Rethinking the Up-Sampling Operations in CNN-based Generative Network for Generalizable Deepfake Detection

Chuangchuang Tan[1,2], Huan Liu[1,2], Yao Zhao[1,2]*, Shikui Wei[1,2], Guanghua Gu[3,4], Ping Liu[5], Yunchao Wei[1,2]

[1]Institute of Information Science, Beijing Jiaotong University
[2]Beijing Key Laboratory of Advanced Information Science and Network Technology
[3]School of Information Science and Engineering, Yanshan University
[4]Hebei Key Laboratory of Information Transmission and Signal Processing
[5]CSE department, University of Nevada, Reno, USA

tanchuangchuang@bjtu.edu.cn, yzhao@bjtu.edu.cn

Abstract

Recently, the proliferation of highly realistic synthetic images, facilitated through a variety of GANs and Diffusions, has significantly heightened the susceptibility to misuse. While the primary focus of deepfake detection has traditionally centered on the design of detection algorithms, an investigative inquiry into the generator architectures has remained conspicuously absent in recent years. This paper contributes to this lacuna by rethinking the architectures of CNN-based generator, thereby establishing a generalized representation of synthetic artifacts. Our findings illuminate that the up-sampling operator can, beyond frequency-based artifacts, produce generalized forgery artifacts. In particular, the local interdependence among image pixels caused by upsampling operators is significantly demonstrated in synthetic images generated by GAN or diffusion. Building upon this observation, we introduce the concept of Neighboring Pixel Relationships(NPR) as a means to capture and characterize the generalized structural artifacts stemming from up-sampling operations. A comprehensive analysis is conducted on an open-world dataset, comprising samples generated by ***28 distinct generative models****. This analysis culminates in the establishment of a novel state-of-the-art performance, showcasing a remarkable* ***12.8%*** *improvement over existing methods. The code is available at* https://github.com/chuangchuangtan/NPR-DeepfakeDetection.

1. Introduction

With the rapid evolution of image synthetic technologies, such as GAN[14, 25, 26], Diffusion[20, 53], AI-generated images have reached a level of realism that makes them virtually indistinguishable from authentic images to human observers. Nevertheless, the misuse of these capabilities poses potential threats in political and economic domains. Addressing this issue requires the development of generalizable deepfake detection methods. In recent years, notable strides [9, 12, 13, 34–36] have been made in forgery detection, particularly in the face forgery detection.

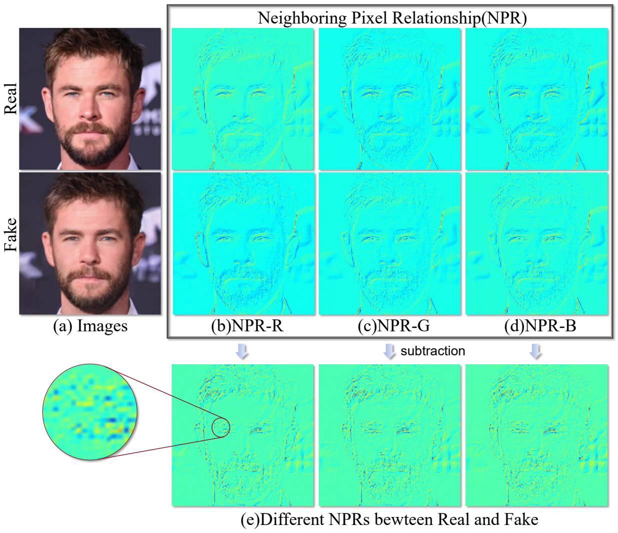

Neighboring Pixel Relationship(NPR)
Real
Fake
(a) Images
(b)NPR-R
(c)NPR-G
(d)NPR-B
subtraction
(e)Different NPRs bewteen Real and Fake

Figure 1. **The visualization of Neighboring Pixel Relationship (NPR) of real image and its inversion [72].** To fully understand the NPR, (a) we invert the real image by [72], and (b-d) present NPR heatmap for the R, G, B channel of images. In addition, to show that NPR can be used as artifacts representation, (e) the differential NPRs between real and fake images is shown. The proposed NPR Effectively reveals the differences between real and fake images.

In the realm of deepfake detection, a significant chal-

[*] Corresponding author

lenge for detectors is to generalize effectively to unseen deepfake sources in real-world scenarios. Recent advancements aimed at enhancing this generalization ability include the refinement of detection algorithms [21, 51], the augmentation of datasets [6, 22, 23], and the development of pre-trained models [49, 60]. Despite these efforts, a conspicuous gap remains in the lack of source-invariant representation exploited from the generator pipeline for forgery image detection. This deficiency leads to failures in detecting unknown forgery domains. Intriguingly, there has been a scarcity of investigative inquiry into generator architectures in recent years.

In addressing this challenge, our work centers on analyzing generator architectures to extract generalized artifact representations. Previous studies [12, 13, 69] have demonstrated the ubiquity of up-sampling components in common GAN pipelines. Simultaneously, given the widespread adoption of U-Net in diffusion models, such as DDPM [20], ADM [11], and LDM [53], the up-sampling layer emerges as a crucial module in diffusion models. The up-sampling cue holds significant potential for advancing generalizable deepfake detection. Building on these insights, in this paper, our focus is on achieving source-invariant forgery detection by rethinking artifacts stemming from the up-sampling component of common generation models. Existing works predominantly consider its impact on the entire image in the frequency domain. In contrast, our approach involves exploring the trace of the up-sampling layer at the level of local image pixels, providing a more nuanced understanding of its influence.

Specifically, in the pipelines of common generation models, up-sampling is employed to transform the low-resolution latent space into high resolution. Within the scaled feature, local pixels exhibit a strong relationship. For instance, employing nearest neighbor interpolation results in the local 2×2 pixels sharing the same value. Subsequent to the up-sampling operation, the scaled features are further processed through convolutional layers to generate images. During this process, a relationship is established among local pixels through the combination of the up-sampling operation and the translation invariance of CNN layers. This, in turn, manifests as discernible relationships among local pixels in the generated images.

Building upon these insights, we propose a simple but effective artifact representation, termed Neighboring Pixel Relationships (NPR), aimed at achieving generalized deepfake detection. NPR serves as the artifact representation for training the detection model. The primary innovation of our approach lies in introducing a simple yet versatile artifact representation derived from the common up-sampling component of generation pipelines. In Fig. 1, we showcase NPR heatmaps for a real face and its inversion. Significantly, NPR effectively captures artifacts related to image details such as hair, eyes, and beard. Despite the generator's tendency to enhance details for realism, traces of the up-sampling layer persist in the local image pixels.

To comprehensively evaluate the generalization ability of our proposed NPR, we conduct simulations using a vast database of images generated by 28 distinct models [1]. Our extensive experiments demonstrate the effectiveness and versatility of the artifact representation generated by the NPR across diverse and unseen sources.

Our paper makes the following contributions:

- We propose a simple yet effective artifact representation, Neighboring Pixel Relationships (NPR), designed to capture local up-sampling artifacts from image pixels. Thanks to the widespread use of up-sampling operations in existing generation models, NPR demonstrates the ability to generalize to unseen sources.
- We demonstrate that up-sampling operators can cause generalized forgery artifacts beyond frequency-based artifacts. The trace of the up-sampling layer from local image pixels exhibits more generalization compared to its influence on the whole image in the frequency domain for deepfake detection.
- Our experiments validate the effectiveness of the proposed NPR, showcasing strong generalization capabilities across 28 different generation models used for forgery image synthesis.

2. Related Work

In this section, we present a concise survey of deepfake detection approaches, categorizing them into two main groups: image-based and frequency-based detection.

2.1. Image-based Fake Detection

Some studies [54] utilize images as input data to train binary classification models for forgery detection. Rossler *et al.* [54] employ images to train a straightforward Xception [9] for detecting fake face images. Other works concentrate on specific regions, such as eyes and lips, to discern fake face media [16, 32]. Yu *et al.* [67] and Marra *et al.* [43] extract the unique fingerprints of the GAN model from generated images to perform detection. Chai *et al.* [5] employ limited receptive fields to identify patches that render images detectable. Some works enhance the generalization of detectors to unseen sources by diversifying training data through augmentation methods [61, 62], adversarial training [7], reconstruction techniques [4, 18], fingerprint generators [22], and blending images [57]. Additionally, Ju *et al.* [24] integrate global spatial information and local informative fea-

[1] ProGAN, StyleGAN, StyleGAN2, BigGAN, CycleGAN, StarGAN, GauGAN, Deepfake, AttGAN, BEGAN, CramerGAN, InfoMaxGAN, MMDGAN, RelGAN, S3GAN, SNGAN, STGAN, DDPM, IDDPM, ADM, LDM, PNDM, VQDiffusion, Glide, Stable Diffusion v1, Stable Diffusion v2, DALLE, and Midjourney.

tures to train a two-branch model. The AltFreezing [63] adopts both spatial and temporal artifacts to achieve Face Forgery Detection. Li *et al*. [30] utilize Continual learning [68, 71] to solve the continual deepfake detection problem. Ojha *et al*. [49] and Tan *et al*. [60] employ feature maps and gradients, respectively, as general representations. DIO [58] utilizes training-free filters to extract artifact representations.

2.2. Frequency-based Fake Detection

Given that GAN architectures heavily rely on up-scaling operations, some studies [12, 13] delve into the impact of up-sampling across the entire image, developing the frequency spectrum as a representation of up-sampling artifacts. LOG [44] integrates information from both color and frequency domains to detect manipulated face images and videos. F3-Net [51] introduces frequency components partition and the discrepancy of frequency statistics between real and forged images into face forgery detection. Luo *et al*. [41] utilize multiple high-frequency features of images to enhance generalization performance. ADD [65] develops two distillation modules for detecting highly compressed deepfakes, including frequency attention distillation and multi-view attention distillation. BiHPF [21] amplifies the magnitudes of artifacts through two high-pass filters. FreGAN [23] observes that unique frequency-level artifacts in generated images can lead to overfitting to training sources. Consequently, FreGAN mitigates the impact of frequency-level artifacts through frequency-level perturbation maps. FreqNet [59] aims to enhance frequency space learning for deepfake detection.

3. Methodology

Our work is dedicated to designing a generalizable artifacts representation through an analysis of common up-sampling operations in popular generators. We introduce a form of local up-sampling artifacts, named Neighboring Pixel Relationships (NPR), the details are presented in this section.

3.1. Problem setup

The overarching objective of Generalizable Deepfake Detection is to develop a universal detector capable of accurately identifying deepfake images, even when faced with limitations in the availability of diverse training sources.

In the given context, we consider a real-world image scenario denoted as X, which is sampled from n different sources:

$$\begin{aligned} X &= \{X_1, X_2, \ldots, X_i, \ldots, X_n\}, \\ X_i &= \{x^i_j, y_j\}^{N_i}_{j=1}, \end{aligned} \tag{1}$$

where N_i represents the number of images originating from the ith source X_i, and x^i_j is the jth image of X_i. Each image is labeled with y, indicating whether it belongs to the category of "real" ($y = 0$) or "fake" ($y = 1$).

Here, we train a binary classifier $D(\cdot)$, utilizing the training source X_i:

$$\begin{aligned} P_i &= f(X_i), \\ D^i &= \arg\min_{\theta}\ loss(D(P_i; \theta),\ y), \end{aligned} \tag{2}$$

where $f()$ is the representation extractor, P_i is the artifact representation of X_i.

Our overarching goal is to design a well extractor $f()$, which extracts a generalized artifact P_i from the training source X_i. Subsequently, the generalizable detector D^i can be obtained by training on the artifact P_i originating from X_i, yet it demonstrates robust performance when faced with images from previously unseen sources denoted as X_t. The ability to generalize across unseen sources is a crucial objective of our detector representation extractor $f()$.

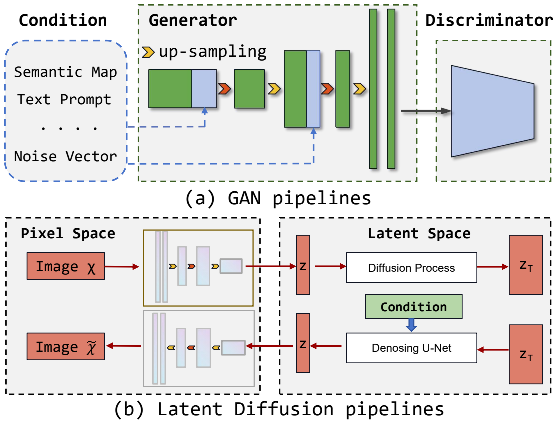
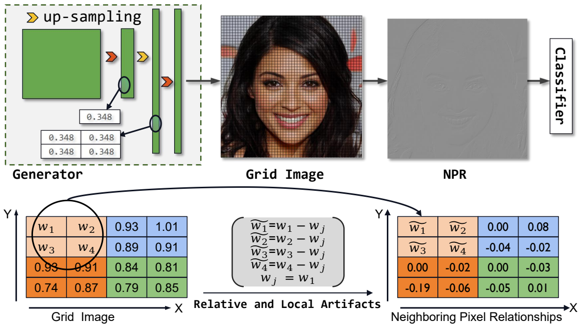


Figure 2. In the pipelines of common generation models, GAN and Diffusion, up-sampling is employed to transform the low-resolution latent space into high resolution.



w_1	w_2	0.93	1.01
w_3	w_4	0.89	0.91
0.93	0.91	0.84	0.81
0.74	0.87	0.79	0.85

$$\begin{aligned} \widetilde{w_1} &= w_1 - w_j \\ \widetilde{w_2} &= w_2 - w_j \\ \widetilde{w_3} &= w_3 - w_j \\ \widetilde{w_4} &= w_4 - w_j \\ w_j &= w_1 \end{aligned}$$

$\widetilde{w_1}$	$\widetilde{w_2}$	0.00	0.08
$\widetilde{w_3}$	$\widetilde{w_4}$	-0.04	-0.02
0.00	-0.02	0.00	-0.03
-0.19	-0.06	-0.05	0.01

Figure 3. **The overview of Neighboring Pixel Relationships.** We rethink artifacts stemming from the up-sampling component of common generation models. The proposed Neighboring Pixel Relationships focus on the local interdependence between image pixels caused by up-sampling operators. The NPR is employed to train detector as artifact representation.

3.2. Up-sampling operations in generator pipeline

Before we dive into the details of the method, let's briefly explore the up-sampling operations commonly used in generator pipelines, such as those in GANs and Diffusions.

GAN pipelines: We present an overview of the fundamental pipeline inherent to Generative Adversarial Networks (GANs), as depicted in Figure 3 (a). It comprises two primary constituents, namely the discriminator and the generator. In the context of a GAN, the generator function serves to establish a mapping that originates in a lower-dimensional latent space and extends to the image space. Within the architecture of the generator, two predominant components are typically incorporated, including convolutional layers and up-sampling layers. In these up-sampling layers, their primary function is to accept low-resolution features as input and subsequently generate high-resolution features as their output. It is noteworthy to emphasize that while the architectural configurations of GAN models exhibit substantial diversity, the adoption of an upsampling module maintains consistency.

Diffusion pipelines: Additionally, the Diffusion pipeline is illustrated in Figure 3 (b). Recently, diffusion models include two structures: diffusion with U-Net and latent diffusion models. In diffusion with U-Net, the U-Net model is employed to estimate the noise component from a noisy image. During inference time, diffusion models sample noise and gradually reduce the noise level until obtaining a clean image. The latent diffusion models include an encoder, denoising U-Net, and decoder. It uses a U-Net to perform diffusion in a latent domain and then decodes the latent signal with a decoder to generate an image. Although the processes of generation in the diffusion model and GAN are different, the decoder of the diffusion model also widely adopts up-sampling layers to generate images.

3.3. Neighboring pixel relationships

Building upon the above analysis of generation pipelines, we observe that up-sampling operations are commonly employed in current image generation techniques, including GANs and Diffusions. While existing research has delved into studying global up-sampling artifacts in the frequency domain [12, 13], Jeong *et al.*[23] have discovered that frequency-based artifacts are insufficient for achieving generalization detection, given the diverse patterns in the frequency domain of GANs. In this context, we reconsider the up-sampling layer in popular generation models and introduce the concept of local up-sampling artifacts in the spatial domain.

We focus on the portion of the generator near the output images, consisting of an up-sampling layer *up* with l scale, convolutional layers *conv* with activate functions, input feature maps $x \in \mathbb{R}^{W\times H\times C}$, and the output images $I \in \mathbb{R}^{(l\times W)\times(l\times H)\times 3}$.

$$\begin{aligned} \hat{x} &= up(x), \\ I &= conv(\hat{x}), \end{aligned} \tag{3}$$

where $\hat{x} \in \mathbb{R}^{(l\times W)\times(l\times H)\times C}$ is the up-scaled feature map. We then divide the image I and $\hat{x}$ into $W \times H$ grids. Each gird is the $l \times l$ patches. Let V_I and $V_{\hat{x}}$ denote grids set of I and $\hat{x}$, respectively. The $v_I^c \in V_I$ and $v_{\hat{x}}^c \in V_{\hat{x}}$ indicate a gird of I and $\hat{x}$, respectively. Most of generators commonly employ an up-sampling layer with $l = 2$ scale.

The elements of $v_{\hat{x}}^c$ exhibit a strong correlation generated by the up-sampling layer. For instance, when adopting nearest neighbor interpolation as the up-sampling layer, the elements of $v_{\hat{x}}^c$ share same value. Here are some key characteristics: 1) The elements of $v_{\hat{x}}^c$ has strong correlation generated by upsampling layer, 2) The function *conv* is fixed during inference, 3) The function *conv* is translation invariance. Consequently, the correlation of elements is presented in v_I^c. We capture the correlation of local pixels in v_I^c as the up-sampling artifacts.

Specifically, the differences in each v_I^c are extracted as artifacts representation, as following:

$$\begin{aligned} &v_I^c = \{w_1, ..., w_i, ..., w_n\}, n = l\times l \\ &\hat{v}_I^c = \{w_1 - w_j, ..., w_i - w_j, ..., w_n - w_j\}, 1 \leq j \leq n, \end{aligned} \tag{4}$$

where w_i is the elements of v_I^c, $\hat{v}_I^c$ denotes the neighboring pixel relationships of v_I^c. We adopt subtraction to capture relative relationship of pixels in v_I^c. The w_j can be employed by any element in v_I^c. The NPR of the whole image is the set of all grids $\hat{v}_I^c$. Our NPR set l and j to 2 and 1, respectively. In the Section 4, We will discuss the effect of l and w_j, and explore the possibility of replacing w_j with max or mean of v_I^c.

We employ the proposed neighboring pixel relationships $\hat{v}_I^c$ as the artifacts representation to train the classifier for deepfake detection. The NPR captures the local relative correlation between pixels in local patches. This correlation, presented in the image domain, derives from the up-sampling layer and benefits from the translation invariance of the convolutional layer. The relative and local nature of the proposed up-sampling artifacts allows the neighboring pixel relationship to be generalized to unknown sources.

4. Experiments

4.1. Settings

Training Dataset:

To ensure a consistent basis for comparison, we employ the training set of ForenSynths [62] to train the detectors, following baselines [21, 23, 62]. The training set consists of 20 distinct categories, each comprising 18,000 synthetic images generated using ProGAN, alongside an equal number of real images sourced from the LSUN dataset. In line with previous research [21, 23], we adopt specific 4-class training settings, denoted as $(car, cat, chair, horse)$.

Testing Dataset:

To assess the generalization ability of the proposed method on the real-world scenarios, we adopt various real images and diverse GAN and Diffusions models. The evaluation dataset consists of **five datasets** containing **28 generation models**.

- **8 models from ForenSynths[62] :** The test set includes fake images generated by 8 generation models [2]. Real images are sampled from 6 datasets (LSUN[66], ImageNet[55], CelebA[39], CelebA-HQ[25], COCO[33], and FaceForensics++[54]).
- **9 GANs from GANGen[10]:** To replicate the unpredictability of wild scenes, we extend our evaluation by collecting images generated by 9 additional GANs [3]. There are 4K test images for each model, with equal numbers of real and fake images.
- **8 Diffusions from DIRE [64]:** To expand the testing scope, we adopt the diffusions dataset of DIRE [64] for evaluation, including ADM [11], DDPM [20], IDDPM [47], LDM [53], PNDM [37], Vqdiffusion [15], Stable Diffusion v1 [53], Stable Diffusion v2 [53]. The real images are sampled from LSUN [66] and ImageNet[55] datasets.
- **4 Diffusions from Ojha [49]:** This test set contains images generated from ADM [11], Glide [46], DALL-E-mini [52], LDM [53]. It adopts images of LAION[56] and ImageNet[55] datasets as the real data.
- **5 Diffusions from Diffusion1kStep:** Moreover, we sample test images generated from diffusion models using 1000 diffusion steps, namely DDPM[20], IDDPM[47], ADM[11], collect images of Midjourney[4], and DALLE[52][5] from social platform Discord.

More detailed information on the test set is given in the supplementary material.

Implementation Details: We design a lightweights CNN network using convolutional layer and Resnet[17] block as the classifiers for NPR with 1.44 million parameters. The detector is trained using the Adam optimizer[28] with a learning rate of 2×10^{-4}, a batch size of 32. Our method is implemented using the PyTorch on Nvidia GeForce RTX 3090 GPU. To assess the performance of the proposed method, we follow the evaluation metrics used in the baselines[21, 23, 49], which include the average precision score (A.P.) and accuracy (Acc.).

Baselines: We perform comparisons the proposed NPR with existing deepfake detection works, including CNNDetection(CVPR2020) [62], Frank(PRML 2020) [13], Durall(CVPR 2020) [12], Patchfor(ECCV 2020) [5], F3Net(ECCV 2020) [51], SelfBland(CVPR 2022)[57], GANDetection(ICIP 2022) [42], BiHPF(WACV 2022) [21], FrePGAN(AAAI 2022)[23], LGrad(CVPR 2023) [60], Ojha(CVPR 2023) [49]. We re-implement baselines [5, 12, 13, 49, 51, 62] with the official codes using 4-classes training setting, and adopt the official pretrained models of baselines[42, 57, 60].

4.2. Generalization capability evaluation

In this section, we demonstrate that the local artifacts representation, Neighboring Pixel Relationships, induced by the up-sampling operations in common generation pipelines, can be easily employed for identifying generated image data. Even a detector trained on a GAN model exhibits the ability to generalize to recently generated diffusion images.

To analyze if the proposed local up-sampling artifacts is a common occurrence for different generation models, we perform the evaluation on a cross-sources dataset comprising images from 28 distinct generation models. The details of test set are given in the Section 4.1 and the supplementary material. The detectors of NPR are trained by the images from ProGAN and subsequently evaluated on 16 GANs, 1 Deepfake, and 11 Diffusion models. We adopt specific 4-classes training settings for all experiments in this paper, denoted as $ProGAN$-$(car, cat, chair, horse)$.

4.2.1 GAN-Sources Evaluation

In order to valid the generalization ability on images of GAN sources, two test sets, ForenSynths[62] and self-synthesis GAN datasets, are employed for evaluation. These datasets encompass 17 distinct generation models used to test the detection performance of the NPR detector trained on ProGAN images. The results are presented in Table 1 and Table 2.

Table 1 provides a comprehensive overview of the performance of detectors on the test set of ForenSynths[62]. The Neighboring Pixel Relationships (NPR) outperforms its counterparts, showcasing higher mean accuracy (Acc.) and comparable mean average precision (A.P.) metrics. Particularly noteworthy are the mean accuracy values of NPR, which reach 92.5%. It is worth emphasizing the remarkable superiority of NPR over the current state-of-the-art methods, LGrad and Ojha. In terms of mean accuracy, NPR surpasses LGrad and Ojha by 6.4% and 3.4%, respectively, underscoring its efficacy in generalizable deepfake detection.

To further assess the generalization ability of Neighboring Pixel Relationships (NPR) across GAN-sources, we expanded the evaluation to include results from 9 addi-

[2] ProGAN[25], StyleGAN[26], StyleGAN2[27], BigGAN[3], CycleGAN [73], StarGAN [8], GauGAN[50] and Deepfake [54]

[3] AttGAN[19], BEGAN[2], CramerGAN[1], InfoMaxGAN[29], MMDGAN[31], RelGAN[48], S3GAN[40], SNGAN[45], and STGAN[38]

[4] discord.com/channels/662267976984297473

[5] discord.com/channels/974519864045756446

Method	ProGAN		StyleGAN		StyleGAN2		BigGAN		CycleGAN		StarGAN		GauGAN		Deepfake		Mean	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
CNNDetection[62]	91.4	99.4	63.8	91.4	76.4	97.5	52.9	73.3	72.7	88.6	63.8	90.8	63.9	92.2	51.7	62.3	67.1	86.9
Frank[13]	90.3	85.2	74.5	72.0	73.1	71.4	88.7	86.0	75.5	71.2	99.5	99.5	69.2	77.4	60.7	49.1	78.9	76.5
Durall[12]	81.1	74.4	54.4	52.6	66.8	62.0	60.1	56.3	69.0	64.0	98.1	98.1	61.9	57.4	50.2	50.0	67.7	64.4
Patchfor[5]	97.8	100.0	82.6	93.1	83.6	98.5	64.7	69.5	74.5	87.2	100.0	100.0	57.2	55.4	85.0	93.2	80.7	87.1
F3Net[51]	99.4	100.0	92.6	99.7	88.0	99.8	65.3	69.9	76.4	84.3	100.0	100.0	58.1	56.7	63.5	78.8	80.4	86.2
SelfBland[57]	58.8	65.2	50.1	47.7	48.6	47.4	51.1	51.9	59.2	65.3	74.5	89.2	59.2	65.5	93.8	99.3	61.9	66.4
GANDetection[42]	82.7	95.1	74.4	92.9	69.9	87.9	76.3	89.9	85.2	95.5	68.8	99.7	61.4	75.8	60.0	83.9	72.3	90.1
BiHPF[21]	90.7	86.2	76.9	75.1	76.2	74.7	84.9	81.7	81.9	78.9	94.4	94.4	69.5	78.1	54.4	54.6	78.6	77.9
FrePGAN[23]	99.0	99.9	80.7	89.6	84.1	98.6	69.2	71.1	71.1	74.4	99.9	100.0	60.3	71.7	70.9	91.9	79.4	87.2
LGrad [60]	99.9	100.0	94.8	99.9	96.0	99.9	82.9	90.7	85.3	94.0	99.6	100.0	72.4	79.3	58.0	67.9	86.1	91.5
Ojha [49]	99.7	100.0	89.0	98.7	83.9	98.4	90.5	99.1	87.9	99.8	91.4	100.0	89.9	100.0	80.2	90.2	89.1	98.3
NPR(our)	99.8	100.0	96.3	99.8	97.3	100.0	87.5	94.5	95.0	99.5	99.7	100.0	86.6	88.8	77.4	86.2	92.5	96.1

Table 1. **Cross-GAN-Sources Evaluation on the test set of ForenSynths[62].** The results of [12, 13, 21, 23, 62] are from [21, 23]. Red and Blue represent the best and second-best performance, respectively.

Method	AttGAN		BEGAN		CramerGAN		InfoMaxGAN		MMDGAN		RelGAN		S3GAN		SNGAN		STGAN		Mean	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
CNNDetection[62]	51.1	83.7	50.2	44.9	81.5	97.5	71.1	94.7	72.9	94.4	53.3	82.1	55.2	66.1	62.7	90.4	63.0	92.7	62.3	82.9
Frank[13]	65.0	74.4	39.4	39.9	31.0	36.0	41.1	41.0	38.4	40.5	69.2	96.2	69.7	81.9	48.4	47.9	25.4	34.0	47.5	54.7
Durall[12]	39.9	38.2	48.2	30.9	60.9	67.2	50.1	51.7	59.5	65.5	80.0	88.2	87.3	97.0	54.8	58.9	62.1	72.5	60.3	63.3
Patchfor[5]	68.0	92.9	97.1	100.0	97.8	99.9	93.6	98.2	97.9	100.0	99.6	100.0	66.8	68.1	97.6	99.8	92.7	99.8	90.1	95.4
F3Net[51]	85.2	94.8	87.1	97.5	89.5	99.8	67.1	83.1	73.7	99.6	98.8	100.0	65.4	70.0	51.6	93.6	60.3	99.9	75.4	93.1
SelfBland[57]	63.1	66.1	56.4	59.0	75.1	82.4	79.0	82.5	68.6	74.0	73.6	77.8	53.2	53.9	61.6	65.0	61.2	66.7	65.8	69.7
GANDetection[42]	57.4	75.1	67.9	100.0	67.8	99.7	67.6	92.4	67.7	99.3	60.9	86.2	69.6	83.5	66.7	90.6	69.6	97.2	66.1	91.6
LGrad [60]	68.6	93.8	69.9	89.2	50.3	54.0	71.1	82.0	57.5	67.3	89.1	99.1	78.5	86.0	78.0	87.4	54.8	68.0	68.6	80.8
Ojha [49]	78.5	98.3	72.0	98.9	77.6	99.8	77.6	98.9	77.6	99.7	78.2	98.7	85.2	98.1	77.6	98.7	74.2	97.8	77.6	98.8
NPR(our)	83.0	96.2	99.0	99.8	98.7	99.0	51.8	70.7	94.5	98.3	98.6	99.0	99.6	100.0	79.0	80.0	88.8	97.4	98.0	100.0

Table 2. **Cross-GAN-Sources Evaluation on the Self-Synthesis 9 GANs dataset.**

Method	ADM		DDPM		IDDPM		LDM		PNDM		VQ-Diffusion		Stable Diffusion v1		Stable Diffusion v2		Mean	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
CNNDetection[62]	53.9	71.8	62.7	76.6	50.2	82.7	50.4	78.7	50.8	90.3	50.0	71.0	38.0	76.7	52.0	90.3	51.0	79.8
Frank[13]	58.9	65.9	37.0	27.6	51.4	65.0	51.7	48.5	44.0	38.2	51.7	66.7	32.8	52.3	40.8	37.5	46.0	50.2
Durall[12]	39.8	42.1	52.9	49.8	55.3	56.7	43.1	39.9	44.5	47.3	38.6	38.3	39.5	56.3	62.1	55.8	47.0	48.3
Patchfor[5]	77.5	93.9	62.3	97.1	50.0	91.6	99.5	100.0	50.2	99.9	100.0	100.0	90.7	99.8	94.8	100.0	78.1	97.8
F3Net[51]	80.9	96.9	84.7	99.4	74.7	98.9	100.0	100.0	72.8	99.5	100.0	100.0	73.4	97.2	99.8	100.0	85.8	99.0
SelfBland[57]	57.0	59.0	61.9	49.6	63.2	66.9	83.3	92.2	48.2	48.2	77.2	82.7	46.2	68.0	71.2	73.9	63.5	67.6
GANDetection[42]	51.1	53.1	62.3	46.4	50.2	63.0	51.6	48.1	50.6	79.0	51.1	51.2	39.8	65.6	50.1	36.9	50.8	55.4
LGrad [60]	86.4	97.5	99.9	100.0	66.1	92.8	99.7	100.0	69.5	98.5	96.2	100.0	90.4	99.4	97.1	100.0	88.2	98.5
Ojha [49]	78.4	92.1	72.9	78.8	75.0	92.8	82.2	97.1	75.3	92.5	83.5	97.7	56.4	90.4	71.5	92.4	74.4	91.7
NPR (our)	88.6	98.9	99.8	100.0	91.8	99.8	100.0	100.0	91.2	100.0	100.0	100.0	97.4	99.8	93.8	100.0	95.3	99.8

Table 3. **Cross-Diffusion-Sources Evaluation on the test of DiffusionForensics [64].**

Method	DALLE		Glide_100_10		Glide_100_27		Glide_50_27		ADM		LDM_100		LDM_200		LDM_200_cfg		Mean	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
CNNDetection[62]	51.8	61.3	53.3	72.9	53.0	71.3	54.2	76.0	54.9	66.6	51.9	63.7	52.0	64.5	51.6	63.1	52.8	67.4
Frank[13]	57.0	62.5	53.6	44.3	50.4	40.8	52.0	42.3	53.4	52.5	56.6	51.3	56.4	50.9	56.5	52.1	54.5	49.6
Durall[12]	55.9	58.0	54.9	52.3	48.9	46.9	51.7	49.9	40.6	42.3	62.0	62.6	61.7	61.7	58.4	58.5	54.3	54.0
Patchfor[5]	79.8	99.1	87.3	99.7	82.8	99.1	84.9	98.8	74.2	81.4	95.8	99.8	95.6	99.9	94.0	99.8	86.8	97.2
F3Net[51]	71.6	79.9	88.3	95.4	87.0	94.5	88.5	95.4	69.2	70.8	74.1	84.0	73.4	83.3	80.7	89.1	79.1	86.5
SelfBland[57]	52.4	51.6	58.8	63.2	59.4	64.1	64.2	68.3	58.3	63.4	53.0	54.0	52.6	51.9	51.9	52.6	56.3	58.7
GANDetection[42]	67.2	83.0	51.2	52.6	51.1	51.9	51.7	53.5	49.6	49.0	54.7	65.8	54.9	65.9	53.8	58.9	54.3	60.1
LGrad [60]	88.5	97.3	89.4	94.9	87.4	93.2	90.7	95.1	86.6	100.0	94.8	99.2	94.2	99.1	95.9	99.2	90.9	97.2
Ojha [49]	89.5	96.8	90.1	97.0	90.7	97.2	91.1	97.4	75.7	85.1	90.5	97.0	90.2	97.1	77.3	88.6	86.9	94.5
NPR (our)	94.5	99.5	98.2	99.8	97.8	99.7	98.2	99.8	75.8	81.0	99.3	99.9	99.1	99.9	99.0	99.8	95.2	97.4

Table 4. **Cross-Diffusion-Sources Evaluation on the diffusion test set of Ojha [49] .**

tional GAN models, as presented in Table 2. The results demonstrate the consistent outperformance of NPR in terms of generalization performance on GAN-sources. NPR achieves an impressive average accuracy of 98.0%, substantially surpassing the best-performing baselines, Ojha [49] and LGrad [60], which attain accuracy values of 77.6% and 75.4%, respectively.

The results obtained across 17 diverse generation models underscore the remarkable generalization capability of the proposed artifacts representation derived from up-sampling

Method	DDPM		IDDPM		ADM		Midjourney		DALLE		Mean	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
CNNDetection[62]	50.0	63.3	48.3	52.68	53.4	64.4	48.6	38.5	49.3	44.7	49.9	52.7
Frank[13]	47.6	43.1	70.5	85.7	67.3	72.2	39.7	40.8	68.7	65.2	58.8	61.4
Durall[12]	54.1	53.6	63.2	71.7	39.1	40.8	45.7	47.2	53.9	52.2	51.2	53.1
Patchfor[5]	54.1	66.3	35.8	34.2	68.6	73.7	66.3	68.8	60.8	65.1	57.1	61.6
F3Net[51]	59.4	71.9	42.2	44.7	73.4	80.3	73.2	80.4	79.6	87.3	65.5	72.9
SelfBland[57]	55.3	57.7	63.5	62.5	57.1	60.1	54.3	56.4	48.8	47.4	55.8	56.8
GANDetection[42]	47.3	45.5	47.9	57.0	51.0	56.1	50.0	44.7	49.8	49.7	49.2	50.6
LGrad [60]	59.8	88.5	45.2	46.9	72.7	79.3	68.3	76.0	75.1	80.9	64.2	74.3
Ojha [49]	69.5	80.0	64.9	74.2	81.3	90.8	50.0	49.8	66.3	74.6	66.4	73.9
NPR (our)	88.5	95.1	77.9	84.8	75.8	79.3	77.4	81.9	80.7	83.0	80.1	84.8

Table 5. **Cross-Diffusion-Sources Evaluation on the Self-Synthesis Diffusion dataset.** The images of DALLE and Midjourney are collected from the official channel of Discord. The images of other diffusion models are sampled from official pre-trained models with 1000 diffusion steps.

Method	Mean Acc. of 38 sub-testsets
CNNDetection[62]	57.3
Frank[13]	56.8
Durall[12]	56.6
Patchfor[5]	80.6
F3Net[51]	78.1
SelfBland[57]	61.2
GANDetection[42]	59.5
LGrad [60]	80.5
Ojha [49]	79.8
NPR (our)	93.3

Table 6. The mean accuracy of all 28 generation models on five datasets.

Size $l \times l$	w_j	ProGAN		StyleGAN		StyleGAN2		BigGAN		CycleGAN		StarGAN		GauGAN		Deepfake		Mean	
		Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
2×2	w_1	99.8	100.0	96.3	99.8	97.3	100.0	87.5	94.5	95.0	99.5	99.7	100.0	86.6	88.8	77.4	86.2	92.5	96.1
2×2	w_2	99.1	100.0	94.1	98.5	90.5	98.9	76.7	83.9	91.7	99.1	98.0	100.0	75.9	78.2	73.7	83.4	87.5	92.7
2×2	w_3	99.9	100.0	97.1	99.9	95.6	99.9	84.2	91.3	89.3	98.6	99.2	100.0	89.7	92.5	72.8	79.8	91.0	95.3
2×2	w_4	99.9	100.0	96.4	99.0	98.9	100.0	85.9	93.3	89.3	99.0	99.5	100.0	88.6	93.6	72.6	81.3	91.4	95.8
2×2	$avg(v_I^c)$	99.9	100.0	95.3	98.8	98.9	100.0	80.1	86.8	89.9	95.1	100.0	100.0	73.6	72.0	68.8	77.0	88.3	91.2
2×2	$max(v_I^c)$	99.9	100.0	98.8	100.0	94.9	99.9	84.2	91.8	93.1	95.8	91.3	98.6	80.0	86.0	84.9	93.0	90.9	95.6
3×3	w_1	99.9	100.00	95.8	99.9	97.8	100.0	82.5	88.1	81.7	93.7	96.7	99.6	81.4	87.9	79.8	83.2	89.4	94.1
3×3	w_2	99.9	100.0	92.6	99.3	95.0	99.8	83.1	92.3	86.0	97.5	99.5	100.0	83.3	88.9	81.7	90.6	90.1	96.1
3×3	w_3	99.6	100.0	90.8	98.5	96.1	99.8	75.2	84.0	88.0	94.2	100.0	100.0	78.7	82.5	84.0	93.0	89.0	94.0
3×3	w_4	99.6	100.0	94.7	99.8	95.3	99.9	82.7	89.7	81.9	96.7	95.4	99.8	78.9	81.7	88.3	94.5	89.6	95.3
3×3	$avg(v_I^c)$	99.9	100.0	97.6	99.9	95.3	99.8	81.8	90.6	86.9	96.6	98.5	99.9	79.9	89.0	57.3	92.4	87.1	96.0
3×3	$max(v_I^c)$	99.9	100.0	99.3	100.0	99.3	100.0	76.3	85.2	84.5	94.1	99.0	100.0	77.8	84.8	87.9	94.7	90.5	94.8

Table 7. **Effect of the hyperparameters of Neighboring Pixel Relationships.**

operations. Notably, training the detector on ProGAN images enables Neighboring Pixel Relationships (NPR) to generalize effectively to previously unseen GAN sources. This success can be attributed to NPR's unique ability to capture and analyze the distinctive traces left by the upsampling component within common GAN pipelines. The insights gained from this localized analysis contribute to NPR's effectiveness across a spectrum of GAN images.

4.2.2 Diffusion-Sources Evaluation

To present a more challenging evaluation scenario, we devise a comprehensive experiment where the detector is trained on images generated by ProGAN and subsequently tested on images produced by a diverse array of diffusion models. Three diffusion datasets are employed to preform evaluation on the diffusion-sources. It's important to note that the detectors are trained using the ProGAN 4-classes setting to ensure consistency. This evaluation setup aims to assess the detector's adaptability and performance when faced with the inherent challenges posed by diffusion-generated images, providing valuable insights into the generalization capability of Neighboring Pixel Relationships (NPR) across different image generation techniques.

The detection performance on DiffusionForensics [64] is presented in Table 3. Despite being trained on images generated by ProGAN, NPR exhibits strong generalization capabilities across various diffusion models. Our method achieves 95.3% and 99.8% in terms of mean Accuracy (Acc.) and mean Average Precision (A.P.), respectively. NPR outperforms the current state-of-the-art methods LGrad and Ojha [49] by 7.1% and 20.9%, respectively, in terms of mean Acc. metric. Additionally, when compared to DIRE [64], which specifically focuses on detection in the diffusion domain, NPR demonstrates comparable results, particularly noteworthy considering our training set comprises ProGAN images while DIRE relies on diffusion models for training.

Given that a significant portion of images in DiffusionForensics [64] belongs to the bedroom class, we further evaluate the performance on the diffusion dataset from Ojha [49]. In this dataset, diffusion models are utilized to generate images with 100 or 200 steps. The results on the Ojha diffusion dataset are presented in Table 4. The proposed NPR achieves a mean Accuracy (Acc.) value of 95.2%, demonstrating its robust performance. When compared to the current state-of-the-art methods LGrad and Ojha [49], our NPR exhibits substantial improvements, surpassing these methods by 4.3% and 8.3% in mean accuracy. This comparison underscores NPR's ability to maintain satisfactory generalization across unseen diffusion datasets.

The diffusion dataset from Ojha [49] adopts only 100 or 200 steps to generate images, which may lack clarity and realism. To address this limitation, we further collect images of DALLE and Midjourney from the official Discord channels and sample images from other models with 1000 diffusion steps. The results are reported in Table 5. The NPR achieves gains of 15.9% and 13.7% compared to LGrad and Ojha, obtaining a mean accuracy value of 80.1%. This result suggests that NPR maintains strong generalization performance even when faced with diffusion datasets generated with a more extended diffusion process (1000 steps).

In terms of the mean accuracy across 28 generation models, our NPR achieves 93.3% shown in Table 6, outperforming Ojha and LGrad by 13.5% and 12.8%, respectively. This result demonstrates that the proposed local up-sampling artifact, Neighboring Pixel Relationships, is capable of generalizing to both unseen GAN sources and diffusion sources, even when trained on ProGAN. This success can be attributed to the NPR's ability to rethink generator architectures and explore the trace of up-sampling from the perspective of local spatial information.

Different Upsampling Techniques. The performance of Neighboring Pixel Relationships on 28 generation techniques indicates a strong generalization ability to unseen sources. Despite being trained on ProGAN using nearest-neighbor up-sampling, the detector performs well on generation models with other up-sampling operations, such as bilinear. This phenomenon can be attributed to several factors: 1) Up-sampling operations are applied to feature maps, while NPR is exploited in image space. 2) NPR captures implicit artifacts representations caused by up-sampling operations. 3) The proposed NPR focuses on local and relative information, which enhances generalization ability across different upsampling techniques.

Effect of choice of NPR's hyperparameters. We evaluate the impact of NPR's size l and index j in Equation 4 on generalization ability. Simultaneously, to validate the effectiveness of the subtraction between elements in Equation 4, we replace w_j with $avg(v_I^c)$ and $max(v_I^c)$ to implement NPR. We employ the $(car, cat, chair, horse)$ of ProGAN as the training set and apply the test set of ForenSynths[62] for evaluation. The results are shown in Table 7. Observations: 1) When $l = 2$, NPR achieves better performance, likely due to most generators employing 2 scaled up-sampling layers. 2) NPR with $avg(v_I^c)$ and $max(v_I^c)$ show similar detection performance. This suggests that information in the 2×2 block of images can effectively reveal differences between real and fake images.

Qualitative Analysis of NPR. The above quantitative experiments have indicated the effectiveness of the proposed Neighboring Pixel Relationships. To obtain a more profound understanding of its intrinsic properties, we conduct a qualitative analysis of NPR, employing the visualization of Class Activation Map (CAM) [70]. Figure 4 illustrates the Class Activation Maps for images sourced from Midjourney, DALLE, and ImageNet. Notably, the CAMs for real images highlight a broader portion of the image, whereas the CAMs for fake images tend to emphasize localized regions.Intriguingly, despite the detector being primarily trained on a dataset encompassing cars, cats, chairs, and horses, it demonstrates the capacity to recognize these diverse images. Certainly, this emphasizes the generalization ability of our detector in identifying various deepfake signatures, showcasing its capacity to extend recognition capabilities beyond the training classes.

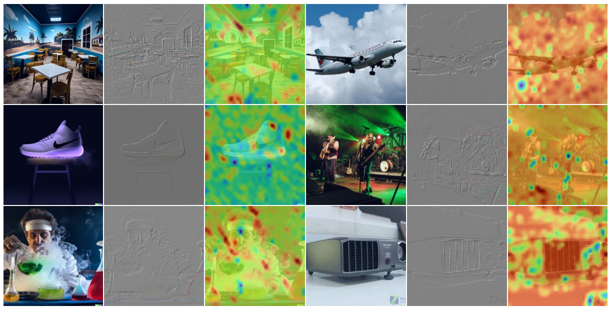


Figure 4. The visualization of CAM [70] extracted from detector on image of Midjourney, DALLE, and ImageNet. Warmer color indicates a higher probability.

5. Conclusion

This work focuses on developing a generalizable artifacts representation for both GANs and diffusions detection. We reconsider the architectures of CNN-based generators, aiming to establish source-invariant forgery detection. Our findings reveal that the up-sampling operator, beyond frequency-based artifacts, can produce generalized forgery artifacts. Existing works typically consider its influence on the whole image in the frequency domain. In contrast, we explore the trace of the up-sampling layer from the local image pixels. We present a simple but effective artifact representation, named Neighboring Pixel Relationships (NPR), to achieve generalized deepfake detection. Extensive experiments on 28 generation models indicate that the proposed NPR contributes to a strong AI-generated image detector.

6. Acknowledgments

This work was supported in part by the National Key R&D Program of China (No.2021ZD0112100), National NSF of China (No.U1936212, No.62120106009, No.U23A20314), National Natural Science Foundation of China under Grants 62072394, Natural Science Foundation of Hebei province under Grant F2021203019, and A*STAR Career Development Funding Award (Grant No:222D800031).

References

[1] Marc G Bellemare et al. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743*, 2017. 5

[2] David Berthelot et al. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017. 5

[3] Andrew Brock et al. Large scale gan training for high fidelity natural image synthesis. In *ICLR*, 2018. 5

[4] Junyi Cao et al. End-to-end reconstruction-classification learning for face forgery detection. In *CVPR*, pages 4113–4122, 2022. 2

[5] Lucy Chai et al. What makes fake images detectable? understanding properties that generalize. In *ECCV*, pages 103–120. Springer, 2020. 2, 5, 6, 7

[6] Liang Chen et al. Ost: Improving generalization of deepfake detection via one-shot test-time training. In *Advances in Neural Information Processing Systems*, 2022. 2

[7] Liang Chen et al. Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection. In *CVPR*, pages 18710–18719, 2022. 2

[8] Yunjey Choi et al. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, pages 8789–8797, 2018. 5

[9] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, pages 1251–1258, 2017. 1, 2

[10] Tan Chuangchuang, Tao Renshuai, Liu Huan, and Yao Zhao. Gangen-detection: A dataset generated by gans for generalizable deepfake detection. `https://github.com/chuangchuangtan/GANGen-Detection`, 2024. 5

[11] Prafulla Dhariwal et al. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2, 5

[12] Ricard Durall et al. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *CVPR*, pages 7890–7899, 2020. 1, 2, 3, 4, 5, 6, 7

[13] Joel Frank et al. Leveraging frequency analysis for deep fake image recognition. In *ICML*, pages 3247–3258. PMLR, 2020. 1, 2, 3, 4, 5, 6, 7

[14] Ian J Goodfellow et al. Generative adversarial nets. In *NIPS*, 2014. 1

[15] Shuyang Gu et al. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 5

[16] Alexandros Haliassos et al. Lips don't lie: A generalisable and robust approach to face forgery detection. In *CVPR*, pages 5039–5049, 2021. 2

[17] Kaiming He et al. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5

[18] Yang He et al. Beyond the spectrum: Detecting deepfakes via re-synthesis. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 2534–2541. International Joint Conferences on Artificial Intelligence Organization, 2021. 2

[19] Zhenliang He et al. Attgan: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing*, 28(11):5464–5478, 2019. 5

[20] Jonathan Ho et al. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851, 2020. 1, 2, 5

[21] Yonghyun Jeong et al. Bihpf: Bilateral high-pass filters for robust deepfake detection. In *WACV*, pages 48–57, 2022. 2, 3, 4, 5, 6

[22] Yonghyun Jeong et al. Fingerprintnet: Synthesized fingerprints for generated image detection. In *ECCV*, pages 76–94. Springer, 2022. 2

[23] Yonghyun Jeong et al. Frepgan: robust deepfake detection using frequency-level perturbations. In *AAAI*, pages 1060–1068, 2022. 2, 3, 4, 5, 6

[24] Yan Ju et al. Fusing global and local features for generalized ai-synthesized image detection. In *2022 IEEE ICIP*, pages 3465–3469. IEEE, 2022. 2

[25] Tero Karras et al. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018. 1, 5

[26] Tero Karras et al. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 1, 5

[27] Tero Karras et al. Analyzing and improving the image quality of stylegan. In *CVPR*, pages 8110–8119, 2020. 5

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015. 5

[29] Kwot Sin Lee et al. Infomax-gan: Improved adversarial image generation via information maximization and contrastive learning. In *WACV*, pages 3942–3952, 2021. 5

[30] Chuqiao Li, Zhiwu Huang, Danda Pani Paudel, Yabin Wang, Mohamad Shahbazi, Xiaopeng Hong, and Luc Van Gool. A continual deepfake detection benchmark: Dataset, methods, and essentials. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1339–1349, 2023. 3

[31] Chun-Liang Li et al. Mmd gan: Towards deeper understanding of moment matching network. *Advances in neural information processing systems*, 30, 2017. 5

[32] Yuezun Li et al. In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International workshop on information forensics and security (WIFS)*, pages 1–7. IEEE, 2018. 2

[33] Tsung-Yi Lin et al. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 5

[34] Huan Liu, Zichang Tan, Qiang Chen, Yunchao Wei, Yao Zhao, and Jingdong Wang. Unified frequency-assisted transformer framework for detecting and grounding multi-modal manipulation. *arXiv preprint arXiv:2309.09667*, 2023. 1

[35] Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Yao Zhao, and Jingdong Wang. Forgery-aware adaptive transformer for generalizable synthetic image detection. *arXiv preprint arXiv:2312.16649*, 2023.

[36] Huan Liu, Xiaolong Liu, et al. Padvg: A simple baseline of active protection for audio-driven video generation. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2024. 1

[37] Luping Liu et al. Pseudo numerical methods for diffusion models on manifolds. In *ICLR*, 2022. 5

[38] Ming Liu et al. Stgan: A unified selective transfer network for arbitrary image attribute editing. In *CVPR*, pages 3673–3682, 2019. 5

[39] Ziwei Liu et al. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015. 5

[40] Mario Lučić et al. High-fidelity image generation with fewer labels. In *ICML*, pages 4183–4192. PMLR, 2019. 5

[41] Yuchen Luo et al. Generalizing face forgery detection with high-frequency features. In *CVPR*, pages 16317–16326, 2021. 3

[42] Sara Mandelli et al. Detecting gan-generated images by orthogonal training of multiple cnns. In *2022 IEEE ICIP*, pages 3091–3095. IEEE, 2022. 5, 6, 7

[43] Francesco Marra et al. Do gans leave artificial fingerprints? In *2019 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 506–511. IEEE, 2019. 2

[44] Iacopo Masi et al. Two-branch recurrent network for isolating deepfakes in videos. In *ECCV*, pages 667–684. Springer, 2020. 3

[45] Takeru Miyato et al. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 5

[46] Alex Nichol et al. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 5

[47] Alexander Quinn Nichol et al. Improved denoising diffusion probabilistic models. In *ICML*, pages 8162–8171. PMLR, 2021. 5

[48] Weili Nie et al. Relgan: Relational generative adversarial networks for text generation. In *ICLR*, 2019. 5

[49] Utkarsh Ojha et al. Towards universal fake image detectors that generalize across generative models. In *CVPR*, pages 24480–24489, 2023. 2, 3, 5, 6, 7, 8

[50] Taesung Park et al. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, pages 2337–2346, 2019. 5

[51] Yuyang Qian et al. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *ECCV*, pages 86–103. Springer, 2020. 2, 3, 5, 6, 7

[52] Aditya Ramesh et al. Zero-shot text-to-image generation. In *ICML*, pages 8821–8831. PMLR, 2021. 5

[53] Robin Rombach et al. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 5

[54] Andreas Rossler et al. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the ICCV*, pages 1–11, 2019. 2, 5

[55] Olga Russakovsky et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 5

[56] Christoph Schuhmann et al. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *NeurIPS Workshop Datacentric AI*, number FZJ-2022-00923. Jülich Supercomputing Center, 2021. 5

[57] Kaede Shiohara et al. Detecting deepfakes with self-blended images. In *CVPR*, pages 18720–18729, 2022. 2, 5, 6, 7

[58] Chuangchuang Tan, Ping Liu, RenShuai Tao, Huan Liu, Yao Zhao, Baoyuan Wu, and Yunchao Wei. Data-independent operator: A training-free artifact representation extractor for generalizable deepfake detection, 2024. 3

[59] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Frequency-aware deepfake detection: Improving generalizability through frequency space learning. *arXiv preprint arXiv:2403.07240*, 2024. 3

[60] Chuangchuang Tan et al. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *CVPR (CVPR)*, pages 12105–12114, 2023. 2, 3, 5, 6, 7

[61] Chengrui Wang et al. Representative forgery mining for fake face detection. In *CVPR*, pages 14923–14932, 2021. 2

[62] Sheng-Yu Wang et al. Cnn-generated images are surprisingly easy to spot... for now. In *CVPR*, pages 8695–8704, 2020. 2, 4, 5, 6, 7, 8

[63] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, and Houqiang Li. Altfreezing for more general video face forgery detection. In *CVPR*, pages 4129–4138, 2023. 3

[64] Zhendong Wang et al. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22445–22455, 2023. 5, 6, 7

[65] Simon Woo et al. Add: Frequency attention and multi-view based knowledge distillation to detect low-quality compressed deepfake images. In *AAAI*, pages 122–130, 2022. 3

[66] Fisher Yu et al. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5

[67] Ning Yu et al. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the ICCV*, pages 7556–7566, 2019. 2

[68] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *ICCV*, pages 19148–19158, 2023. 3

[69] Xu Zhang et al. Detecting and simulating artifacts in gan fake images. In *2019 IEEE international workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2019. 2

[70] Bolei Zhou et al. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 8

[71] Hongguang Zhu, Yunchao Wei, et al. Ctp: Towards vision-language continual pretraining via compatible momentum contrast and topology preservation. In *ICCV*, pages 22257–22267, 2023. 3

[72] Jiapeng Zhu et al. In-domain gan inversion for real image editing. In *ECCV*, pages 592–608. Springer, 2020. 1

[73] Jun-Yan Zhu et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2223–2232, 2017. 5