

Detecting Adversarial Data Using Perturbation Forgery

Qian Wang¹, Chen Li¹, Yuchen Luo², Hefei Ling^{1*}, Shijuan Huang¹, Ruoxi Jia³, Ning Yu⁴

¹Huazhong University of Science and Technology, China ²Wuhan University, China

³Virginia Tech, USA ⁴Netflix Eycline, USA

{yqwq1996, lhefei, llichen, shijuan_huang}@hust.edu.cn, {yuchenluo}@whu.edu.cn,

ruoxijia@vt.edu, ningyu.hust@gmail.com

Abstract

As a defense strategy against adversarial attacks, adversarial detection aims to identify and filter out adversarial data from the data flow based on discrepancies in distribution and noise patterns between natural and adversarial data. Although previous detection methods achieve high performance in detecting gradient-based adversarial attacks, new attacks based on generative models with imbalanced and anisotropic noise patterns evade detection. Even worse, the significant inference time overhead and limited performance against unseen attacks make existing techniques impractical for real-world use. In this paper, we explore the proximity relationship among adversarial noise distributions and demonstrate the existence of an open covering for these distributions. By training on the open covering of adversarial noise distributions, a detector with strong generalization performance against various types of unseen attacks can be developed. Based on this insight, we heuristically propose Perturbation Forgery, which includes noise distribution perturbation, sparse mask generation, and pseudo-adversarial data production, to train an adversarial detector capable of detecting any unseen gradient-based, generative-based, and physical adversarial attacks. Comprehensive experiments conducted on multiple general and facial datasets, with a wide spectrum of attacks, validate the strong generalization of our method.¹

1. Introduction

Numerous studies have demonstrated the effectiveness of deep neural networks (DNNs) in various tasks [36]. However, it is also well-known that DNNs are vulnerable to adversarial attacks, which generate adversarial data by adding imperceptible perturbations to natural images, potentially causing the model to make abnormal predictions. This vul-

nerability reduces the reliability of deep learning systems, highlighting the urgent need for defense techniques. Some approaches employ adversarial training techniques [48], which incorporate adversarial data into the training process to inherently bolster the model’s immunity against adversarial noise (adv-noise). Other methods utilize adversarial purification [47] to cleanse the input data of adv-noise before model inference. Unfortunately, adversarial training often results in diminished classification accuracy on adversarial data and may sacrifice performance on natural data [1]. While purification techniques may seem more reliable, the denoising process can inadvertently smooth the high-frequency texture of clean images, leading to suboptimal accuracy on both natural and adversarial images.

Another branch of adversarial defense is adversarial detection [12, 28], which aims to filter out adversarial data before they are fed into target systems. These methods rely on discerning the discrepancies between adversarial and natural distributions, thereby maintaining the integrity of target systems while achieving high performance in defending against adversarial attacks. However, existing adversarial detection approaches primarily train specialized detectors tailored for specific attacks or classifiers [8], which may not generalize well to unseen advanced attacks [2, 39].

Some recent approaches [40, 49] have attempted to enhance the generalization of adversarial detectors by leveraging diffusion models and manually designed pseudo-noise. Nevertheless, as shown in Table 1, this improvement often comes at the expense of increased inference overhead, and even well-trained detectors still struggle with adversarial data produced by generative models such as generative adversarial networks (GANs) and diffusion models. Specifically, generative-based adversarial attacks, such as M3D [51] and Diff-PGD [44], are recently developed techniques designed to generate more natural adversarial data. Unlike traditional gradient-based attacks like PGD [29], generative-based attacks tend to create imbalanced and anisotropic perturbations that are intense in high-frequency and salient areas but mild in low-frequency and background

*Corresponding author

¹Code at <https://github.com/cc13qq/PFD>

Table 1. **Properties of our method vs Other Adversarial Detection:** We summarize four metrics for adversarial detection. “Detect Attacks” specifies the types of attacks each method is designed to detect. “Model-Agnostic” indicates whether the detection method operates independently of the protected model. “Unseen-Attack Detection” assesses the method’s generalization to detect previously unseen attacks, with the reported value representing the AUROC score for detecting the MIFGSM attack on the ImageNet dataset. Lastly, “Time Overhead” quantifies the time required for the detector to process 100 samples from the ImageNet dataset.

Methods	Detect Attacks	Model-Agnostic	Unseen-Attack Detection	Time Overhead
LID [28]	Gradient	×	0.9146	1.80s
LiBRe [8]	Gradient	×	0.8725	2.56s
SPAD [40]	Gradient + GAN	✓	0.9820	4.56s
EPSAD [49]	Gradient	✓	0.9918	396.81s
Ours	Gradient + GAN + Diffusion	✓	0.9931	4.85

areas. These perturbations make them harder to detect with current methods, posing a new challenge to the field of adversarial detection. Therefore, there is an urgent need for a low-time-cost detection method with strong generalization performance against both unseen gradient-based attacks and unseen generative-based attacks.

In this paper, we investigate the association between the distributions of different adv-noises. As shown in Figure 1, by modeling noise as multivariate Gaussian distributions, we define the proximity relationship of the noise distributions in a metric space and establish a perturbation method to construct the proximal distributions. Based on the assumption and definition, we demonstrate that all adv-noise distributions are proximal using the Wasserstein distance, and we deduce a corollary that an open covering of the adv-noise distributions exists. We then propose the core idea of this paper: **by training on the open covering, we obtain a detector with strong generalization performance against various types of unseen attacks.**

Based on this idea, we propose Perturbation Forgery, a pseudo-adversarial data (pseudo-adv-data) generation method designed to train a detector with strong generalization capabilities for detecting various types of adversarial attacks. As shown in Figure 2, Perturbation Forgery comprises noise distribution perturbation, sparse mask generation, and pseudo-adversarial data (pseudo-adv-data) production. By continuously perturbing the noise distribution of a commonly used attack, such as FGSM [15], we obtain a nearly complete family of distributions that form the aforementioned open covering. Simultaneously, sparse masks are generated using attention maps and saliency maps of the natural data, converting the sampled global noises into local forms to help the detector focus on local noise patterns so as to detect generative-based adv-noise. By sampling noise from the perturbed noise distributions, we sparsify it using sparse masks and inject it into natural data. This process converts half of the natural data into pseudo-adv-data, replacing actual adversarial data. The pseudo-adv-data is then combined with the remaining uncontaminated data to train

the binary adversarial detector.

To the best of our knowledge, we are the first to empower a detector with the robust ability to detect various types of adversarial attacks, including gradient-based, GAN-based, diffusion-based, and physical attacks, with minor inference time overhead.

Our contributions are summarized in four thrusts:

- By modeling adversarial noise as Gaussian distributions, we investigate the association between different adversarial noise distributions and demonstrate their proximity in a metric space.
- Based on our assumption and definition, we deduce the existence of an open covering for adversarial noise distributions and propose training a detector to distinguish this open covering from the natural data distribution, thereby achieving strong generalization in detection.
- Building on our deductions, we propose Perturbation Forgery, which includes noise distribution perturbation, sparse mask generation, and pseudo-adversarial data production. This method is designed to train a binary detector with strong generalization performance against various types of unseen attacks, including gradient-based, GAN-based, diffusion-based, and physical adversarial data, with minor inference time overhead.
- We conduct evaluations on multiple general and facial datasets across a wide spectrum of gradient-based, generative-based, and physical attacks, demonstrating the consistently strong performance of our method in adversarial detection tasks.

2. Related Work

2.1. Adversarial Attack

Gradient-based adversarial attack Adversarial attacks [15] attempt to fool a classifier into changing prediction results by injecting subtle perturbations to the input data while maintaining imperceptibility from human eyes. By introducing iterative perturbation, PGD [29] and BIM [24] generate adversarial data with significantly im-

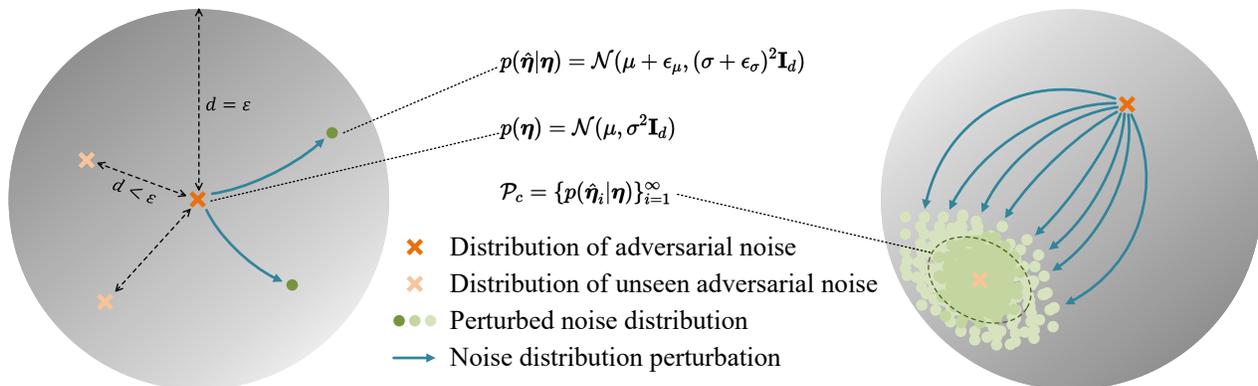


Figure 1. Illustration of the proximity of adversarial noise distributions and perturbed noise distributions. Left: all the distributions of adversarial noise and the perturbed distributions are in a ϵ -ball centered on a given adversarial noise distribution. Right: by continuously perturbing the given adversarial noise distribution, we obtain an open covering of distributions of adversarial noise.

proved attack performance, while DIM [43], NIM [26], and MIM [9] improve the transferability under the black-box setting. Besides, adaptive attacks [6, 13, 18] are proposed to exploit the weak points of each defense, achieving a high attack rate against several defense techniques.

Generative-based Adversarial Attacks Recently, another branch of attacks uses generative models to produce adversarial data with a more natural style and imbalanced noise compared to gradient-based attacks. Among them, CDA [31], TTP [30], and M3D [51] utilize GAN to craft adversarial data and achieve higher efficiency when attacking large-scale datasets. Diff-Attack [4] and Diff-PGD [44] employ the diffusion model as a strong prior to better ensure the realism of generated data. Designed for perturbing facial recognition systems, AdvMakeup [46] and AMT-GAN [19] are presented to generate imperceptible adversarial images of the face.

In this paper, a wide spectrum of gradient-based, generative-based, and physical adversarial attacks on ImageNet and facial datasets is used to evaluate the detection performance of our method.

2.2. Adversarial Detection

As one of the technical solutions to ensure the safety of DNNs, there has been a significant amount of exploration in the field of adversarial detection [28].

LiBRe [8] leverages Bayesian neural networks to endow pre-trained DNNs with the ability to defend against unseen adversarial attacks, SPAD [40] leverages a data augmentation method to detect adversarial face images from unseen attack, while EPSAD [49] incorporates a diffusion model to detect adversarial data with minimal attack intensity, albeit at the cost of inference time.

While current methods have achieved remarkable improvements in adversarial detection, generative-based ad-

versarial attacks are harder to detect, posing a new challenge. To address the problems above, our method is proposed to detect various types of unseen attacks, including gradient-based, generative-based, and physical attacks.

3. Proximity of Adversarial Noise Distribution

In this section, we provide an overview of preliminary concepts related to adversarial data generation and explore the relationship between adversarial noise distributions. We define the proximity of noise distributions and present the core idea of this paper: by training on the open covering of adversarial noise distributions, we obtain a detector with strong generalization performance against various types of unseen attacks.

3.1. Adversarial data generation

Given a DNN f that works on dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $(\mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{C})$ being (sample, ground-truth) pair, an adversarial example $\hat{\mathbf{x}}$ regarding sample \mathbf{x} with attack intensity ϵ is calculated as:

$$\hat{\mathbf{x}} = \underset{\hat{\mathbf{x}} \in \mathcal{B}(\mathbf{x}, \epsilon)}{\operatorname{argmax}} \mathcal{L}(f(\hat{\mathbf{x}}), y), \quad (1)$$

where \mathcal{L} is some loss function and $\mathcal{B}(\mathbf{x}, \epsilon) = \{\mathbf{x}' | \ell^p(\mathbf{x}, \mathbf{x}') < \epsilon\}$ restricts the perturbation under ℓ^p norm. Following [15], we simply denote adversarial sample as $\hat{\mathbf{x}} = \mathbf{x} + \boldsymbol{\eta}$.

3.2. Proximity of noise distribution

Due to the subtle and imperceptible nature of adv-noise, adversarial data and natural data appear visually similar. Thus, we assume that this similarity also exists between the noises of different adversarial attacks. To investigate the similarity of these noises, we provide the following definition, which depicts a close relationship between two noise distributions.

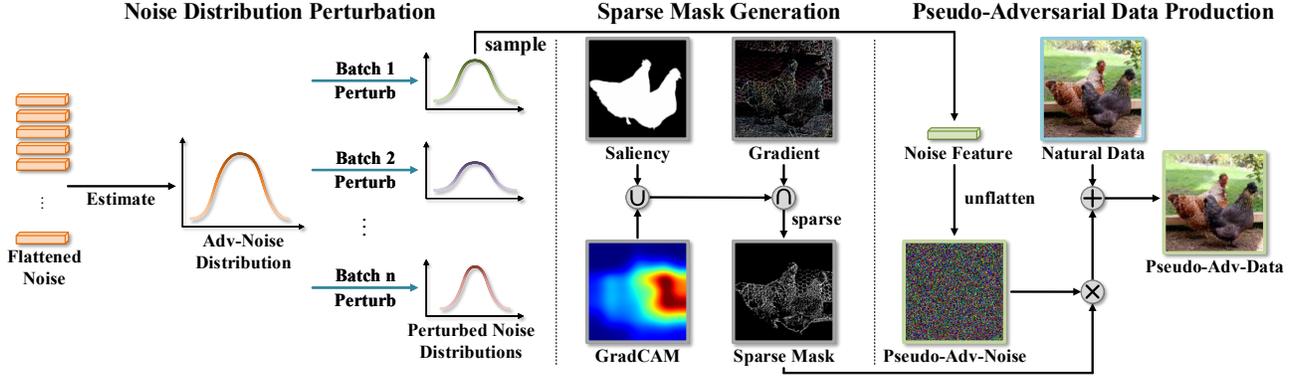


Figure 2. Illustration of Perturbation Forgery. Before training, we estimate the noise distribution from a commonly used attack, then continuously perturb it in each batch to create an open covering of the adversarial noise distributions. Next, noises sampled from these perturbed distributions are converted into localized noise by applying sparse masks. Finally, pseudo-adversarial data is generated by adding these localized noises to natural samples.

Definition 1 (Proximal Noise Distribution). Let P and Q be the distribution functions of two independent noise samples \mathbf{x}_1 and \mathbf{x}_2 . We call them proximal noise distributions under ε if we have a metric $d(\cdot, \cdot)$ and $\exists \varepsilon > 0$, s.t. $d(P, Q) < \varepsilon$.

A common assumption for instance feature distribution is to model it as a Gaussian distribution [10]. Obtained by subtracting from natural data and adversarial data, adv-noises naturally form a Gaussian distribution. Therefore, we assume that noise r.v $\boldsymbol{\eta} \in \mathbb{R}^d$ satisfied truncated Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_d)$ with it's truncated interval is $[-\varepsilon, \varepsilon]^d$. Built upon Definition 1, we derive the following theorem about the association between the distributions of different adv-noises:

Theorem 1. Let \mathcal{P}_a be the distribution set composed of all the adversarial noise distributions. Given independent noise distributions $Q_i, i \in N^+$. For $\forall i \neq j, Q_i$ and Q_j are proximal noise distributions if the following conditions are met.

- 1) $Q_i, Q_j \in \mathcal{P}_a$.
- 2) $\exists \epsilon_\mu, \epsilon_\sigma > 0$ s.t. $\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\| < \epsilon_\mu$ and $|\sigma_i - \sigma_j| < \epsilon_\sigma$ where the $\|\cdot\|$ is a Euclidean norm on \mathbb{R}^n .

Theorem 1 tells us: 1) All the adversarial noise distributions under the same ℓ^p norm are proximal noise distributions. 2) If the parameters of two noise distributions are sufficiently close to those of a given distribution, we can regard these two distributions as proximal noise distributions.

We can further deduce several corollaries of the theorem. 1) All noise distributions proximal to the known adversarial distribution $p(\boldsymbol{\eta})$ form an open ε -ball centered on $p(\boldsymbol{\eta})$, denoting as $\mathcal{P}_\varepsilon = \{p(\boldsymbol{\eta}_i) | d(p(\boldsymbol{\eta}_i), p(\boldsymbol{\eta})) < \varepsilon\}$, and also form a metric space $(\mathcal{P}_\varepsilon, d)$. 2) The adv-noise distribution set is also located in this ball: $\mathcal{P}_a \subset \mathcal{P}_\varepsilon$. 3) Based on the prop-

erties of the metric space $(\mathcal{P}_\varepsilon, d)$, for all subset $\mathcal{P}_i \subset \mathcal{P}_\varepsilon$, there exists an open covering of the subset: $\mathcal{P}_i \subset \{G_c\}_{c \in \mathbb{I}}$.

Based on the corollaries, we deduce the core idea of this paper: Let $p(\hat{\boldsymbol{\eta}}_i | \boldsymbol{\eta}) = \mathcal{N}(\boldsymbol{\mu} + \epsilon_{\mu i}, (\sigma + \epsilon_{\sigma i})^2 \mathbf{I}_d), i = 1, 2, \dots, n$ denote generated proximal noise distributions of the given adv-noise distribution $p(\boldsymbol{\eta}) = \mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_d)$, where $\epsilon_{\mu i}, \epsilon_{\sigma i}$ are randomly sampled from uniform distributions and restricted by $\epsilon_{\mu i}, \epsilon_{\sigma i} < \varepsilon$. As shown in Figure 1, **when $n \rightarrow \infty$, we can always obtain a distribution set $\mathcal{P}_c = \{p(\hat{\boldsymbol{\eta}}_i | \boldsymbol{\eta})\}_{i=1}^\infty$ that forms an open covering of the adversarial distribution set \mathcal{P}_a . By training on the open covering, the detector learns to distinguish between the distributions of natural and adversarial data, enabling it to effectively detect unseen adversarial attacks.**

Analysis of distribution assumption, proofs of theorem, and corollaries are provided in the supplementary materials.

4. Perturbation Forgery

Based on the theory in the last section, we heuristically propose Perturbation Forgery to train a detector with strong generalization against various types of unseen attacks.

As illustrated in Figure 2, Perturbation Forgery encompasses three elements: noise distribution perturbation, sparse mask generation, and pseudo-adversarial data production. At first, the noise distribution of a commonly used attack is continuously perturbed using noise distribution perturbation into numerous noise distributions that remain proximal to the original, thereby forming the open covering. After that, we sample noises from the perturbed distribution and convert them into sparse local noise using sparse mask generation. Subsequently, we convert half of the natural data into pseudo-adv-data using pseudo-adversarial data production. This pseudo-adv data replaces actual adversarial data and is combined with the remaining natural data to

train a binary detector.

4.1. Noise Distribution Perturbation

Given the training dataset $\{(\mathbf{x}_k, y_k)\}_{k=1}^n$ composed by natural data, we add adv-noise on it using a commonly used attack (such as FGSM) and generate adversarial noise features $\mathcal{D}_{\text{noise}} := \{\mathbf{z}_k = h(\boldsymbol{\eta}_k)\}_{k=1}^n$ by

$$\boldsymbol{\eta}_k = \text{Attack}(\mathbf{x}) - \mathbf{x}, k \in [1, n], \quad (2)$$

where $h : \mathcal{X} \rightarrow \mathbb{R}^d$ is the flattening operation that converts multidimensional data into vectors.

Assuming the noise features form a multivariate Gaussian distribution $p(\mathbf{z}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ [10], we estimate the parameters mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ of $\mathcal{D}_{\text{noise}}$ by

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_k \mathbf{z}_k, \quad \hat{\boldsymbol{\Sigma}} = \frac{1}{N} \sum_k (\mathbf{z}_k - \hat{\boldsymbol{\mu}})(\mathbf{z}_k - \hat{\boldsymbol{\mu}})^\top. \quad (3)$$

For the i^{th} batch of training, we perturb the estimated distribution to a proximal noise distribution by

$$\hat{\boldsymbol{\mu}}_i = \hat{\boldsymbol{\mu}} + \alpha_i \cdot \mathbf{m}_i, \quad \hat{\boldsymbol{\Sigma}}_i = \hat{\boldsymbol{\Sigma}} + \beta_i \cdot \mathbf{V}_i, \quad (4)$$

where $\mathbf{m}_i \sim \mathcal{N}(0, \mathbf{I}_d)$ and $\mathbf{V}_i \sim \mathcal{N}(0, \mathbf{I}_{d \times d})$ are vector and matrix of perturbation. $\alpha_i \sim \mathcal{U}(-\epsilon_\mu, \epsilon_\mu)$ and $\beta_i \sim \mathcal{U}(0, \epsilon_\sigma)$ are scale parameters. After n batches of training, we have a perturbed distribution set $\{\mathcal{N}(\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Sigma}}_i)\}$, thereby forming the approximate open covering \mathcal{P}_c . Trained with \mathcal{P}_c , the detector can distinguish the distributions of natural and adversarial data.

4.2. Sparse Mask Generation

Unlike gradient-based adversarial attacks, generative-based attacks tend to generate imbalanced and anisotropic perturbations, showing much intensity in high-frequency and salient areas and mild in low-frequency and background areas. Besides, in physical scenarios, adversarial attacks also tend to add patches to local areas. Noticing this, we propose sparse masks to perturbation forgery to assist the detector in detecting local adv-noises. Previous research has demonstrated that the spectra of real and fake images differ significantly in high-frequency regions [27]. Additionally, generative-based adversarial data manipulate characteristics, such as abnormal color aberrations, in these high-frequency regions [40]. To augment the detection performance on generative-based adversarial data, we need to sparse the pseudo-noise sampled from perturbed noise distribution and highlight the noise in high-frequency regions to mimic the style of generative-based attacks.

Given this, for a natural sample $\mathbf{x}_{i,k}$ in the i^{th} batch, we first mask the high-frequency area using saliency detection [33] and GradCAM [34]:

$$\text{Mask}_1(\mathbf{x}_{i,k}) = \text{Map}(f_s(\mathbf{x}_{i,k}); \gamma_s) \cup \text{Map}(f_c(\mathbf{x}_{i,k}); \gamma_c), \quad (5)$$

Algorithm 1 Training Procedure with Perturbation Forgery

Input: Training dataset, a commonly used attack, saliency detection model f_s , and GradCAM model f_c .

Output: A detector with strong generalization capabilities against various types of unseen attacks.

- 1: Generate adversarial noise using the commonly used attack in Eq. 2.
 - 2: Estimate a multivariate Gaussian distribution $p(\mathbf{z}) = \mathcal{N}(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ of adv-noise using Eq. 3.
 - 3: **for** batch i **in** $1, \dots, n$ **do**
 - 4: Generate a perturbed noise distribution $p(\mathbf{z}_i) = \mathcal{N}(\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Sigma}}_i)$ using Eq. 4.
 - 5: **for** sample $\mathbf{x}_{i,k}$ in half of batch i **do**
 - 6: Generate sparse masks $\text{Mask}(\mathbf{x}_{i,k})$ using Eq. 8.
 - 7: Sample pseudo-adv noise $\hat{\boldsymbol{\eta}}_{i,k}$ using Eq. 9.
 - 8: Generate pseudo-adv data $\hat{\mathbf{x}}_{i,k}$ using Eq. 11 and replace $\mathbf{x}_{i,k}$.
 - 9: **end for**
 - 10: Train the detector on batch i .
 - 11: **end for**
-

where f_s and f_c are the saliency detection model and GradCAM model respectively. $\text{Map}(\mathbf{x}; \gamma)$ is a binary indicator function parameterized by γ :

$$\text{Map}(\mathbf{x}_{i,k}; \gamma) = \text{Clip}_{[0,1]}(\mathbf{x}_{i,k} - \gamma \mathbf{I}_{\mathcal{X}}). \quad (6)$$

The adv-noise that occurs in the high-frequency region tends to occupy only a small part of the region. Therefore, we select the high-frequency point from the gradient map obtained by Sobel operator [11] to sparsify the mask of sample $\mathbf{x}_{i,k}$:

$$\text{Mask}_2(\mathbf{x}_{i,k}) = \text{Map}(\text{Sobel}(\mathbf{x}_{i,k}); \gamma_l), \quad (7)$$

$$\text{Mask}(\mathbf{x}_{i,k}) = \text{Mask}_1(\mathbf{x}_{i,k}) \cap \text{Mask}_2(\mathbf{x}_{i,k}). \quad (8)$$

4.3. Pseudo-Adversarial Data Production

For the i^{th} batch of training, we sample pseudo-adv noise from the perturbed distribution by:

$$\hat{\boldsymbol{\eta}}_{i,k} = h^{-1}(\mathbf{z}_{i,k}), \quad (9)$$

$$\mathbf{z}_{i,k} \in \{\mathbf{z}_k | \mathbf{z}_k \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Sigma}}_i) \text{ and } p(\mathbf{z}_k) < \gamma_p\} \quad (10)$$

where $h^{-1} : \mathbb{R}^d \rightarrow \mathcal{X}$ is an inverse function of h to unflatten noise features, and γ_p is a threshold that excludes data with a low likelihood. Note that $\boldsymbol{\eta}_i$ is a global noise and needs to be localized by a sparse mask. The pseudo-adv data $\hat{\mathbf{x}}_{i,k}$ is produced by adding the masked local noise to $\mathbf{x}_{i,k}$:

$$\hat{\mathbf{x}}_{i,k} = \mathbf{x}_{i,k} + \hat{\boldsymbol{\eta}}_{i,k} \otimes \text{Mask}(\mathbf{x}_{i,k}). \quad (11)$$

where \otimes is Hadamard product.

Table 2. Comparison of AUROC scores of detecting gradient-based and adaptive attacks on ImageNet100 under $\epsilon = 4/255$.

Detector	BIM	PGD	DIM	MIM	NIM	AA	BPDA+EOT	MM
LID [28]	0.9782	0.9750	0.8942	0.9146	0.8977	0.9124	0.9202	0.9227
LiBRe [8]	0.9259	0.9548	0.9243	0.8725	0.9013	0.8653	0.8714	0.8573
SPAD [40]	0.9846	0.9851	0.9815	0.9820	0.9823	0.9890	0.9885	0.9811
EPSAD [49]	0.9998	0.9989	0.9923	0.9918	0.9972	0.9998	0.9985	0.9923
ours	0.9911	0.9912	0.9863	0.9931	0.9934	0.9941	0.9927	0.9944

For each batch, we convert half of the natural data to pseudo-adv data by Perturbation Forgery and train a binary detector using the Cross-Entropy classification loss [50]. The overall training procedure with Perturbation Forgery is presented in Algorithm 1.

5. Experiments

To validate the detection performance of our approach against various adversarial attack methods, we conduct extensive experiments on multiple general and facial datasets. For additional experiments, analyses, and a discussion of limitations, please refer to the supplementary material.

5.1. Experiment Settings

Datasets The datasets used for evaluation include: 1) general datasets: ImageNet100 [7] and CIFAR-10 [23] for evaluate gradient-based and generative-based attacks, and 2) face datasets: Makeup [16] for Adv-Makeup [46], CelebA-HQ [21] for AMT-GAN [19], and LFW [14] for TIPIM [45] and physical attacks.

Adversarial Attacks We adopt 5 gradient-based attacks, 3 adaptive attacks, 5 generative-based attacks, and 6 face attacks to evaluate our method. For gradient-based attacks, BIM [24], PGD [29], MIM [9], DIM [43], and NIM [26] are used as the unseen attacks. For adaptive attacks, we adopt AutoAttack (AA) [6], BPDA+EOT [18], and Minimum-Margin Attack (MM) [13]. For generative-based attacks, we adopt 3 GAN-based attacks including CDA [31], TTP [30], and M3D [51], and 2 Diffusion-based attacks including Diff-Attack [4] and Diff-PGD [44]. For face attacks, we adopt 1 gradient-based attack TIPIM [45], 2 GAN-based attacks including AdvMakeup [46] and AMT-GAN [19], and 3 physical attacks including Adv-Sticker [38], Adv-Glasses [35], and Adv-Mask [38]. For each of the above attack methods, adversarial data is generated with the attack intensity of $\epsilon = 4/255$ (except for the unrestricted attacks), and iterative attacks run for 10 steps using step size 2.

Baselines To evaluate our method across various types of adversarial attacks, we compare it with state-of-the-art detection methods from three related research domains:

adversarial detection, synthetic image detection, and face forgery detection. For gradient-based attacks, we benchmark against four adversarial detection methods: LID [28], LiBRe [8], SPAD [40], and EPSAD [49]. Since most adversarial detection methods are not designed to address generative-based adversarial attacks, we include a comparison with four synthetic image detection techniques: CNN-Detection [41], LGrad [37], Universal-Detector [32], and DIRE [42]. For detecting GAN-based face attacks, we also compare against two face forgery detection methods, Luo et al.[27] and He et al.[17], as well as ODIN [25].

Evaluation Metrics Following previous adversarial detection works [49], we employ the Area Under the Receiver Operating Characteristic Curve (AUROC) [20] as the primary evaluation metric to assess the performance of our classification models. The AUROC is a widely recognized metric in the field of machine learning and is particularly useful for evaluating binary classification problems. For fair comparisons, the number of natural data and adversarial data in the experiments remained consistent.

Implementation details XceptionNet [5] is selected as our backbone model and is trained on all the above datasets. For adversarial noise distribution estimation, we utilize Torchattacks [22] to generate 1000 noise images for each dataset. The uniform distribution parameters are set as $\epsilon_\mu = 3$ and $\epsilon_\sigma = 0.005$. The threshold of the sample control is set as $\gamma_s = 0.05$. The thresholds of the binary indicator function are set as $\gamma_s = 0$, $\gamma_c = 125$ and $\gamma_l = 100$. FGSM [15] is used as the initial attack, i.e. the commonly used attack, for Perturbation Forgery. Training epochs are set to 10 and convergence is witnessed.

5.2. Detecting Gradient-based Attacks

We begin by comparing our method with state-of-the-art techniques in detecting gradient-based attacks on ImageNet100. All the attacks are unseen by both the baseline methods and our method. For each attack, 1,000 adversarial images are generated in the gray-box setting, where the attacker has access to the victim model but is blind to the detection model. All baseline methods are sourced from existing adversarial detection research.

Table 3. Comparison of AUROC scores of detecting generative-based attacks on ImageNet100.

Detector	CDA	TTP	M3D	Diff-Attack	Diff-PGD
CNN-Detection [41]	0.7051	0.6743	0.6917	0.3963	0.5260
LGrad [37]	0.6144	0.6077	0.6068	0.5586	0.5835
Universal-Detector [32]	0.7945	0.8170	0.8312	0.9202	0.5531
DIRE [42]	0.8976	0.9097	0.9129	0.9097	0.9143
SPAD	0.9385	0.9064	0.8984	0.8862	0.8879
EPSAD	0.9674	0.6997	0.7265	0.4700	0.1507
ours	0.9878	0.9678	0.9364	0.9223	0.9223

Table 4. Comparison of AUROC scores for detecting various adversarial attacks on the face dataset.

Detector	TIPIM	Adv-Sticker	Adv-Glasses	Adv-Mask	Detector	Adv-Makeup	AMT-GAN
LID	0.6974	0.5103	0.6472	0.5319	He et al. [17]	0.5252	0.8823
LiBRe	0.5345	0.9739	0.8140	0.9993	Luo et al. [27]	0.6178	0.6511
EPSAD	0.9647	0.9462	0.9260	0.9578	ODIN[25]	0.6325	0.6970
SPAD	0.9256	0.9975	0.9244	0.9863	SPAD	0.9657	0.8969
ours	0.9885	0.9999	0.9999	0.9999	ours	0.9762	0.9216

As reported in Table 2, although EPSAD performs well, our method demonstrates effective detection (achieving an AUROC score of 0.9931 against MIM and 0.9944 against MM) across a variety of attack methods. These results validate our assumptions about the proximity of the noise distribution. By continuously perturbing the estimated noise distribution of a given attack (FGSM), we obtain an open covering noise of distributions from all the attacks. Given one seen attack, we are able to detect all the unseen attacks. The consistent detection performance verifies the effectiveness of our core idea of training a detector to distinguish the open covering of adv-noise distributions from natural ones and demonstrates the strong generalization capability of our method against gradient-based attacks.

5.3. Detecting Generative-based Attacks

To investigate our method’s advantage in detecting generative-based attacks, we compared it with state-of-the-art synthetic image detection and adversarial detection methods on ImageNet100. As shown in Table 3, our method significantly outperforms the baseline methods, demonstrating highly effective detection capabilities.

For adversarial detection baselines, the imbalanced noise generated by GAN and diffusion models makes it difficult for these methods, which are typically designed for gradient-based attacks, to detect effectively. In contrast, our sparse mask approach enables effective detection of such localized noise.

Most synthetic image detection methods are less effective in detecting adversarial data crafted from natural samples, with the exception of DIRE[42]. This effectiveness may stem from DIRE’s use of reconstruction error from im-

ages generated by diffusion models, which maintains its robustness in this setting.

5.4. Detecting Face Adversarial Examples

Adversarial attacks targeting face recognition models represent an important branch of adversarial example research. Therefore, the detection ability of adversarial face examples is an important indicator of the performance of the detector. We perform various adversarial face attacks and compare our approach with state-of-the-art methods from adversarial face detection and face forgery detection.

As shown in Table 4, our method shows excellent performance against a variety of adversarial attacks targeting faces and surpasses many existing adversarial face detection techniques. It further indicates the ability of the sparse mask to capture the essential characteristics of face adversarial examples and the strong generalization capability of our method.

5.5. Ablation Study and Analysis

To further evaluate our method, we report an ablation study on the sparse mask, white-box attack detection results, the impact of the initial attack and the data size for distribution estimation, and the functionality of Perturbation Forgery through visualizations. The impact of hyper-parameters and other analyses are reported in the supplementary materials.

5.5.1. Impact of each mask

As argued, the sparse mask is proposed to assist the detector in identifying local adversarial noises in generative-based adversarial data. As shown in Table 5, the absence of the saliency detection model f_s or the GradCAM model f_c has

Table 5. Ablation study on masks (AUROC score), where f_s denotes saliency detection model and f_c represents the CAM model.

Ablation	CDA	TTP	M3D	Diff-Attack	Diff-PGD
w/o f_s	0.9755	0.9585	0.9320	0.9197	0.9135
w/o f_c	0.9864	0.9510	0.9262	0.9031	0.9062
w/o Sobel	0.9868	0.7235	0.6984	0.5231	0.5547

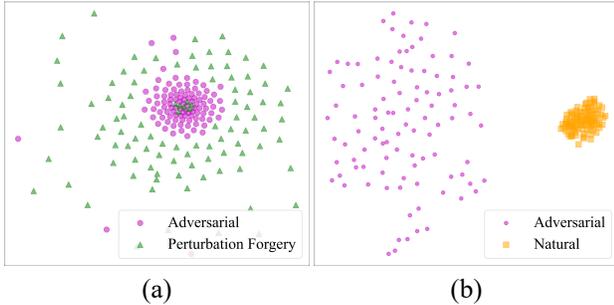


Figure 3. 2D T-SNE visualizations. (a) ImageNet100 flattened noises of adversarial data and Perturbation Forgery-generated data. (b) ImageNet100 features extracted by the backbone model trained with Perturbation Forgery.

Table 6. AUROC scores against white-box attacks under $\epsilon = 4/255$ on CIFAR-10.

Attack	BIM	PGD	DIM	MIM	AA
AUROC	0.9180	0.9168	0.9062	0.9031	0.9019

little effect on the overall system. This is because the regions masked by f_s and f_c are somewhat complementary. However, without the Sobel operation to sparsify the mask, the detector fails to detect most attacks, except for CDA. This is because generative-based adversarial noise tends to be imbalanced and localized, while CDA’s noise is more global. The generated pseudo-adversarial noise without Sobel becomes too dense, preventing the detector from effectively focusing on localized noise.

5.5.2. Functionality Visualization

To validate the functionality of our method, we extract features and flattened noise from adversarial data and Perturbation Forgery-generated data, then visualize them using 2D t-SNE projections. All adversarial data originate from unseen attacks. As shown in Figure 3(a), Perturbation Forgery continuously perturbs the adv-noise distribution of the initial attack, thereby forming an open covering for noises from unseen attacks. Individual samples appear as outliers, floating outside the collective noise distribution. As shown in Figure 3(b), by learning to separate natural data from those generated by Perturbation Forgery, the trained detector can effectively distinguish between natural and adversarial data.

Table 7. Impact of data size on distribution estimation: AUROC score against PGD under $\epsilon = 4/255$ on CIFAR-10.

Data size	100	250	500	750	1000	2000
AUROC	0.8278	0.9368	0.9866	0.9921	0.9965	0.9948

5.5.3. Detecting White-box Attack

To verify the robustness of our approach to white-box attacks, i.e., both the victim model and the detector are available to the attacker, we conduct the experiment on CIFAR-10. Following [3], a modified objective of attacks is:

$$\max_{\eta} \mathcal{L}(f(\hat{x}), y) + \alpha \cdot \mathcal{L}(f_d(\hat{x}), y_d), \text{ s.t. } \|\eta\| \leq \epsilon \quad (12)$$

where y is the ground-truth label, $y_d = 0$ indicates that \hat{x} is an adversarial example, and $\alpha = 1$ is a constant balancing between the attack of the classifier f and detector f_d . As shown in Table 6, our method achieves a high average AUROC of 0.9092, demonstrating its robustness against white-box attacks. This may be attributed to the comprehensive open covering formed by perturbation forgery, which effectively encompasses adversarial data from white-box attacks, making it harder to mislead the trained detector.

5.5.4. Data size for Distribution Estimation

To assess the influence of data size for distribution estimation, we train detectors using distributions estimated from various data sizes and test them against PGD on CIFAR-10. As shown in Table 7, our method only requires a small amount of data (≥ 500 samples) to estimate the adversarial noise distribution so as to train an effective detector. Notably, using fewer than 500 samples leads to a significant decrease in detection performance.

6. Conclusion

In this paper, we explore and define the proximity relationship between adversarial noise distributions. Based on our assumption and definition, we develop a perturbation method to craft proximal distributions and demonstrate an open covering of adversarial noise spaces. Building on this, we propose training detectors on the open coverings to achieve strong generalization against diverse adversarial attacks. We introduce Perturbation Forgery, which combines noise perturbation, sparse mask generation, and pseudo-adversarial data synthesis, enabling detection of unseen gradient-based, generative, and physical attacks, independent of specific models. Extensive experiments validate our approach’s effectiveness and generalization with minimal inference overhead.

7. Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 62372203 and 62302186, in part by the Major Scientific and Technological Project of Shenzhen (202316021), in part by the National key research and development program of China(2022YFB2601802), in part by the Major Scientific and Technological Project of Hubei Province (2022BAA046, 2022BAA042).

References

- [1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, pages 274–283. PMLR, 2018. 1
- [2] Oliver Bryniarski, Nabeel Hingun, Pedro Pachuca, Vincent Wang, and Nicholas Carlini. Evading adversarial example detection defenses with orthogonal projected gradient descent. *arXiv preprint arXiv:2106.15023*, 2021. 1
- [3] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 3–14, 2017. 8
- [4] Jianqi Chen, Hao Chen, Keyan Chen, Yilan Zhang, Zhengxia Zou, and Zhenwei Shi. Diffusion models for imperceptible and transferable adversarial attack. *arXiv preprint arXiv:2305.08192*, 2023. 3, 6
- [5] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, pages 1800–1807, 2017. 6
- [6] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, pages 2206–2216. PMLR, 2020. 3, 6
- [7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6
- [8] Zhijie Deng, Xiao Yang, Shizhen Xu, Hang Su, and Jun Zhu. Libre: A practical bayesian approach to adversarial detection. In *CVPR*, pages 972–982, 2021. 1, 2, 3, 6
- [9] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, pages 9185–9193, 2018. 3, 6
- [10] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. Vos: Learning what you don’t know by virtual outlier synthesis. *arXiv preprint arXiv:2202.01197*, 2022. 4, 5
- [11] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Willey & Sons, New York, 1973. 5
- [12] Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017. 1
- [13] Ruize Gao, Jiong Xiao Wang, Kaiwen Zhou, Feng Liu, Binghui Xie, Gang Niu, Bo Han, and James Cheng. Fast and reliable evaluation of adversarial robustness with minimum-margin attack. In *ICML*, pages 7144–7163. PMLR, 2022. 3, 6
- [14] B. Huang Gary, Ramesh Manu, Berg Tamara, and Learned-Miller Erik. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, 2007. 6
- [15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2014. 2, 3, 6
- [16] Qiao Gu, Guanzhi Wang, Mang Tik Chiu, Yu-Wing Tai, and Chi-Keung Tang. Ladrn: Local adversarial disentangling network for facial makeup and de-makeup. In *ICCV*, pages 10480–10489, 2019. 6
- [17] Yang He, Ning Yu, Margret Keuper, and Mario Fritz. Beyond the spectrum: Detecting deepfakes via re-synthesis. In *IJCAI*, pages 2534–2541. IJCAI, 2021. Main Track. 6, 7
- [18] Mitch Hill, Jonathan Mitchell, and Song-Chun Zhu. Stochastic security: Adversarial defense using long-run dynamics of energy-based models. *arXiv preprint arXiv:2005.13525*, 2020. 3, 6
- [19] Shengshan Hu, Xiaogeng Liu, Yechao Zhang, Minghui Li, Leo Yu Zhang, Hai Jin, and Libing Wu. Protecting facial privacy: Generating adversarial identity masks via style-robust makeup transfer. In *CVPR*, pages 15014–15023, 2022. 3, 6
- [20] Alberto Jiménez-Valverde. Insights into the area under the receiver operating characteristic curve (auc) as a discrimination measure in species distribution modelling. *Global Ecology and Biogeography*, 21(4):498–507, 2012. 6
- [21] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 6
- [22] Hoki Kim. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020. 6
- [23] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical report*, 2009. 6
- [24] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale, 2016. 2, 6
- [25] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017. 6, 7
- [26] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *ICLR*, 2019. 3, 6
- [27] Yuchen Luo, Yong Zhang, Junchi Yan, and Wei Liu. Generalizing face forgery detection with high-frequency features. In *CVPR*, pages 16312–16321, 2021. 5, 6, 7
- [28] Xingjun Ma, Bo Li, Yisen Wang, Sarah M. Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E. Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality, 2018. 1, 2, 3, 6
- [29] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017. 1, 2, 6
- [30] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. On generating transferable targeted perturbations. In *ICCV*, pages 7708–7717, 2021. 3, 6

- [31] Muhammad Muzammal Naseer, Salman H Khan, Muhammad Haris Khan, Fahad Shahbaz Khan, and Fatih Porikli. Cross-domain transferability of adversarial perturbations. *NeurIPS*, 32, 2019. 3, 6
- [32] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *CVPR*, pages 24480–24489, 2023. 6, 7
- [33] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, 2019. 5
- [34] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 5
- [35] Mahmood Sharif, Sruti Bhagavatula, Lujio Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *SIGSAC*, pages 1528–1540, 2016. 6
- [36] Yuxuan Shi, Hefei Ling, Lei Wu, Jialie Shen, and Ping Li. Learning refined attribute-aligned network with attribute selection for person re-identification. *Neurocomputing*, 402: 124–133, 2020. 1
- [37] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *CVPR*, pages 12105–12114, 2023. 6, 7
- [38] Liang Tong, Zhengzhang Chen, Jingchao Ni, Wei Cheng, Dongjin Song, Haifeng Chen, and Yevgeniy Vorobeychik. Facesec: A fine-grained robustness evaluation framework for face recognition systems. In *CVPR*, pages 13254–13263, 2021. 6
- [39] Florian Tramèr. Detecting adversarial examples is (nearly) as hard as classifying them. In *ICML*, pages 21692–21702. PMLR, 2022. 1
- [40] Qian Wang, Yongqin Xian, Hefei Ling, Jinyuan Zhang, Xiaorui Lin, Ping Li, Jiazhong Chen, and Ning Yu. Detecting adversarial faces using only real face self-perturbations. In *IJCAI*, pages 1488–1496, 2023. Main Track. 1, 2, 3, 5, 6
- [41] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *CVPR*, pages 8695–8704, 2020. 6, 7
- [42] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *ICCV*, pages 22445–22455, 2023. 6, 7
- [43] Weibin Wu, Yuxin Su, Michael R. Lyu, and Irwin King. Improving the transferability of adversarial samples with adversarial transformations. In *CVPR*, pages 9020–9029, 2021. 3, 6
- [44] Haotian Xue, Alexandre Araujo, Bin Hu, and Yongxin Chen. Diffusion-based adversarial sample generation for improved stealthiness and controllability. *NeurIPS*, 36, 2024. 1, 3, 6
- [45] Xiao Yang, Yinpeng Dong, Tianyu Pang, Hang Su, Jun Zhu, Yuefeng Chen, and Hui Xue. Towards face encryption by generating adversarial identity masks. In *ICCV*, pages 3877–3887, 2021. 6
- [46] Bangjie Yin, Wenxuan Wang, Taiping Yao, Junfeng Guo, Zelun Kong, Shouhong Ding, Jilin Li, and Cong Liu. Advmakeup: A new imperceptible and transferable attack on face recognition. In *IJCAI*, pages 1252–1258, 2021. 3, 6
- [47] Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *ICML*, pages 12062–12072, 2021. 1
- [48] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *ICML*, pages 7472–7482, 2019. 1
- [49] Shuhai Zhang, Feng Liu, Jiahao Yang, Yifan Yang, Changsheng Li, Bo Han, and Mingkui Tan. Detecting adversarial data by probing multiple perturbations using expected perturbation score. In *ICML*, pages 41429–41451. PMLR, 2023. 1, 2, 3, 6
- [50] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *NeurIPS*, 2018. 6
- [51] Anqi Zhao, Tong Chu, Yahao Liu, Wen Li, Jingjing Li, and Lixin Duan. Minimizing maximum model discrepancy for transferable black-box targeted attacks. In *CVPR*, pages 8153–8162, 2023. 1, 3, 6