)\right). \tag{A19}$$

Using Taylor expansions for $G(u;\theta)$, we get

$$\begin{aligned} &J(\theta+\epsilon\phi) = J(\theta) + A_1\epsilon + A_2\epsilon^2 + O(\epsilon^3). \\ &A_1 = \int a_1\left[p_d(u)(1-h(u;\theta)) - \gamma p_r(u)h(u;\theta)\right]\mathrm{d}u, \\ &A_2 = \int -\frac{1}{2}a_1^2(1-h(u;\theta))h(u;\theta)\left(p_d(u)+\gamma p_r(u)\right)\mathrm{d}u + \\ &\int a_2\left(p_d(u)(1-h(u;\theta)) - \gamma p_r(u)h(u;\theta)\right)\mathrm{d}u. \end{aligned} \tag{A20}$$

In Eq. (A20), the term of order ϵ should be 0 for any ϕ when $\theta = \theta^*$. It means:

$$p_d(u)(1-h(u;\theta)) = \gamma p_r(u)h(u;\theta). \tag{A21}$$

Thus, the objective function $J(\theta^* + \epsilon\phi)$ becomes:

$$\begin{aligned} J(\theta^*+\epsilon\phi) = &J(\theta^*) - \frac{\epsilon^2}{2}\int a_1^2(1-h(u;\theta^*))h(u;\theta) \\ &(p_d(u)+\gamma p_r(u))du + O(\epsilon^3). \end{aligned} \tag{A22}$$

With the help of the formulation:

$$\begin{aligned} h(u;\theta^*) &= \frac{p_d(u)}{p_d(u)+\gamma p_r(u)}, \\ 1-h(u;\theta^*) &= \frac{\gamma p_r(u)}{p_d(u)+\gamma p_r(u)}, \end{aligned} \tag{A23}$$

we get the expression of $J(\theta^* + \epsilon\phi)$:

$$J\left(\theta^* + \epsilon\phi\right) = J\left(\theta^*\right) - \frac{\epsilon^2}{2}\phi^T\left[\int g(u)g(u)^T P_\gamma(u)p_d(u)\mathrm{d}u\right]\phi + O\left(\epsilon^3\right). \tag{A24}$$

Obviously, $J(\theta^*)$ is a maximum when Assumption 3 holds. Based on the aforementioned results, we find that $J(\theta^*)$ is a global maximum, suggesting that there exists a $\delta(\epsilon)$ to satisfy $J(\theta) + \delta(\epsilon) \leq J(\theta^*)$. Thus, we have:

$$P(||\hat{\theta}_T - \theta^*|| > \epsilon) < P(J(\theta) + \delta(\epsilon) \leq J(\theta^*)). \tag{A25}$$

When $T_d + T_r \to \infty$, the difference between $J(\theta^*)$ and $J(\hat{\theta}_T)$ can be limited by:

$$\begin{aligned} |J(\theta^*) - J(\hat{\theta}_T)| &= |J(\theta^*) - J_T(\theta^*) + J_T(\theta^*) - J(\hat{\theta}_T)| \\ &\leq |J(\theta^*) - J_T(\theta^*) + J_T(\hat{\theta}_T) - J(\hat{\theta}_T)| \\ &\leq |J(\theta^*) - J_T(\theta^*)| + |J_T(\hat{\theta}_T) - J(\hat{\theta}_T)| \\ &\leq 2\sup_\theta |J(\theta) - J_T(\theta)|, \end{aligned} \tag{A26}$$

where $\hat{\theta}_T$ is the argument that maximizes $J_T(\cdot)$. With Assumption 2, we have the final result:

$$P(J(\hat{\theta}_T) + \delta(\epsilon) \leq J(\theta^*)) \leq \epsilon_1,\ \forall \epsilon_1 > 0, \tag{A27}$$

which indicates the conclusion in Corollary 1. □

A.2.3 Proof for Corollary 2 Proof for Corollary 2

Proof: By calculation, we derive the following results:

$$\begin{aligned} \nabla_\theta J_T(\theta^*) =& \frac{1}{T_d}\sum_{t=1}^{T_d}(1 - h(x_t;\theta^*))g(x_t) \\ & - \gamma\frac{1}{T_d}\sum_{t=1}^{T_r} h(y_t;\theta^*)g(y_t), \\ H_J(\theta^*) =& \frac{1}{T_d}\sum_{t=1}^{T_d}\{-(1 - h(x_t;\theta^*))h(x_t;\theta^*)g(x_t)g(x_t)^\top \\ & + (1 - h(x_t;\theta^*))H_G(x_t;\theta^*)\} \\ & - \gamma\frac{1}{T_r}\sum_{t=1}^{T_r}\{(1 - h(y_t;\theta^*))h(y_t;\theta^*)g(y_t)g(y_t)^\top \\ & + h(y_t;\theta^*)H_G(y_t;\theta^*)\}, \end{aligned} \tag{A28}$$

where $H_J(\theta^*)$ is the Hessian of $J_T(\theta)$ at θ^*, and $\nabla_\theta J_T(\theta^*)$ is the gradient of $J_T(\theta)$ at θ^*. And we have

$$\nabla_\theta J_T\left(\theta^*\right) + H_J\left(\theta^*\right)\left(\hat{\theta}_T - \theta^*\right) + O\left(||\hat{\theta}_T - \theta^*||^2\right) = 0. \tag{A29}$$

Up to terms of order $O(||\hat{\theta}_T - \theta^*||^2)$, we have:

$$-\sqrt{T_d}H_J^{-1}\left(\theta^*\right)\nabla_\theta J_T\left(\theta^*\right) = \sqrt{T_d}\left(\hat{\theta}_T - \theta^*\right). \tag{A30}$$

And we calculate the expectation of $\nabla_\theta J_T(\theta^*)$. The expectation $E\{\nabla_\theta J_T(\theta^*)\}$ can be formulated as:

$$\begin{aligned} E\{\nabla_\theta J_T(\theta^*)\} =& \int g(u)\left(1 - h\left(u;\theta^*\right)\right)p_d(u)\mathrm{d}u - \\ & \gamma\int g(u)h\left(u;\theta^*\right)p_r(u)\mathrm{d}u, \end{aligned} \tag{A31}$$

with the assist of the $i.i.d$ assumption. According to Eq. (A19), we get the following result:

$$E\{\nabla_\theta J_T(\theta^*)\} = 0. \tag{A32}$$

Let the variance of $\nabla_\theta J_T(\theta^*)$ be V, we see that:

$$\sqrt{T_d}\nabla_\theta J_T(\theta^*) \to N(0, V),\ (T_d + T_r \to \infty). \tag{A33}$$

Due to $H_J(\theta^*) \xrightarrow{P} -I_\gamma$ for large sample sizes $T_d + T_r \to \infty$, we set Σ as $I_\gamma^{-1}VI_\gamma^{-1}$ and get Corollary 2. □

A.2.4 Proof for Theorem 2

Proof. The up-sampling can be divided into two steps:

$$X_{s-ori} \in R^{\frac{H}{2}\times\frac{W}{2}} \to X_{mid} \in R^{H\times\frac{W}{2}} \to X_s \in R^{H\times W}. \tag{A34}$$

And X_{s-ori}, X_{mid} can be formulated as:

$$\begin{aligned} X_{s-ori} &= (\vec{x}_1, \cdots, \vec{x}_{\frac{W}{2}}), \\ X_{mid} &= (\vec{x}_1^m, \cdots, \vec{x}_{\frac{W}{2}}^m), \end{aligned} \tag{A35}$$

where $\vec{x}_i \in R^{\frac{H}{2}}$ and $\vec{x}_i^m \in R^{\frac{H}{2}}$ are the i-th row vector of X_{s-ori} and X_{mid}, respectively. Obviously, we get $\max(rank(X_{s-ori}), rank(X_{mid})) \leq \frac{W}{2}$ and X_s can be written as:

$$X_s = (\vec{x}_1^s, \cdots, \vec{x}_W^s), \tag{A36}$$

where $\vec{x}_1^s$ is the i-th row vector of X_s. Note that:

$$\begin{aligned} \vec{x}_{2k}^s &= (\vec{x}_k^m + \vec{x}_{k+1}^m)/2, \\ \vec{x}_{2k-1}^s &= \vec{x}_k^m. \end{aligned} \tag{A37}$$

Since the rows of X_s are linear combinations of the rows of X_{mid}, we get the following results:

$$rank(X_s) \leq rank(X_{mid}) \leq \frac{W}{2}. \tag{A38}$$

Without loss of generality, we can also get:

$$rank(X_s) \leq \frac{H}{2}. \tag{A39}$$

Therefore, we get Theorem 2 with Eqs. (A38) and (A39). □

Funding This work was supported by Technical Basic Scientific Research Project (Grant No. JSZL2023416A001-058).

Data Availibility Statement The datasets that support the findings of this study are available in the following repositories: Faceforencis++ (Rossler et al., 2019) at https://github.com/ondyari/FaceForensics, CelebDF-v2 (Li et al., 2020b) at https://github.com/yuezunli/celeb-deepfakeforensics, DFDC (Dolhansky et al., 2020) at https://ai.meta.com/datasets/dfdc/, DFD at https://github.com/ondyari/FaceForensics, DFo (Jiang et al., 2020a) at https://github.com/EndlessSora/DeeperForensics-1.0.

References

Afchar, D., Nozick, V., Yamagishi, J.,& et al. (2018). Mesonet: A compact facial video forgery detection network. In *2018 IEEE international workshop on information forensics and security (WIFS), IEEE*, pp. 1–7.

Agarwal, S., Farid, H., Gu, Y., & et al. (2019). Protecting world leaders against deep fakes. In *CVPR workshops*, p 38.

Awasthi, P., Jain, H., Rawat, A. S., et al. (2020). Adversarial robustness via robust low rank representations. *Advances in Neural Information Processing Systems, 33*, 11391–11403.

Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale gan training for high fidelity natural image synthesis. arXiv:1809.11096

Carlini, N., & Farid, H. (2020). Evading deepfake-image detectors with white-and black-box attacks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 658–659.

Chai, L., Bau, D., Lim, S. N., & et al. (2020). What makes fake images detectable? Understanding properties that generalize. In *European conference on computer vision*, Springer, pp. 103–120.

Chatterjee, S. (2012). Matrix estimation by universal singular value thresholding. *Annals of Statistics, 43*, 177–214.

Chen, L., Zhang, Y., Song, Y., & et al. (2022). Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18710–18719.

Chen, Q., & Koltun, V. (2017). Photographic image synthesis with cascaded refinement networks. In *2017 IEEE International Conference on Computer Vision (ICCV)* pp. 1520–1529.

Chen, T., Kornblith, S., Norouzi, M., & et al. (2020a). A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning, PMLR*, pp. 1597–1607.

Chen, T., Kornblith, S., Swersky, K., et al. (2020). Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems, 33*, 22243–22255.

Chen, X., Fan, H., Girshick, R., & et al. (2020c). Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297

Choi, Y., Choi, M., Kim, M., & et al. (2018). Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8789–8797.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251–1258.

Cozzolino, D., Rössler, A., Thies, J., & et al. (2021). Id-reveal: Identity-aware deepfake video detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15108–15117.

Dai, T., Cai, J., Zhang, Y., & et al. (2019). Second-order attention network for single image super-resolution. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11057–11066.

Dale, K., Sunkavalli, K., Johnson, M. K., & et al. (2011). Video face replacement. In *Proceedings of the 2011 SIGGRAPH Asia conference*, pp. 1–10.

Deng, J., Dong, W., Socher, R., & et al. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee*, pp. 248–255.

Deng, J., Guo, J., Ververas, E., & et al. (2020). Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5203–5212.

Ding, G. W., Wang, L., & Jin, X. (2019). Advertorch v0. 1: An adversarial robustness toolbox based on pytorch. arXiv preprint arXiv:1902.07623

Dolhansky, B., Bitton, J., Pflaum, B., & et al. (2020). The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397

Dong, X., Bao, J., Chen, D., & et al. (2022). Protecting celebrities from deepfake with identity consistency transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9468–9478.

Dong, Y., Liao, F., Pang, T., & et al. (2018). Boosting adversarial attacks with momentum. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193.

Dong, Y., Pang, T., Su, H., & et al. (2019). Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4312–4321.

Durall, R., Keuper, M., & Keuper, J. (2020). Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7890–7899.

Dziugaite, G. K., Ghahramani, Z., & Roy, D. M. (2016). A study of the effect of jpg compression on adversarial images. arXiv preprint arXiv:1608.00853

Feng, W., Wu, B., Zhang, T., & et al. (2021). Meta-attack: Class-agnostic and model-agnostic physical adversarial attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7787–7796.

Feng, W., Xu, N., Zhang, T., et al. (2023). Robust and generalized physical adversarial attacks via meta-gan. *IEEE Transactions on Information Forensics and Security, 19*, 1112–1125.

Feng, W., Xu, N., Zhang, T., & et al. (2023b). Dynamic generative targeted attacks with pattern injection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16404–16414.

Feng, W., Xu, N., Zhang, T., et al. (2024). Enhancing cross-task transferability of adversarial examples via spatial and channel attention. *IEEE Transactions on Multimedia*. https://doi.org/10.1109/TMM.2024.3349925

Garrido, P., Valgaerts, L., Rehmsen, O., & et al. (2014). Automatic face reenactment. In*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4217–4224.

Garrido, P., Valgaerts, L., Sarmadi, H., et al. (2015). Vdub: Modifying face video of actors for plausible visual alignment to a dubbed audio track. *Computer graphics forum* (pp. 193–204). New Jersey: Wiley Online Library.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., & et al. (2014a). Generative adversarial networks. arXiv preprint arXiv:1406.2661

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014b). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572

Gutmann, M. U., & Hyvärinen, A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research, 13*(2), 307–361.

Haliassos, A., Vougioukas, K., Petridis, S., & et al. (2021). Lips don't lie: A generalisable and robust approach to face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5039–5049.

Haliassos, A., Mira, R., Petridis, S., & et al. (2022). Leveraging real talking faces via self-supervision for robust forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14950–14962.

He, K., Zhang, X., Ren, S., & et al. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, K., Fan, H., Wu, Y., & et al. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738.

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems, 33*, 6840–6851.

Huh, M., Liu, A., Owens, A., & et al. (2018). Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp, 101–117.

Hussain, S., Neekhara, P., Jere, M., & et al. (2021). Adversarial deepfakes: Evaluating vulnerability of deepfake detectors to adversarial examples. In *Proceedings of the IEEE/CVF winter Conference on Applications of Computer Vision*, pp. 3348–3357.

Jia, S., Ma, C., Yao, T., & et al. (2022). Exploring frequency adversarial attacks for face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4103–4112.

Jiang, L., Li, R., Wu, W., & et al. (2020a). Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2889–2898.

Jiang, W., Liu, S., Gao, C., & et al. (2020b). Psgan: Pose and expression robust spatial-aware gan for customizable makeup transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5194–5202.

Karras, T., Laine, S., & Aila, T. (2019a). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410.

Karras, T., Laine, S., Aittala, M., & et al. (2019b). Analyzing and improving the image quality of stylegan. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (CVPR), pp. 8107–8116.

Kim, H., Garrido, P., Tewari, A., et al. (2018). Deep video portraits. *ACM Transactions on Graphics (TOG), 37*(4), 1–14.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114

Li, D., Wang, W., Fan, H., & et al. (2021a). Exploring adversarial fake images on face manifold. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5789–5798.

Li, J., Xie, H., Li, J., & et al. (2021b). Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6458–6467.

Li, L., Bao, J., Zhang, T., & et al. (2020a). Face x-ray for more general face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5001–5010.

Li, Y., & Lyu, S. (2018). Exposing deepfake videos by detecting face warping artifacts. arXiv preprint arXiv:1811.00656

Li, Y., Chang, M. C., & Lyu, S. (2018a). In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS), IEEE*, pp. 1–7.

Li, Y., Ma, T., & Zhang, H. (2018b). Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *Conference on Learning Theory, PMLR*, pp. 2–47.

Li, Y., Yang, X., Sun, P., & et al. (2020b). Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3207–3216.

Liu, D., Jiang, T., & Wang, Y. (2019). Completeness modeling and context separation for weakly supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1298–1307.

Liu, H., Li, X., Zhou, W., & et al. (2021). Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 772–781.

Luo, C., Lin, Q., Xie, W., & et al. (2022). Frequency-driven imperceptible adversarial attack on semantic similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15315–15324.

Luo, Y., Zhang, Y., Yan, J., & et al. (2021). Generalizing face forgery detection with high-frequency features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16317–16326.

Madry, A., Makelov, A., Schmidt, L., & et al. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083

Masi, I., Killekar, A., Mascarenhas, R. M., & et al. (2020). Two-branch recurrent network for isolating deepfakes in videos. In *European Conference on Computer Vision*, Springer, pp. 667–684.

Meng, X., Wang, L., Guo, S., & et al. (2023). Ava: Inconspicuous attribute variation-based adversarial attack bypassing deepfake detection. arXiv preprint arXiv:2312.08675

Miao, C., Tan, Z., Chu, Q., et al. (2022). Hierarchical frequency-assisted interactive networks for face manipulation detection. *IEEE Transactions on Information Forensics and Security, 17*, 3008–3021.

Miao, C., Tan, Z., Chu, Q., et al. (2023). F 2 trans: High-frequency fine-grained transformer for face forgery detection. *IEEE Transactions on Information Forensics and Security, 18*, 1039–1051.

Neekhara, P., Dolhansky, B., Bitton, J., & et al. (2021). Adversarial threats to deepfake detection: A practical perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 923–932.

Nguyen, H. H., Fang, F., Yamagishi, J., et al. (2019). Multi-task learning for detecting and segmenting manipulated facial images and videos. In *2019 IEEE 10th International Conference on Biometrics Theory. Applications and Systems (BTAS)* IEEE, pp. 1–8.

Nguyen, H. H., Yamagishi, J., & Echizen, I. (2019). Capsule-forensics: Using capsule networks to detect forged images and videos. In *ICASSP 2019–2019 IEEE International Conference on Acoustics. Speech and Signal Processing (ICASSP): IEEE*, pp. 2307–2311.

Oord Avd, Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748

Panariello, M., Ge, W., Tak, H., & et al. (2023). Malafide: a novel adversarial convolutive noise attack against deepfake and spoofing detection systems. In *INTERSPEECH 2023, 24th Conference of the International Speech Communication Association*.

Qian, Y., Yin, G., Sheng, L., & et al. (2020). Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European Conference on Computer Vision*, Springer, pp. 86–103.

Rossler, A., Cozzolino, D., Verdoliva, L., & et al. (2019). Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1–11.

Sabir, E., Cheng, J., Jaiswal, A., et al. (2019). Recurrent convolutional strategies for face manipulation detection in videos. *Interfaces (GUI), 3*(1), 80–87.

Sagonas, C., Antonakos, E., Tzimiropoulos, G., et al. (2016). 300 faces in-the-wild challenge: Database and results. *Image and Vision Computing, 47*, 3–18.

Selvaraju, R. R., Cogswell, M., Das, A., et al. (2019). Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision, 128*(2), 336–359.

Shiohara, K., & Yamasaki, T. (2022). Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18720–18729.

Sun, H., Li, Z., Liu, L., & et al. (2023). Real is not true: Backdoor attacks against deepfake detection. In *2023 9th International Conference on Big Data and Information Analytics (BigDIA), IEEE*, pp. 130–137.

Sun, Z., Han, Y., Hua, Z., & et al. (2021). Improving the efficiency and robustness of deepfakes detection through precise geometric features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3609–3618.

Szegedy, C., Zaremba, W., Sutskever, I., & et al. (2014). Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR* 2014.

Thies, J., Zollhöfer, M., Nießner, M., et al. (2015). Real-time expression transfer for facial reenactment. *ACM Trans Graph, 34*(6), 183–1.

Thies, J., Zollhofer, M., Stamminger, M., & et al. (2016). Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2387–2395.

Thies, J., Zollhöfer, M., & Nießner, M. (2019). Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG), 38*(4), 1–12.

Wang, C., & Deng, W. (2021). Representative forgery mining for fake face detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14923–14932.

Wang, S. Y., Wang, O., Zhang, R., & et al. (2020). Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8695–8704.

Xie, C., Zhang, Z., Zhou, Y., & et al. (2019). Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2730–2739.

Xu, N., & Feng, W. (2023). Metafake: Few-shot face forgery detection with meta learning. In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security*, pp. 151–156.

Xu, N., Feng, W., Zhang, T., et al. (2024). A unified optimization framework for feature-based transferable attacks. *IEEE Transactions on Information Forensics and Security, 19*, 4794–4808.

Xu, W., Evans, D., & Qi, Y. (2017). Feature squeezing: Detecting adversarial examples in deep neural networks. arXiv preprint arXiv:1704.01155

Yang, X., Li, Y., & Lyu, S. (2019). Exposing deep fakes using inconsistent head poses. In *ICASSP 2019–2019 IEEE International Conference on Acoustics. Speech and Signal Processing (ICASSP): IEEE*, pp. 8261–8265.

Zhang, R. Y., Sojoudi, S., & Lavaei, J. (2019). Sharp restricted isometry bounds for the inexistence of spurious local minima in nonconvex matrix recovery. *Journal of Machine Learning Research, 20*(114), 1–34.

Zhao, H., Zhou, W., Chen, D., & et al. (2021) Multi-attentional deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2185–2194.

Zheng, Y., Bao, J., Chen, D., & et al. (2021). Exploring temporal coherence for more general video face forgery detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15044–15054.

Zhuo, J., Kwon, J., Ho, N., & et al. (2021) On the computational and statistical complexity of over-parameterized matrix sensing. arXiv preprint arXiv:2102.02756