Learning on Gradients: Generalized Artifacts Representation for GAN-Generated Images Detection

Chuangchuang Tan[1,2], Yao Zhao[1,2]*, Shikui Wei[1,2], Guanghua Gu[3,4], 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
{21112002, yzhao, shkwei}@bjtu.edu.cn, guguanghua@ysu.edu.cn, wychao1987@gmail.com

Abstract

Recently, there has been a significant advancement in image generation technology, known as GAN. It can easily generate realistic fake images, leading to an increased risk of abuse. However, most image detectors suffer from sharp performance drops in unseen domains. The key of fake image detection is to develop a generalized representation to describe the artifacts produced by generation models. In this work, we introduce a novel detection framework, named Learning on Gradients (LGrad), designed for identifying GAN-generated images, with the aim of constructing a generalized detector with cross-model and cross-data. Specifically, a pretrained CNN model is employed as a transformation model to convert images into gradients. Subsequently, we leverage these gradients to present the generalized artifacts, which are fed into the classifier to ascertain the authenticity of the images. In our framework, we turn the data-dependent problem into a transformation-model-dependent problem. To the best of our knowledge, this is the first study to utilize gradients as the representation of artifacts in GAN-generated images. Extensive experiments demonstrate the effectiveness and robustness of gradients as generalized artifact representations. Our detector achieves a new state-of-the-art performance with a remarkable gain of 11.4%. The code is released at https://github.com/chuangchuangtan/LGrad.

1. Introduction

Over the past years, remarkable progress has been made in deep generative models, *i.e.* generative adversarial networks (GAN) [14], its variations [3, 21–23, 36], and VAE [25]. The generated media is highly realistic and indistinguishable from real to human eyes. Although it has the potential for many novel applications [8], it also brings new risks due to the abuse of fake information. The misuse of DeepFakes has been confirmed that some people swap the faces of women onto pornographic videos [8]. In addition, some individuals, even non-experts, can malevolently manipulate or create fake images or videos for political or economic purposes, leading to serious social problems [17]. Thus, it is extremely necessary to develop forgery detection techniques to help people determine the credibility of the

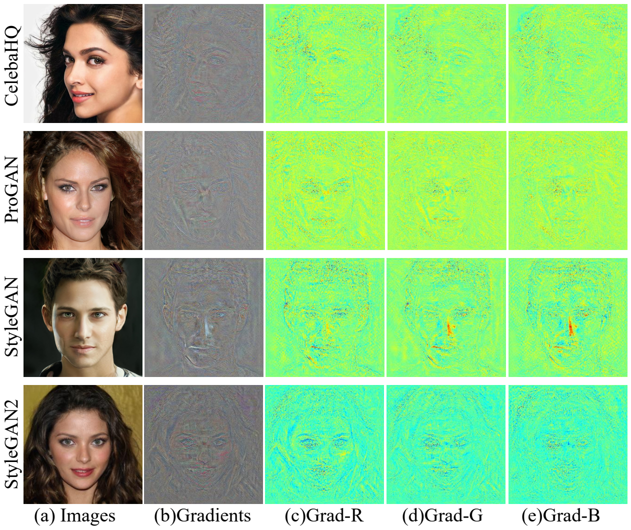

Figure 1. Visualization of gradients of real images and GAN-generated images extracted from a pre-trained model. To fully understand the gradients, heatmaps for R, G, and B channels also are shown, where red is high. In the gradients, the content of images is filtered out, and only the discriminative pixels are retained for the pre-trained model's target task. We utilize the gradients as the generalized artifacts representation to develop a novel detection framework.

*Corresponding author

media [17,40].

Various detectors [11–13,16,18,19,30,42,45] have been developed to detect GAN-generated images. Some studies [16, 30, 45] focus on human face images, while others [11,18,19,42] handle various categories of images. They mainly depend on local regions artifacts [4, 44], blending boundary [26], global textures [30], and frequency-level artifacts [13, 18, 19, 45]. However, those methods heavily rely on the training settings, resulting in failure detection of images from unseen categories or GAN models. The test images in the actual scene are always from unknown sources [17], rendering it challenging to develop generalized detectors. There are some works [19, 42] exploiting pre-processing, data augmentation, and reducing the effects of frequency-level artifacts to develop a robust detector. Nevertheless, there still needs to be a more generalized representation of the clue produced by generation models, which is critical for robust fake image detection.

To tackle this problem, we propose a novel and simple detection framework, referred to as Learning on Gradients (LGrad). A new generalized feature, Gradients, is developed to serve as a representation of the artifacts produced by GAN models. We believe that the gradients of a trained CNN model can highlight the important pixels in the target task, thereby serving as a valuable cue for detecting fake images. As shown in Figure 1, we adopt a pre-trained discriminator of ProGAN [21] to extract gradients of images produced by Celeba-HQ [21], ProGAN [21], StyleGAN [22], StyleGAN2 [23]. In these gradients, the content of images is filtered out, and only the discriminative pixels that are relevant to the pre-trained model's target task are retained. Therefore, the gradients are more dependent on the pre-trained model rather than on training sources, thereby enhancing the detector's performance with unseen data. In our framework, a pretrained model, called *transformation model*, is employed to convert images to gradients. These gradients serve as the generalized artifacts and are fed into the classifier to obtain a robust detector. Since the transformation model is indeterminate in our framework, targeted anti-detection cannot be effectively launched.

To validate the performance of our LGrad, we only use images generated by ProGAN to train the detector and evaluate it with various sources, including cross-category, cross-model, and cross-model & category. Numerous experiments prove the effectiveness and robustness of gradients as generalized artifacts. Our detector achieves a new state-of-the-art performance in known and unseen settings.

Our paper makes the following contributions:

- We develop a new detection framework, Learning on Gradient (LGrad), to detect GAN-generated images. Our detector achieves a new state-of-the-art performance.
- We introduce a new generalized artifact representation, Gradients, for GAN-generated image detection. Furthermore, we are the first to use gradients as the representation of artifacts.
- Our framework turns the data-driven problem into a transformation-model-driven problem. The robustness of the detector is improved with the introduction of the transformation model.
- We prove the great potential of the discriminator of GAN in detecting GAN-generated images.

2. Related Work

In this section, we discuss recent approaches for generated image detection. The previous methods attempt to exploit spatial information or frequency artifacts as the representation of clues produced by the generation model for fake image detection.

2.1. Image-based Fake Detection

Early studies employ spatial information from generated images to identify the fake images, such as color spaces [45] and global texture [30]. Rossler *et al*. [37] introduce a large-scale manipulated face dataset FaceForensics++ and adopt Xception [7] as detector to identify the fake images with compression. Li *et al*. [27] detect fake face videos by analyzing eye blinking in the videos. ForensicTransfer [9] presents a new autoencoder-based architecture to learn a discriminative feature representation in the latent space for detection well on new domains. Yu *et al*. [44] and Marra *et al*. [31] believe that every GAN model has unique fingerprints after training, and extracting it from generated images to perform detection. Bayar *et al*. [2] design a new convolutional network architecture to suppress the image of the content and learn manipulation features for a generalized detector. Gram-Net [30] leverages the Gram matrix to extract global texture as the robust representation for fake face detection. Face X-ray [26] develops the blending boundary to present the artifacts for detecting manipulated face images. Chai *et al*. [4] design a patch-based classifier to limit the receptive field of the model for learning redundant artifacts in image patches. The patch-based predictions focus on local artifacts rather than global structure, improving the robustness of the detector. Yu *et al*. [45] use the channel difference image and spectrum image to mine the intrinsic clues of images from the view of the camera imaging process. Wang *et al*. [42] directly adopt a large number of real and fake images with data augmentation to train a binary classifier for improving unseen image performance. They use some post-processing operations to improve robustness during training, such as JPEG compression, blurring, and resizing. PCL [46] extracts the distinct source features of

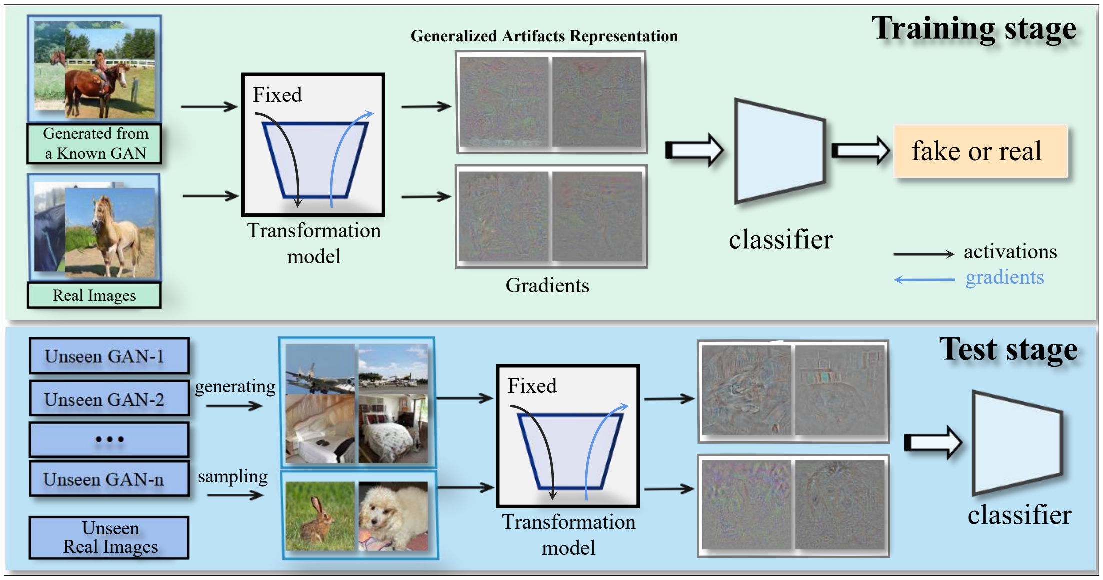

Figure 2. The overall pipeline of the proposed framework. The method uses the same transformation model and the same classifier in the testing and training phases.

images by self-consistency learning for detecting deepfake. He *et al*. [16] design a re-synthesis procedure to extract visual cues for robust deepfake detection.

2.2. Frequency-based Fake Detection

Since GAN architectures depend on up-scaling operations, the generated images contain unique frequency-level artifacts compared to neutral images [11]. Several works [1, 11, 12, 31, 32, 34] leverage the spectral distributions of images as the representation of artifacts for fake image detection. LOG [32] proposes a two-branch recurrent network to integrate information from the color domain and the frequency domain for detecting manipulated faces. F^3-Net [34] designs a two-stream collaborative learning network to mine the forgery patterns in fake images, which exploits two frequency-aware clues, frequency-aware decomposed image components and local frequency statistics, to present subtle forgery patterns and high-level semantics, respectively. BiHPF [18] adopts two high-pass filters to amplify the magnitudes of the artifacts. FrePGAN [19] validates that the frequency-level artifacts of fake images are evident but uniquely vary by the type of GAN model or object category, which can cause overfitting issues for fake image detection. Thus, it generated the frequency-level perturbation maps to remove the effect of the frequency-level artifacts.

3. Learning on Gradients framwork

The key to the fake image detection task is developing a generalized representation of artifacts generated by GAN models. It should be generalized and robust enough to span diverse categories and different GAN models. In order to achieve this, we design a novel detection framework to improve the cross-source performance in this work. The overview of our method is shown in Figure 2. Our framework employs gradients as the generalized representation to obtain the robust detector. We transform the images to the gradients by a pretrained CNN model named transformation model. In the process of transforming, most of the content of images is filtered out due to the pooling layer in the CNN model, and the essential pixels for the transformation model are highlighted. Figure 1 provides the visualization evidence for our explanation. Extensive experiments in Section 4 demonstrate the effectiveness of the gradients using as the artifact representation.

3.1. Transformation to Gradients

The previous methods based on pixel and frequency-level artifacts highly rely on the training data, resulting in failure with unseen images. To solve this overfitting issue, we adopt any one pretrained CNN model as the transformation model to convert all images consisting of training images and test images to gradients. This approach effectively maps all data to the gradient domain, which is determined by the chosen transformation model. By relying on this more generalized representation of artifacts, we can achieve improved performance in detecting GAN-generated images across diverse categories and GAN models.

We denote the images set as $I = \{(I_i, y_i)\}_{i=0}^{N-1}$, where $y_i = \{0, 1\}$ is the label of the images $I_i \in \mathbb{R}^{w \times h \times c}$, which is $real(y = 0)$ or $fake(y = 1)$. The w, h, c present width, height, and the number of channels. The transformation model is defined as $M(\cdot)$. We first feed images I_i into M model.

$$l = M(I_i), \tag{1}$$

where $l \in \mathbb{R}^{n \times c}$ is the output vectors of the transformation model M. Then, we calculate the gradients of $sum(l)$ with respect to the input I_i,

$$G = \frac{\partial sum(l)}{\partial I_i}. \tag{2}$$

G is the generalized artifacts representation of our method. Note that the transformation model M is fixed, and its parameter would not update in our framework. The quality of gradients G highly depends on the transformation model M, which is helpful to reduce the reliance on training data and enhance the generalization of the representations. In our experiments, we adopt various popular CNN models to implement the transformation model, including the classification model, segmentation model, discriminator of GAN, contrastive learning model, and GAN Inversion.

3.2. Detecting Fake Images

During the training stage of our framework, we use normalized gradients ranging from 0 to 255 to train the binary classification network. We adopt the popular classifier to distinguish whether the input gradients correspond to GAN-generated images or not, and employ a cross-entropy loss function to optimize the network. During the inference time, the test images are first converted to gradients by the transformation model employed in the training stage. Then, we obtain the final result by feeding gradients into the trained classification network.

4. Experimental Results

4.1. Dataset

To evaluate the proposed approach, we adopt the dataset provided by Wang *et al.* [42] to conduct experiments following the previous works [18, 19]. It contains various fake images generated by ProGAN [21], StyleGAN [22], StyleGAN2 [23], BigGAN [3], CycleGAN [49], StarGAN [6], GauGAN [33] and Deepfake [37]. The real images are sampled from LSUN [43], ImageNet [38], CelebA [29], CelebA-HQ [21], COCO [28], and FaceForensics++ [37]. During the training stage, we train the classifier with images generated by ProGAN.

In addition, to verify the usefulness of our method for face data, we sample 20,000 real images from Celeba-HQ, and generate 20,000 face images by ProGAN as the training set. And the images generated by StyleGAN, StyleGAN2 trained by Celeba-HQ [21] are employed as the test set. All images are resized to 256×256 resolution.

4.2. Implementation Details

We apply the Resnet50 [15] model pre-trained with ImageNet [38] as the classifier, and train it by Adam [24] with learning rate of 5×10^{-4}. The learning rate is decreased by twenty percent after every ten epochs. The batch size is set to 16, and the number of epochs is 100. In this paper, the results of the last epoch are used as the final performance. In addition, we randomly crop the input images during the training phase and directly feed images into the classifier in the test stage.

To fully understand the gradients as the generalized artifacts, we further employ various popular models to implement the transformation model, $i.e.$ VGG16 [39], InceptionV3 [41], Resnet50 [15], DeeplabV3 [5], CLIP [35], ViT [10], discriminator of ProGAN [21], StyleGAN [22], StyleGAN2 [23], and GAN Idinvert [48]. All of those models in this paper are released by the official code. Following the baselines [18, 19], we utilize the average precision score (A.P.) and accuracy (Acc.) as evaluation metrics to evaluate the proposed method.

4.3. Detection Performance Evaluations

To assess the effectiveness of our detector, we conducted evaluations in four different settings: cross-model images, cross-category images, cross-model & category images, and perturbed images. Specifically, the detector is trained on a known GAN model, $e.g.$ *ProGAN-Horse* [21] dataset, while evaluated with unseen images: 1) cross-category images, produced by the same GAN model trained on different datasets, $e.g.$ *ProGAN-Airplane* and *ProGAN-Diningtable*; 2) cross-model images, generated by different GAN models trained on the same dataset, $e.g.$ *StyleGAN-Horse* [22] and *StyleGAN2-Horse* [23]; and 3) cross-model & category images, where both training data and GAN model are different from the training setting, $e.g.$ *BigGAN-ImageNet* [3, 38] and *GauGAN-COCO* [28, 33]. Meanwhile, to fully understand the proposed LGrad framework in terms of different transformation models, we further implement the transformation model with various popular CNN networks.

Trans. Models	ProGAN-CelebaHQ		StyleGAN-CelebaHQ		StyleGAN2-CelebaHQ	
	Acc.	A.P.	Acc.	A.P.	Acc.	A.P.
Input Image	99.6	100.0	76.8	98.9	64.2	96.2
ProGAN-RandomInit	91.4	97.3	57.4	66.3	52.4	56.9
VGG16	96.9	99.8	79.6	96.3	68.2	91.6
CLIP-RN50	99.5	100.0	99.4	100.0	99.2	100.0
ProGAN-bedroom	98.8	100.0	98.4	99.9	96.5	99.6
ProGAN-bridge	96.4	99.5	84.8	95.0	81.9	93.6

Table 1. Cross-model Performance on face images data.

Test-categorys	CLIP Trans.		Bedroom Trans.		Bridge Trans.	
	ACC	AP	ACC	AP	ACC	AP
airplane	92.9	97.0	99.4	100.0	99.4	100.0
bicycle	80.5	94.6	94.5	98.5	94.8	98.5
bird	85.6	94.4	99.2	100.0	97.5	99.7
boat	90.6	96.6	99.6	100.0	98.5	99.9
bottle	94.7	98.2	97.7	100.0	99.2	100.0
bus	81.5	94.9	98.1	100.0	96.7	99.8
car	84.0	95.7	99.7	100.0	99.1	100.0
cat	90.2	96.1	99.7	100.0	98.7	99.9
chair	94.5	98.5	96.2	99.9	99.1	100.0
cow	85.5	95.0	99.5	100.0	97.7	99.8
diningtable	92.4	97.7	98.5	99.8	96.9	99.5
dog	89.8	96.1	99.6	100.0	98.4	99.9
horse	89.0	96.3	100.0	100.0	99.8	100.0
motorbike	79.2	94.8	97.0	99.7	96.9	99.3
person	93.6	97.1	98.9	100.0	99.1	100.0
pottedplant	77.9	95.6	97.5	99.7	95.7	99.1
sheep	83.2	94.2	99.2	100.0	95.8	99.3
sofa	94.1	98.6	99.1	100.0	99.2	100.0
train	83.6	94.4	94.4	99.7	96.1	99.6
tvmonitor	93.0	98.7	96.5	100.0	99.2	100.0
Mean	87.8	96.2	98.2	99.9	97.9	99.7

Table 2. Cross-category Performance on the ProGAN models trained on different LSUN [43] object datasets.

4.3.1 Cross-model Performance.

We employ face images to show the cross-model performance of our proposed LGrad framework. The training set contains 25,000 fake images generated by ProGAN [21] and 25,000 real images randomly sampled from Celeba-HQ [21]. The test set consists of 10,000 images, half of which are real images randomly sampled from Celeba-HQ [21], while the other half are fake images generated by StyleGAN [22] and StyleGAN2 [23]. We adopt popular CNN models such as VGG16 [39], CLIP-RN50 [35], and discriminators of ProGAN [21] as the transformation models, and also randomly initialize a discriminator of ProGAN to perform the transformation to demonstrate the effectiveness of pre-trained models. In addition, we train the detector directly on the images to compare it with the gradients-based detector.

The experimental results are shown in Table 1. We can observe that image-based detectors suffer from sharp performance drops on StyleGAN [22] and StyleGAN2 [23]. It only achieves the Acc. of 64.2% on StyleGAN2, which is much lower than the performance on ProGAN. In contrast, all gradients-based detectors transformed by pre-trained models outperform the one with random initialization of ProGAN. The results demonstrate that the transformation model eliminates the contents of images while retaining the discriminative regions in the gradients. Thus, the gradients can effectively present the generalized artifacts produced by GAN models. The detector based on CLIP-RestNet50, trained on various (image, text) pairs, achieves the best accuracy of 99.4% and 99.2% on StyleGAN and StyleGAN2, respectively. The discriminators of ProGAN trained on bedroom or bridge datasets yield comparable results with the CLIP-RestNet50 model. Our proposed framework shows remarkable performance on face data.

4.3.2 Cross-category Performance.

We leverage the training set from Wang *et al*. [42] to validate the cross-category performance, which contains 20 object categories generated by ProGAN models. Each category comprises 18,000 fake images and an equal number of real images. We trained our detector using the horse dataset and evaluated its performance on the remaining 19 datasets. We implement the transformation model with two discriminators of ProGAN and Contrastive learning model, including ProGAN-bedroom, ProGAN-bridge, and CLIP-ResNet50. Table 2 reports the results of our detectors. The performance of the horse is calculated on the training data. When adopting the CLIP-RestNet50 model as the transformation model, the detector achieves the best result on cross-model performance compared to the other models. We also evaluate the performance of CLIP-RestNet50 on the cross-category case. In Table 2, the CLIP-RestNet50 achieves mean Acc. of 87.8% on 20 categories, while the ProGAN-bedroom and ProGAN-bridge obtain higher accuracies of 98.2% and 97.9%, respectively. It can be observed that discriminators are more suitable as the transformation model to detect cross-category images. In summary, we can confirm that our detectors show excellent performance in detecting unseen categories.

4.3.3 Cross-model & category Performance.

We further expand the testing scope to evaluate the robustness of our proposed method. The gradients-based detector is performed on eight different models following baselines [18, 19] including ProGAN, StyleGAN, StyleGAN2, BigGAN, CycleGAN, StarGAN, GauGAN, and Deepfake. We adopt the discriminators of ProGAN and StyleGAN, trained using the LSUN bedroom dataset, as the transformation model. For fair comparisons, we use the same experimental setting employed in baselines [19] to evaluate the generality of the detector. Specifically, three training configurations are employed to train the detector, including 1-class (horse), 2-class (chair, horse), and 4-class (car, cat, chair, horse) settings.

Table 3 presents the results of our proposed LGrad framework in comparison to various previous methods. We

Methods	Settings		Test Models																	
	Input	#class	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.
Wang [42]	Image	1	50.4	63.8	50.4	79.3	68.2	94.7	50.2	61.3	50.0	52.9	50.0	48.2	50.3	67.6	50.1	51.5	52.5	64.9
Frank [12]	Freq	1	78.9	77.9	69.4	64.8	67.4	64.0	62.3	58.6	67.4	65.4	60.5	59.5	67.5	69.1	52.4	47.3	65.7	63.3
Durall [11]	Freq	1	85.1	79.5	59.2	55.2	70.4	63.8	57.0	53.9	66.7	61.4	99.8	99.6	58.7	54.8	53.0	51.9	68.7	65.0
BiHPF [18]	Freq	1	82.5	81.4	68.0	62.8	68.8	63.6	67.0	62.5	75.5	74.2	90.1	90.1	73.6	92.1	51.6	49.9	72.1	72.1
FrePGAN [19]	Image	1	95.5	99.4	80.6	90.6	77.4	93.0	63.5	60.5	59.4	59.9	99.6	100.0	53.0	49.1	70.4	81.5	74.9	79.3
LGrad (ProGAN-bedroom)	Grad	1	98.4	99.9	82.6	95.6	83.3	98.4	76.2	81.8	82.3	90.6	99.7	100.0	71.7	75.0	52.8	57.8	80.9	87.4
LGrad (StyleGAN-bedroom)	Grad	1	99.4	99.9	96.0	99.6	93.8	99.4	79.5	88.9	84.7	94.4	99.5	100.0	70.9	81.8	66.7	77.9	**86.3**	**92.7**
Wang [42]	Image	2	64.6	92.7	52.8	82.8	75.7	96.6	51.6	70.5	58.6	81.5	51.2	74.3	53.6	86.6	50.6	51.5	57.3	79.6
Frank [12]	Freq	2	85.7	81.3	73.1	68.5	75.0	70.9	76.9	70.8	86.5	80.8	85.0	77.0	67.3	65.3	50.1	55.3	75.0	71.2
Durall [11]	Freq	2	79.0	73.9	63.6	58.8	67.3	62.1	69.5	62.9	65.4	60.8	99.4	99.4	67.0	63.0	50.5	50.2	70.2	66.4
BiHPF [18]	Freq	2	87.4	87.4	71.6	74.1	77.0	81.1	82.6	80.6	86.0	86.6	93.8	80.8	75.3	88.2	53.7	54.0	78.4	79.1
FrePGAN	Image	2	99.0	99.9	80.8	92.0	72.2	94.0	66.0	61.8	69.1	70.3	98.5	100.0	53.1	51.0	62.2	80.6	75.1	81.2
LGrad (ProGAN-bedroom)	Grad	2	99.5	100.0	85.8	99.3	83.5	99.4	78.9	87.7	78.8	89.0	99.6	100.0	70.5	77.6	51.9	52.7	81.1	88.2
LGrad (StyleGAN-bedroom)	Grad	2	99.8	100.0	94.8	99.7	92.4	99.6	82.5	92.4	85.9	94.7	99.7	99.9	73.7	83.2	60.6	67.8	**86.2**	**92.2**
Wang [42]	Image	4	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 [12]	Freq	4	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 [11]	Freq	4	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
BiHPF [18]	Freq	4	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 [19]	Image	4	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 (ProGAN-bedroom)	Grad	4	99.7	100.0	87.8	99.1	91.7	99.7	80.9	89.3	78.2	89.0	99.8	100.0	73.5	78.6	53.1	55.0	83.1	88.8
LGrad (StyleGAN-bedroom)	Grad	4	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**

Table 3. Classification accuracy with cross-model & category.

Perturbed	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.
No	99.4	99.9	96.0	99.6	93.8	99.4	79.5	88.9	84.7	94.4	99.5	100.0	70.9	81.8	66.7	77.9	86.3	92.7
blur	90.1	96.5	90.6	95.7	87.5	93.4	71.1	76.3	70.6	73.6	93.6	98.0	63.7	69.5	61.2	62.6	78.5	83.2
cropping	99.2	99.9	95.9	99.6	94.0	99.6	80.1	88.0	78.5	88.1	94.4	99.4	70.6	78.1	67.5	82.3	85.0	91.9
jpeg	76.2	90.0	74.4	90.2	72.6	89.1	66.0	74.6	72.7	83.2	76.0	89.5	60.1	67.6	58.0	65.2	69.5	81.2
noise	77.1	87.7	73.3	84.8	74.3	84.9	68.2	77.4	66.4	77.2	76.0	88.8	60.4	69.1	57.5	65.1	69.1	79.4
combined	86.3	94.6	83.5	93.0	82.4	92.3	71.2	78.9	73.0	80.4	84.7	94.3	64.7	71.5	61.2	67.4	75.9	84.1

Table 4. Performance of common image perturbations.

can find that the proposed LGrad framework successfully surpasses the counterparts in the metric of average precision score and accuracy. Our detector shows significantly better generalization to other methods, except on GauGAN and Deepfake. Notably, our LGrad framework achieves higher accuracy values increase to 99.4%, 96.0%, 93.8%, 79.5%, 84.7% on the ProGAN, StyleGAN, StyleGAN2, BigGAN, CycleGAN models with the 1-class setting, respectively. We obtain comparable results on the GauGAN model and deepfake. Furthermore, our LGrad with StyleGAN-bedroom model successfully achieves a new state-of-the-art accuracy, surpassing FrePGAN [19] by 11.4% and 13.4% in terms of mean accuracy and mean average precision score, respectively. We also compare to FingerprintNet [20] evaluated on six GAN models. When evaluated on StyleGAN, StyleGAN2, BigGAN, CycleGAN, StarGAN, and GauGAN, FingerprintNet and the proposed method achieve the mean Accuracy of 82.6% and 87.4%, respectively. We achieve a gain of 4.8% in terms of mean accuracy.

Trans. Model	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.
VGG16 [39]	96.0	99.5	65.5	88.8	74.7	93.8	73.0	78.1	77.8	86.1	99.8	100.0	60.7	63.1	60.4	67.8	76.0	84.7
InceptionV3 [41]	64.9	74.4	58.8	66.9	65.4	73.8	50.9	52.1	59.1	68.4	52.6	58.5	54.0	56.5	49.8	50.1	56.9	62.6
Resnet50 [15]	86.4	95.0	81.0	92.2	83.7	93.5	57.2	56.2	68.3	75.8	96.4	99.5	51.2	52.7	63.4	70.4	73.4	79.4
CLIP-Resnet50 [35]	87.6	95.8	80.2	90.0	78.9	91.0	60.1	61.1	84.2	87.9	88.5	95.1	72.6	71.6	64.9	64.4	77.1	82.1
ViT [10]	51.2	71.1	51.7	65.7	52.4	67.9	50.6	53.3	54.1	75.2	51.6	64.0	50.6	61.1	50.0	53.7	51.5	64.0
DeeplabV3 [5]	81.6	91.6	68.7	80.4	70.6	84.5	54.5	55.7	66.2	71.0	87.9	94.7	51.7	53.1	59.3	58.9	67.6	73.7
Idinvert [48]	97.4	99.8	71.6	95.3	71.2	95.4	86.6	94.8	78.7	85.7	97.4	99.7	72.0	82.1	60.1	72.1	79.4	90.6
ProGAN-bedroom [21]	98.4	99.9	82.6	95.6	83.3	98.4	76.2	81.8	82.3	90.6	99.7	100.0	71.7	75.0	52.8	57.8	80.9	87.4
ProGAN-bridge [21]	97.8	99.7	86.4	97.5	85.7	97.3	72.5	78.7	76.8	87.5	94.1	99.9	62.5	75.8	53.2	61.3	78.6	87.2
StyleGAN-bedroom [22]	99.4	99.9	96.0	99.6	93.8	99.4	79.5	88.9	84.7	94.4	99.5	100.0	70.9	81.8	66.7	77.9	**86.3**	**92.7**
StyleGAN-cats [22]	97.4	99.7	83.4	97.3	77.4	96.4	69.8	74.6	79.3	90.2	97.8	99.8	68.0	77.4	65.9	72.9	79.9	88.5
StyleGAN2-church [23]	99.1	100.0	88.2	97.7	91.9	99.6	70.1	71.7	80.6	89.1	95.6	99.8	60.8	68.9	72.7	76.5	82.4	87.9

Table 5. Performance of different transformation models.

Methods	Num. of	ProGAN		StyleGAN		StyleGAN2		BigGAN		CycleGAN		StarGAN		GauGAN		Deepfake		Mean	
	Train. data	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.
FrePGAN [19]	36k	95.5	99.4	80.6	90.6	77.4	93.0	63.5	60.5	59.4	59.9	99.6	100.0	53.0	49.1	70.4	81.5	74.9	79.3
LGrad	4k	95.9	99.5	88.9	98.0	91.7	99.0	59.1	55.2	58.7	58.4	87.8	99.8	59.2	66.2	65.9	83.5	75.9	82.4
LGrad	9k	98.3	99.9	92.4	99.4	92.2	99.4	71.7	74.5	73.0	76.8	98.1	100.0	64.0	74.2	68.0	79.3	82.2	87.9
LGrad	18k	98.8	100.0	95.1	99.8	91.4	99.7	76.8	87.1	79.8	90.7	99.2	100.0	68.9	80.4	63.9	73.0	84.2	91.3
LGrad	36k	99.4	99.9	96.0	99.6	93.8	99.4	79.5	88.9	84.7	94.4	99.5	100.0	70.9	81.8	66.7	77.9	**86.3**	**92.7**
LGrad	72k	99.8	100.0	94.8	99.7	92.4	99.6	82.5	92.4	85.9	94.7	99.7	99.9	73.7	83.2	60.6	67.8	86.2	92.2

Table 6. Results with variance in number of training data.

In comparison to the performance of FrePGAN [19], our detector obtains the most significant improvement on StyleGAN, increasing accuracy by 15.4%, 14.0%, 14.1% with 1-class, 2-class, and 4-class settings, respectively. It is worth noting that the LGrad framework fails to perform well on the deepfake model, which is not a GAN model and is trained with MSE loss and SSIM loss, resulting in an accuracy value of only 66.7%. The results indicate that the proposed LGrad approach is less reliant on the amount of data, as it achieves similar results with 1-class, 2-class, and 4-class settings. As the number of training data increases, the deepfake performance decreases, resulting in a drop in the mean accuracy of all generative models.

In addition, we leverage the high-frequency component of images as the artifacts representation to train the classifier with the same settings. High-frequency-based method obtains the mean accuracy of 75.1%, 75.0%, and 73.0% with 1-class, 2-class, and 4-class settings, respectively. Our LGrad increases accuracy by 11.2%, 11.2%, and 13.1% on the same setting, respectively. Gradients are different from high-frequency information. This suggests that gradients can be used as an effective representation of artifacts that are capable of detecting GAN-generated images across diverse categories and various GAN models.

4.3.4 Robustness against Image Perturbations.

To evaluate the robustness of the proposed framework to image perturbations, we apply common image perturbations on the test images with a probability of 50% following [13]. These perturbations include blurring, cropping, compression, adding random noise, and a combination of all of them. In this subsection, the discriminator of StyleGAN-bedroom is used as the transformation model. The results are presented in Table 4. It can be observed that our detectors are robust with respect to cropping and blur, but obtain degraded performance on jpeg and noise. This could be due to the fact that the influence of jpeg and noise for the transformation model results in the transformed gradients that do not work well. We could further improve the robustness of the transformation model to tackle this problem.

4.4. Visualization

Figure 3 shows fake images generated by ProGAN and StyleGAN, real images sampled from CelebA-HQ, the gradients transformed from images, and Class Activate Map (CAM) [47] extracted from the detector. We adopt the discriminator of ProGAN-bedroom as the transformation model. It can be observed from gradients that most of the

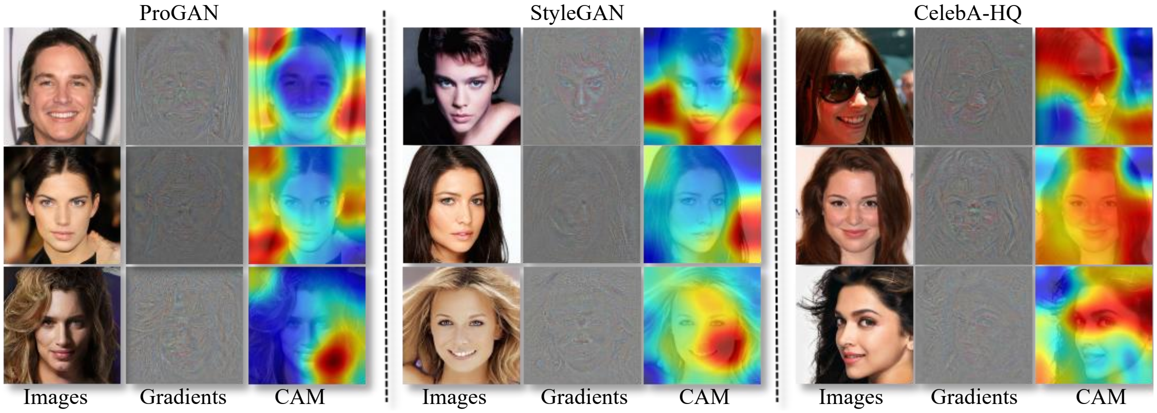

Figure 3. The visualization of gradients and Class Activate Map (CAM) [47] extracted from detector on face images.

content of images is filtered out, and the essential pixels for the transformation model are highlighted in the process of transforming. Although some texture information is preserved, its value is relatively low. Furthermore, the class activate map of the detector trained on gradients highlights the discriminative region for detection, and it can be seen that the maps of the real image highlight the face region, while most maps of the generated image are located in the background area. It reflects that the gradients, as the generalized representation of artifacts, contain high-level rather than low-level features such as texture.

5. Ablation Study

Effect of Transformation Model. In our framework, we turn the data-dependent problem into a transformation-model-dependent problem. The images are converted into the gradient, which serves as the generalized representation of artifacts. The performance of the detector highly relies on the quality of the transformation model. Now we discuss the influence of the transformation model on the detection performance, by employing ten pre-trained models. Table 5 shows the across-model & data results of different transformation models. We can observe that gradients generated by the discriminator of GAN perform better on unseen data compared to the classification model and contrastive learning model. In addition, discriminators with different structures or trained with different data exhibit varying performances. Thus, our framework turns the data-dependent problem into a transformation-model-dependent problem. In our experiment, the discriminator of StyleGAN-bedroom obtains the best performance.

Effect of the number of training data In order to test the influence of the number of training data for the detector, we sample 4,000, 18,000, and 36,000 images which contain equal numbers of real and fake images from the horse dataset as the training set, respectively. We also use 72k images of all the chairs and horses set to train the detector. In this subsection, the discriminator of StyleGAN-bedroom is also used as the transformation model. Table 6 compares the proposed framework with the different number of training data. Our detector achieves similar across-model & data performance with 18,000, 36,000, and 72,000 training images. In addition, the proposed LGrad with 4,000 training data still outperforms FrePGAN with 36,000 images, indicating the superiority of our method.

6. Conclusion

In this work, we propose a novel detection framework, called Learning on Gradients(LGrad), designed to identify GAN-generated images. In our method, the gradients are employed as a generalized representation to describe the artifacts produced by generation models. Specifically, we utilize a pre-trained CNN model as the transformation model to convert images into gradients effectively transforming the data-dependent problem into a transformation-model-dependent problem. Numerous experiments in our paper prove the great potential of the discriminator of GAN in detecting GAN-generated images. Moreover, we demonstrate the effectiveness and robustness of using gradients as generalized artifact representations in detecting GAN-generated images. Nonetheless, we acknowledge that the LGrad framework may not perform well in detecting non-GAN generative models, such as deepfake, and we aim to address this limitation in future research.

Acknowledgments This work is supported in part by the National Key R&D Program of China (No.2021ZD0112100), National NSF of China (No.61972022, No.U1936212, No.62120106009, No.62072394), Natural Science Foundation of Hebei province (F2021203019).

References

[1] Jawadul H Bappy, Cody Simons, Lakshmanan Nataraj, BS Manjunath, and Amit K Roy-Chowdhury. Hybrid lstm and encoder–decoder architecture for detection of image forgeries. *IEEE Transactions on Image Processing*, 28(7):3286–3300, 2019. 3

[2] Belhassen Bayar and Matthew C Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM workshop on information hiding and multimedia security*, pages 5–10, 2016. 2

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2018. 1, 4

[4] Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? understanding properties that generalize. In *European conference on computer vision*, pages 103–120. Springer, 2020. 2

[5] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 4, 7

[6] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018. 4

[7] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 2

[8] Samantha Cole. We are truly fucked: Everyone is making ai-generated fake porn now. *blog post*, 2018. Accessed April 17, 2020. 1

[9] Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*, 2018. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 4, 7

[11] Ricard Durall, Margret Keuper, and Janis Keuper. 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*, pages 7890–7899, 2020. 2, 3, 6

[12] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pages 3247–3258. PMLR, 2020. 2, 3, 6

[13] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pages 3247–3258. PMLR, 2020. 2, 7

[14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 1

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 7

[16] Yang He, Ning Yu, Margret Keuper, and Mario Fritz. 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, 3

[17] Nils Hulzebosch, Sarah Ibrahimi, and Marcel Worring. Detecting cnn-generated facial images in real-world scenarios. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 642–643, 2020. 1, 2

[18] Yonghyun Jeong, Doyeon Kim, Seungjai Min, Seongho Joe, Youngjune Gwon, and Jongwon Choi. Bihpf: Bilateral high-pass filters for robust deepfake detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 48–57, 2022. 2, 3, 4, 5, 6

[19] Yonghyun Jeong, Doyeon Kim, Youngmin Ro, and Jongwon Choi. Frepgan: Robust deepfake detection using frequency-level perturbations. *arXiv preprint arXiv:2202.03347*, 2022. 2, 3, 4, 5, 6, 7

[20] Yonghyun Jeong, Doyeon Kim, Youngmin Ro, Pyounggeon Kim, and Jongwon Choi. Fingerprintnet: Synthesized fingerprints for generated image detection. In *European Conference on Computer Vision*, pages 76–94. Springer, 2022. 6

[21] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 1, 2, 4, 5, 7

[22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 1, 2, 4, 5, 7

[23] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 1, 2, 4, 5, 7

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015. 4

[25] Diederik P Kingma and Max Welling. Auto-encoding variational {Bayes}. In *Int. Conf. on Learning Representations*. 1

[26] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5001–5010, 2020. 2

[27] Yuezun Li, Ming-Ching Chang, and Siwei Lyu. 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

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4

[29] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 4

[30] Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8060–8069, 2020. 2

[31] Francesco Marra, Diego Gragnaniello, Luisa Verdoliva, and Giovanni Poggi. Do gans leave artificial fingerprints? In *2019 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 506–511. IEEE, 2019. 2, 3

[32] Iacopo Masi, Aditya Killekar, Royston Marian Mascarenhas, Shenoy Pratik Gurudatt, and Wael AbdAlmageed. Two-branch recurrent network for isolating deepfakes in videos. In *European conference on computer vision*, pages 667–684. Springer, 2020. 3

[33] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 4

[34] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European conference on computer vision*, pages 86–103. Springer, 2020. 3

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 4, 5, 7

[36] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 1

[37] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1–11, 2019. 2, 4

[38] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 4

[39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, 2015. 4, 5, 7

[40] Ke Sun, Hong Liu, Qixiang Ye, Yue Gao, Jianzhuang Liu, Ling Shao, and Rongrong Ji. Domain general face forgery detection by learning to weight. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 2638–2646, 2021. 2

[41] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 4, 7

[42] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8695–8704, 2020. 2, 4, 5, 6

[43] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 4, 5

[44] Ning Yu, Larry S Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7556–7566, 2019. 2

[45] Yang Yu, Rongrong Ni, and Yao Zhao. Mining generalized features for detecting ai-manipulated fake faces. *arXiv preprint arXiv:2010.14129*, 2020. 2

[46] Tianchen Zhao, Xiang Xu, Mingze Xu, Hui Ding, Yuanjun Xiong, and Wei Xia. Learning self-consistency for deepfake detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15023–15033, 2021. 2

[47] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 7, 8

[48] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *European conference on computer vision*, pages 592–608. Springer, 2020. 4, 7

[49] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 4