From Detection to Deception: Are AI-Generated Image Detectors Adversarially Robust?

Yun-Yun Tsai, Ruize Xu, Chengzhi Mao, Junfeng Yang
Columbia University
{yt2781, rx2246}@columbia.edu {mcz,junfeng}@cs.columbia.edu

Abstract

Generative models are revolutionizing industries by synthesizing high-quality images, yet they pose societal risks as they are exploited at scale for generating disinformation, propaganda, scams, and phishing attacks. Recent work has developed detectors with remarkable accuracy in identifying images generated by current models, but the robustness of the detectors remains to be explored. This paper investigates the robustness of these detectors against adversarial perturbations designed to elude detection. We observe that an end-to-end adversarial attack on the entire detection pipeline is ineffective due to the long stochastic process of diffusion models. Instead, we create intermediate guidance for the attack at the model internals. Empirical results on both black box and white box attacks demonstrate the importance of our proposed intermediate supervision when constructing the attack. Our results show that our approach can fool the detector, reducing the detection accuracy by up to 69 points in the black-box setting and 91 to 100 points in the white-box setting. In addition, our attack transfers well to generated images from unknown models, including StyleGAN. Our work suggests that existing AI-generated image detectors are easily deceived by adversarial perturbations, highlighting the need for more robust detectors.

1. Introduction

Generative model-based tools such as Stable Diffusion [21], DALL-E [7], Midjourney [1], and Runway [2] are revolutionizing industries from design, entertainment, and education to marketing by synthesizing high-quality images [18, 23]. However, these models also pose societal risks as they are exploited at scale for generating disinformation, propaganda, scams, and phishing attacks [20], highlighted by instances like the Taylor Swift deepfakes [3].

Addressing this threat, recent advancements have developed detectors that demonstrate remarkable accuracy in identifying images generated by current models [16, 26]. The core methodology involves computing the Diffusion Reconstruction Error (DIRE), which quantifies the discrepancy between an input image and its reconstruction via a pre-trained diffusion model, followed by employing a classifier trained on these DIRE values to distinguish between AI-generated and human-produced images.

Unfortunately, the robustness and reliability of existing detectors remain to be explored. In an adversarial setting, the input images to these detectors may be manipulated by the attackers, causing the detectors to misclassify AI-generated images as natural ones, and vice versa. In this paper, we investigate the robustness of state-of-the-art AI-generated image detection and construct two novel attacks, termed GIDAttack, to substantially decrease the effectiveness of detection. We show that prior gradient-based adversarial attack methods against CNNs [5, 8, 17] are not effective when attacking the entire diffusion and classification pipeline in current detectors, because the long stochastic process in diffusion and classification leads to exploding or vanishing gradients in the optimization process. Our key idea to solve this problem is to add intermediate supervision to guide the gradient-based optimization for attack generation. Figure 1 and 2 show the flow of GIDAttack in black-box and white-box settings. Visualizations and empirical results show that our proposed attack can successfully mislead State-of-the-Art diffusion-generated image detectors.

2. Related Works

Generated Image Detection The greater success of generating high-quality images has raised concern about digital integrity and drawn attention to constructing a robust detector to detect whether the sample is from generative models or not. Lorenz et al. [15] modified the widely-used Local Intrinsic Dimensionality (LID) to detect diffusion-based images. Based on the observation that generated images vary less than real images after reconstruction, DIRE [26] proposed to utilize reconstructed error to detect diffusion-generated images and was proved better than traditional RGB-based detectors. SeDID [16] extended it by ensembling step-wise noise error of corresponding inverse and reverse stages. To facilitate diffusion image detection, benchmarks at image

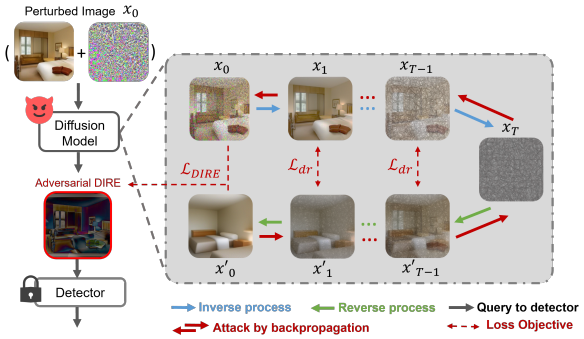

Figure 1. The flow of Black-box GIDAttack. We assume the detector can not be accessed during the attack process. Given an input image x, we perturb the sample by adding the noise δ. We attack by maximizing the loss objective for AI-generated images and minimizing it for real images. We extract sequences of samples from both inverse (blue arrow) and reverse processes (green arrow) at the intermediate diffusion time step to calculate the deviated reconstruction loss $\mathcal{L}_{dr}$. After our attack, we infer via the detector $\mathcal{F}$ with the adversarial image and obtain the output prediction.

level [26] and fine-grained region level [25] were established for a fair comparison. Despite the preliminary work, the adversarial robustness of diffusion detectors remains untouched.

Adversarial Robustness of Diffusion Models Recently, the diffusion model has shown outstanding success in detecting and purifying the adversarial sample [4, 19, 24, 27, 29]. Diffpure [19] purifies the adversarial samples with a diffusion model by solving the stochastic differential equation (SDE) and calculating the gradient during the reverse process. DIFFender [10] leverages a text-guided diffusion model to reconstruct the adversarial regions in the images for patch attack. Liu et al., [14] paints the out-of-distribution (OOD) samples with diffusion models and shows the mapped images of OOD would have a large distance away from its original manifold. Although a large number of works show the diffusion models can significantly improve the adversarial robustness, DiffAttack [11] specifically attacks against diffusion-based purification and demonstrate its effectiveness, which motivates us to study the adversarial robustness of diffusion-based image detectors.

3. Method

Preliminaries • **Denoising Diffusion Probabilistic Models (DDPM)** is first proposed in [22] inspired by nonequilibrium thermodynamics, strengthening the image quality during the generation process. The whole DDPM includes a forward process and a reverse process.

Given an image x_0 sampled from $\mathcal{X}_S$, DDPM first gradually adds Gaussian noise to the data point x_0 through a fixed Markov Chain during the forward process for T steps. Specifically, for every timestep $t \in [1, ..., T]$, they sample

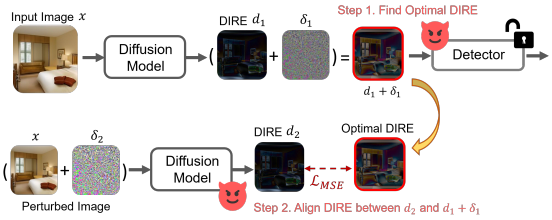

Figure 2. The flow of White-box GIDAttack. We assume the adversary can access the detector during the attack process. Our attack includes two main steps. (1.) Find an optimal DIRE: Given an input image x, we first obtain the DIRE representation d_1 of x from the offline diffusion model. We directly add a noise δ_1 on DIRE representation d_1 and optimize δ_1 with the output loss from the detector. After T attack iteration, we can find an optimal DIRE that fools the detector. (2.) Align DIRE representation: We start to attack the input image x, adding another noise δ_2 on x and getting the DIRE d_2 of $x + \delta_2$. We optimize δ_2 by aligning the DIRE d_2 with the optimal DIRE obtained from step 1.

data sequence $[x_1, ..., x_T]$ by adding Gaussian noise with variance $\beta_t \in (0, 1)$ during the forward process, which is defined as:

$$q(x_t|x_0) = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon\,, \tag{1}$$

where $\epsilon \sim \mathcal{N}(0, 1)$ is the noise sampeled from Gaussian distribution, $\alpha_t = 1 - \beta_t$, and $\bar{\alpha}_t = \Pi_{s=1}^{t}\alpha_s$. The reverse process then generates a sequence of denoised image $[x_t^g, x_{t-1}^g ..., x_0^g]$ from timestep $t \in [T, ...1]$. For timestep t in the reverse process, the denoised image can be defined as:

$$x_{t-1}^g = \frac{1}{\sqrt{\alpha_t}}\left(x_t^g - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t^g, t)\right) + \sigma_t\epsilon\,, \tag{2}$$

where ϵ_θ is a trainable noise predictor that generates a prediction for the noise at the current timestep and removes the noise. σ_t is the variance of noise. Ideally, the generated sample x_0^g should be moved forward to the distribution of the source domain trained for the diffusion model.

• **Diffusion Reconstruction Error** Prior research has shown that existing generated-image detectors for GAN or varational autoencoder (VAE) suffer from performance drops for detecting images generated from the diffusion model. In [26], they proposed the diffusion reconstruction error (DIRE), which can well capture the signal of the diffusion-generated image. Given an image x_0 drawn from any $\mathcal{X}$ and a pre-trained diffusion model, the DIRE is calculated by

$$DIRE(x_0) = |x_0 - \mathbf{R}(\mathbf{I}(x_0))| \tag{3}$$

where $\mathbf{I}$ represents the inversion process and $\mathbf{R}$ represents the reconstruction process. The output of DIRE is calculated by the absolute difference between x_0 and its corresponding reconstructed version.

• **AI-generated Image Detector** To build a robust detector for detecting the AI-generated images, instead of training

a binary classifier with real and generated images, [26] use a binary DNN classifier trained with corresponding DIRE representation for real and generated images. For generated samples, they apply multiple sources of diffusion models (e.g., DDPM, PNDM, ADM, iDDPM,...,etc.) to extract the DIRE representation. The model architecture is ResNet50.

Black-box GIDAttack We assume the adversary can not access the target detector to get the output prediction during the attack process, where the detector is a fully black-box model for the adversary. The attacker can access an offline diffusion model to generate the DIRE representation and run the attack algorithm. We assume this offline diffusion model does not need to be the same diffusion model that provides the generated image for training the target detector.

• **Problem Formulation** We focus on attacking the input such that the diffusion process can generate the adversarial DIRE to fool the detector $\mathcal{F}$. Suppose we have a diffusion model that consists of the inverse process $\mathbf{I}$ and reverse process $\mathbf{R}$. The attack formulation is defined as:

$$\mathcal{L}\left(\mathcal{F}\left(|x_{adv} - \mathbf{R}(\mathbf{I}(x_{adv}))|, y\right)\right) \quad s.t. \quad \delta \leq \epsilon, \tag{4}$$

where the objective function $\mathcal{L}(\cdot)$ can be selected as classifier-guided loss, such as binary cross-entropy. However, under the black-box setting, it is unrealistic to query the detector $\mathcal{F}$ and get the output prediction. For example, the detector is deployed as an API in the cloud services and has been rate-limited for the query number. Therefore, our attack calculates the distance between x_{adv} and $\mathbf{R}(\mathbf{I}(x_{adv}))$. The objective function for the DIRE loss attack is defined as:

$$\mathcal{L}_{dire} = d(x_{adv}, \mathbf{R}(\mathbf{I}(x_{adv}))) \quad s.t. \quad \delta \leq \epsilon, \tag{5}$$

where the DIRE loss $\mathcal{L}_{DIRE}$ is computed by the distance function d, the absolute distance between x_{adv} and its reconstructed version. However, the distance loss can only apply to the output of the diffusion process, thus inducing the gradient exploding/vanishing problem in the long diffusion process. To tackle this challenge, we incorporate the deviated reconstruction loss with $\mathcal{L}_{DIRE}$ to provide intermediate guidance during the attack.

• **Deviated Reconstruction Loss** The deviated reconstruction loss (DRL) is first proposed in [11], where they attack the adversarial purification process of the diffusion model. In [11], the DRL aims to force discrepancy for the samples extracted from intermediate steps of the inverse and reverse processes. Specifically, a sequence of noisy samples in timestep $t \in [1...T]$ for x_0 is $x_1, ..., x_T$ during the inverse process $\mathbf{I}$ and another sequence of denoised samples $x'_{T-1}, ...x'_1$ can be sampled from timestep $t \in [T-1...1]$ during the reverse process. The deviated reconstruction loss can be calculated by the image pairs (x_t, x'_t) sampled from timesteps t, which is defined as:

$$\mathcal{L}_{dr} = \mathbb{E}_{(x_t, x'_t)|x_0}[\alpha_t d(x_t, x'_t)], \tag{6}$$

where α_t is the weight coefficient at timestep t and d is the ℓ_2 distance function $\|\cdot\|_2$ between x_t and x'_t. The deviated reconstruction loss $\mathcal{L}_{dr}$ is computed by averaging the results of d at the uniformly sampled timesteps in $[1, T]$. The α_t is set as the reciprocal of sampled timestep (e.g., if t is 5, then the $\alpha_{t=5}$ is $1/5$). By adding the deviated reconstruction loss with the DIRE loss $\mathcal{L}_{DIRE}$, we can better control the optimization process and tackle the problem of gradient vanishing/exploding. The objective function for query-free attack for finding the optimal x_{adv} to generate adversarial DIRE representation is defined as:

$$\mathcal{L}(x; \delta) = \lambda_1 \mathcal{L}_{dr}(\cdot) + \lambda_2 \mathcal{L}_{dire}(\cdot), \tag{7}$$

We extend the iterative gradient sign method to optimize the perturbations δ for Q attack iterations, defined as:

$$\delta^{q+1} = \text{clip}_{\epsilon}(\delta^q + \beta \cdot \nabla_{\delta}(\mathcal{L}(x; \delta^q))), \tag{8}$$

where $q \in [0, Q]$ is the current attack timestep, β is the step size of each iteration, $\nabla_{\delta}\mathcal{L}(\cdot)$ is the gradient of a loss function $\mathcal{L}$ with respect to the perturbation variable δ^q. The $\text{clip}_{\epsilon}(\cdot)$ function constraint all values in δ under a range $\pm\epsilon$.

White-box GIDAttack Assume the attacker can access the detector and retrieve the output information (e.g., probability score from $\mathcal{F}$), one can easily perturb the input and end-to-end generate the wrong DIRE representation from offline diffusion model via the optimization process in Eq. 4 with classifier-guided loss (e.g., binary cross entropy).

However, We observe that the end-to-end attack suffers from the gradient vanishing/exploding problem caused by the diffusion model. To solve the unstable problem of end-to-end attack, we propose a novel two-step white-box GIDAttack, which can improve the end-to-end attack efficiency.

• **Problem Formulation** Given an input pair (x_0, y) without attack, we denote the DIRE representation of x_0 as d_0, where $d_0 = |x_0 - \mathbf{R}(\mathbf{I}(x_0))|$. Our attack includes two steps: (1.) Find intermediate perturbation for attack: we directly add a noise vector δ_1 on d_0 and optimize it by the output loss from detector $\mathcal{F}$. We define the objective function for finding the adversarial DIRE $d'_0 = d_0 + \delta_1$ as follows:

$$\max \mathcal{L}_{bce}(\mathcal{F}(d_0 + \delta_1), y) \quad s.t. \quad \delta_1 \leq \epsilon_d, \tag{9}$$

where the loss function $\mathcal{L}_{bce}$ is binary cross entropy. (2.) Align the DIRE representation for attack: after obtaining the adversarial DIRE d'_0, we reversely find the perturbation δ_2 for x_0, where the DIRE representation extracted from $x_0 + \delta_2$ should be approximately closed to adversarial DIRE d'_0. We denote the DIRE representation generated from $x_0 + \delta_2$ as u_0 The attack process can be defined as:

$$\min \mathcal{L}_{mse}(u_0, d'_0) \quad s.t. \quad \delta \leq \epsilon, \tag{10}$$

where the loss function $\mathcal{L}_{mse}$ is the mean square error calculated between the adversarial DIRE d'_0 and the DIRE u_0 generated from adversarial image $x_0 + \delta_2$.

Detector	Method	Sources of Testing Set								Total
		ADM	iDDPM	PNDM	DDPM	SD-v1	LDM	StyleGAN	Dalle-2	Avg.
ADM	w/o Attack	100	100	99	100	99	100	100	100	99
	DIRE Attack	66	100	99.3	98	99	99	93	99	94.1
	GIDAttack	**65**	**95**	**99**	**95**	**96**	**98**	**87**	**95**	**91**
iDDPM	w/o Attack	99	100	98	100	99	99.9	100	100	99.6
	DIRE Attack	95	71	89	95.3	65	76	89	73	81.7
	GIDAttack	**92**	**31**	**79**	**90**	**61**	**75**	**88**	**61**	**72**
PNDM	w/o Attack	99	100	100	100	100	100	98	100	99
	DIRE Attack	96	95	80	100	97	100	94	100	95
	GIDAttack	**94**	**94**	**54**	**99**	**92**	**93**	**94**	**90**	**88**
Style-GAN	w/o Attack	100	100	100	43	57	85	98	96	84.8
	DIRE Attack	**93**	93.7	94.3	16	26	53	86.7	80	67.8
	GIDAttack	**91**	**92.3**	**91.7**	**14**	**25**	**51**	**85**	**77**	**66.5**

Table 1. Performance of Black-box GIDAttack on diffusion detectors trained from four sources of generated images, including iDDPM, ADM, PNDM, and StyleGAN. We apply GIDAttack to 8 kinds of generated samples and the real sample from the original LSUN-bedroom.

Detector	Method	Sources of Testing Set				Total
		ADM	iDDPM	Dalle-2	StyleGAN	Avg.
ADM	w/o Attack	100	100	100	100	100
	E2E Attack	29	67	20	44	40
	GIDAttack	**0**	**3**	**1**	**0**	**1**
iDDPM	w/o Attack	99.6	100	100	100	99.9
	E2E Attack	10	43	32	29	28.9
	GIDAttack	**6**	**8**	**4**	**7**	**6.3**
PNDM	w/o Attack	99	100	100	98	99.3
	E2E Attack	30	52	44	60	46.5
	GIDAttack	**9**	**4**	**2**	**1.5**	**4.1**

Table 2. Performance of White-box GIDAttack diffusion detectors. We evaluate our method on five sources of testing sets and report the robust accuracy on w/o attack, end-to-end attack, and GIDAttack. The number in bold shows the best results.

4. Experiment

• **Dataset and Data Preprocess** We evaluate GIDAttack on the DiffusionForensic [26] benchmark. The DiffusionForensic dataset includes samples generated from multiple sources of generative models (e.g., ADM [6], iDDPM [18], PNDM [13], Dalle-2 [7], StyleGAN [12], ... etc.), which are trained on LSUN-Bedroom [28] dataset. The size of every generated sample is 256*256 and will be resized to 224*244 during testing. • **Model** Our offline reconstruction model is an unconditional model pre-trained on LSUN-Bedroom [28] dataset. We apply DDIM [22] strategy in the inverse and reverse process and set up the sampling steps as 20. We evaluate our attack on four detectors, trained with real and fake images of the LSUN-bedroom dataset. The real images are from the test set of LSUN-bedroom, and the fake images are from four different sources of generated models, including ADM, iDDPM, PNDM, and StyleGAN. Our detector is a ResNet50 [9] model. • **Implementation Details** Our GIDAttack applies the projected gradient descent (PGD) optimization on both white-box and black-box settings. For the black-box setting, we iteratively update the perturbation added on input by maximizing the $\mathcal{L}_{dire}$ and $\mathcal{L}_{dr}$ loss with the attack iteration as 20. We set up the attack epsilon ϵ from [8/255 to 64/255]. We show the results with optimal epsilon $16/255$. • **Baseline Details** (1.) w/o Attack: The samples are directly input through the offline diffusion model to obtain the DIRE representation and then tested on the detectors. (2.) DIRE Attack: The black-box attack baseline updates the perturbation by optimizing the ℓ_{dire} loss only. (3.) End-to-end attack: The white-box attack baseline that updates the perturbation by optimizing the loss calculated from the output of detectors. • **Experimental Results** In Table 1, We evaluate our GIDattack on four black-box detectors, which are trained with different sources of generated samples, including ADM, iDDPM, PNDM, and StyleGAN. Our testing sets are from 8 sources of different generated models (e.g., ADM, LDM, DDPM, ..., Dalle-2). Compared with the method '*DIRE Attack*' and '*w/o Attack*', the results after GIDAttack have a larger improvement, and the accuracy drops by 8.67% to 27.32%. For four detectors, GIDAttack outperforms the DIRE Attack when the source of testing sets are the same as detector (e.g., ADM (35%), iDDPM (69%), PNDM (46%), and StyleGAN (13%)). For unseen testing sets (e.g., SD-v1, DDPM, LDM, and Dalle-2), GIDAttack outperforms DIRE Attack by 2% to 5.75% on and w/o attack by 3.95% to 28.5% on average. In Table 2, we show the performance of GIDAttack in the white-box setting, where the attacker can access the detector during the attack process. We compare GIDAttack with the end-to-end attack. The attacks are applied to four sources of generated samples. The attack iteration is set as 20, and the epsilon ϵ is set as 16/255. Compared with *w/o Attack*, the end-to-end attack achieves 33% to 80% attack accuracy on ADM detector, 38% to 69% on PNDM detector, and 57% to 90% on iDDPM detector. For our white-box GIDAttack, we can achieve nearly 91% to 100% attack accuracy on all three detectors and improve the attack performance of end-to-end attacks by 4% to 64%.

5. Conclusion

GIDAttack is a novel approach to attack the generated-image detectors The GIDAttack combines the dire and deviated reconstruction loss calculated by the samples from intermediate diffusion steps, which can better control the learning of perturbation during the diffusion process. We demonstrate that the GIDAttack outperforms the baselines on SOTA detectors. Our work suggests that existing AI-generated image detectors can be deceived easily by adversarial perturbations, highlighting the need for more robust detectors.

References

[1] Midjourney. (2023). midjourney (v5.1) [text-to-image model]. https://www.midjourney.com/. 1

[2] Runway. (2024) [text-to-image model]. https://research.runwayml.com/. 1

[3] Look what you made me do: Why deepfake taylor swift matters, 2024. 1

[4] Tsachi Blau, Roy Ganz, Bahjat Kawar, Alex Bronstein, and Michael Elad. Threat model-agnostic adversarial defense using diffusion models. *arXiv preprint arXiv:2207.08089*, 2022. 2

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, pages 39–57, 2017. 1

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 4

[7] Aditya Ramesh et al. Hierarchical text-conditional image generation with clip latents, 2022. 1, 4

[8] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations*, 2015. 1

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 4

[10] Caixin Kang, Yinpeng Dong, Zhengyi Wang, Shouwei Ruan, Yubo Chen, Hang Su, and Xingxing Wei. Diffender: Diffusion-based adversarial defense against patch attacks, 2023. 2

[11] Mintong Kang, Dawn Xiaodong Song, and Bo Li. Diffattack: Evasion attacks against diffusion-based adversarial purification. *ArXiv*, abs/2311.16124, 2023. 2, 3

[12] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018. 4

[13] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 4

[14] Zhenzhen Liu, Jin Peng Zhou, Yufan Wang, and Kilian Q Weinberger. Unsupervised out-of-distribution detection with diffusion inpainting. *arXiv preprint arXiv:2302.10326*, 2023. 2

[15] Peter Lorenz, Ricard L. Durall, and Janis Keuper. Detecting images generated by deep diffusion models using their local intrinsic dimensionality. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 448–459, 2023. 1

[16] RuiPeng Ma, Jinhao Duan, Fei Kong, Xiaoshuang Shi, and Kaidi Xu. Exposing the fake: Effective diffusion-generated images detection. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023. 1

[17] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. 1

[18] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 1, 4

[19] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning (ICML)*, 2022. 2

[20] Jonas Ricker, Simon Damm, Thorsten Holz, and Asja Fischer. Towards the detection of diffusion model deepfakes. *arXiv preprint arXiv:2210.14571*, 2022. 1

[21] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1

[22] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 2, 4

[23] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 1

[24] Jiachen Sun, Weili Nie, Zhiding Yu, Z. Morley Mao, and Chaowei Xiao. Pointdp: Diffusion-driven purification against adversarial attacks on 3d point cloud recognition, 2022. 2

[25] Sai Wang, Ye Zhu, Ruoyu Wang, Amaya Dharmasiri, Olga Russakovsky, and Yu Wu. Deter: Detecting edited regions for deterring generative manipulations, 2023. 2

[26] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. *arXiv preprint arXiv:2303.09295*, 2023. 1, 2, 3, 4

[27] Chaowei Xiao, Zhongzhu Chen, Kun Jin, Jiongxiao Wang, Weili Nie, Mingyan Liu, Anima Anandkumar, Bo Li, and Dawn Song. Densepure: Understanding diffusion models towards adversarial robustness, 2022. 2

[28] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *CoRR*, abs/1506.03365, 2015. 4

[29] Kui Zhang, Hang Zhou, Jie Zhang, Qidong Huang, Weiming Zhang, and Nenghai Yu. Ada3diff: Defending against 3d adversarial point clouds via adaptive diffusion, 2023. 2