EXPECTED PERTURBATION SCORE FOR ADVERSARIAL DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Adversarial detection aims to determine whether a given sample is an adversarial one based on the discrepancy between natural and adversarial distributions. Unfortunately, estimating or comparing two data distributions is extremely difficult, especially in the high-dimension space. Recently, the *gradients* of log probability density (a.k.a., score) w.r.t. samples is used as an alternative statistic to compute. However, we find that the score is sensitive in identifying adversarial samples due to insufficient information with one sample only. In this paper, we propose a new statistic called *expected perturbation score* (EPS), which is essentially the expected score of a sample after various perturbations. Specifically, to obtain adequate information regarding one sample, we can perturb it by adding various noises to capture its multi-view observations. We theoretically prove that EPS is a proper statistic to compute the discrepancy between two distributions under mild conditions. In practice, we can use a pre-trained diffusion model to estimate EPS for each sample. Last, we propose an EPS-based adversarial detection (EPS-AD) method, in which we develop EPS-based *maximum mean discrepancy* (MMD) as a metric to measure the discrepancy between the test sample and natural samples. To verify the validity of our proposed method, we also prove that the EPS-based MMD between natural and adversarial samples is larger than that among natural samples. Empirical studies on CIFAR-10 and ImageNet across different network architectures including ResNet, WideResNet, and ViT show the superior adversarial detection performance of EPS-AD compared to existing methods.

1 INTRODUCTION

Deep neural networks (DNNs) are known to be sensitive to adversarial samples that are generated by adding imperceptible perturbations to the input but may mislead the model to make unexpected predictions (Szegedy et al., 2014; Goodfellow et al., 2015). Adversarial samples threaten widespread machine learning systems (Li & Vorobeychik, 2014; Ozbulak et al., 2019), which raises an urgent requirement for advanced techniques to improve the robustness of models. Among them, *adversarial training* introduces adversarial data into training to improve the robustness of models but suffers from significant performance degradation and high computational complexity (Sriramanan et al., 2021; Laidlaw et al., 2021; Wong et al., 2020); *adversarial purification* relies on generative models to purify adversarial data before classification, which still has to compromise on unsatisfactory natural and adversarial accuracy (Shi et al., 2021; Yoon et al., 2021; Nie et al., 2022).

In contrast, another class of defense methods, called *adversarial detection*, could be achieved by detecting and rejecting adversarial examples, which are friendly to existed machine learning systems due to the lossless natural accuracy, and can help to identify security-compromised input sources (Abusnaina et al., 2021). Adversarial detection aims to tell whether an upcoming sample is an adversarial sample, for which the key is to find the discrepancy between the adversarial and natural distributions. However, existing adversarial detection approaches primarily train a tailored detector for specific attacks (Feinman et al., 2017; Ma et al., 2018; Lee et al., 2018) or for a specific classifier (Deng et al., 2021), which largely overlook modeling the adversarial and natural distributions, resulting in their limited performance against unseen attacks or transferable attacks.

Unfortunately, it is non-trivial to estimate or compare two data distributions, especially in the high-dimension space (*e.g.*, image-based space). One alternative approach is to estimate the *gradient* of

log probability density with respect to samples, *i.e.*, score. This statistic has emerged as a powerful means for adversarial defense (Yoon et al., 2021; Nie et al., 2022) and diffusion models (Song & Ermon, 2019; Song et al., 2021; Kingma et al., 2021; Huang et al., 2021). However, how to effectively exploit the score function for adversarial detection is not well studied.

Recently, Yoon et al. (2021) purify adversarial samples by gradually removing the adversarial noise from the (attacked) samples with the score function for adversarial defense. During the purification process, they employ the *norm* of scores (between being-purified adversarial samples and natural samples) to set a threshold for determining which timestep to stop purifying. They empirically find that natural samples usually have lower score norms than adversarial samples across purification timesteps. Intuitively, the score could represent the momentum of the sample towards the high-density areas of natural data (Song & Ermon, 2019). From this point of view, a lower score norm indicates the sample is closer to the high-density areas of natural data, *i.e.*, a higher probability of the sample following the natural distribution. To further understand this, we demonstrate the score norms of natural samples and adversarial samples at different purification timesteps. According to Figure 1, most natural samples have lower score norms than adversarial samples at the same timestep, but they are very sensitive to the timesteps due to the significant overlap across all timesteps. This suggests that the score of one sample is useful but not effective enough in identifying the adversarial samples.

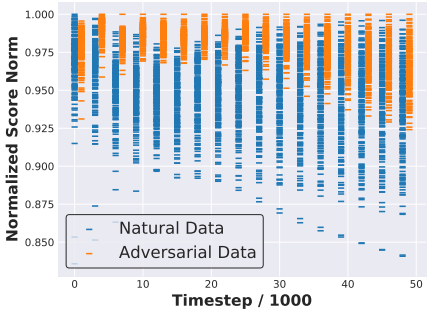


Figure 1: An illustration of score norms of 200 random sampled natural images and adversarial images on ImageNet, in which most natural images have lower score norms than adversarial images at the same timestep but are very sensitive to the timesteps due to the significant overlap.

In this paper, we propose a new statistic called *expected perturbation score* (EPS), which represents the expected score after multiple perturbations of a given sample. In EPS, we consider multiple levels of noise perturbations to diversify one sample, allowing us to capture multi-view observations of one sample and thus extract adequate information from the data. Our theoretical analysis shows EPS is a valid statistic in distinguishing between natural and adversarial samples under mild conditions. Thus, we propose an *EPS-based adversarial detection* method (EPS-AD) for adversarial detection, as illustrated in Figure 2. Specifically, given a pre-trained score-based diffusion model, EPS-AD consists of three steps: 1) we simultaneously add multiple perturbations to a set of natural images and an upcoming image following the forward diffusion process with a time step T^*; 2) we obtain their EPSs via the score model; 3) we adopt the *maximum mean discrepancy* (MMD) between any test sample and natural samples relying on EPS.

We provide both empirical and theoretical analyses to demonstrate the effectiveness of EPS-AD. Empirically, we achieve superior performance on both CIFAR-10 and ImageNet across many network architectures including ResNet, WideResNet and ViT compared to existing methods. Especially for the extremely low attack intensity $\epsilon = 1/255$ on ImageNet, we achieve the area under the receiver operating characteristic (AUROC) of $+93\%$ against 12 attacks over ResNet-50. Theoretically, we prove that the MMD between EPSs of the natural samples is smaller than that between natural and adversarial samples.

We summarize our main contributions as follows:

- We propose a new and reliable statistic called *expected perturbation score* (EPS) to capture sufficient information regarding a sample from its multi-view observations by adding various perturbations. We theoretically prove that EPS is a proper statistic to compute the discrepancy between two distributions under mild conditions.
- Relying on the proposed EPS, we exploit the *maximum mean discrepancy* (MMD) as a metric to measure the discrepancy between the test sample and natural samples and then develop a novel adversarial detection method called EPS-AD. We theoretically show that the EPS-based MMD between natural and adversarial samples is larger than that among natural samples, which verifies the validity of the proposed adversarial detection method.

2 PRELIMINARIES

We start by recalling the setting of score-based continuous-time diffusion models, and then present the concept of *maximum mean discrepancy*.

Adversarial data generation. Given a well-trained classifier $\hat{f}$ on a data set $D\{(\mathbf{x}_i, l_i)\}_{i=1}^n$ with $\mathbf{x}_i$ being a sample from the input space $\mathcal{X}$ and l_i being its ground-truth label defined in a label set $\mathcal{C} = \{1, \cdots, C\}$, adversarial data $\hat{\mathbf{x}}$ regarding $\mathbf{x}$ with perturbation ϵ is given by

$$\hat{\mathbf{x}} = \underset{\hat{\mathbf{x}} \in \mathcal{B}(\mathbf{x}, \epsilon)}{\arg\max}\, \ell\left(\hat{f}(\hat{\mathbf{x}}), l\right), \tag{1}$$

where $\mathcal{B}(\mathbf{x}, \epsilon) = \{\mathbf{x}' \in \mathcal{X} \mid d(\mathbf{x}, \mathbf{x}') \leq \epsilon\}$, d is some distance (*e.g.*, ℓ_2 or ℓ_∞ distance), and ℓ is some loss function. For simplicity, we denote $\hat{\mathbf{x}} = \mathbf{x} + \epsilon$ as the adversarial data regarding $\mathbf{x}$.

Continuous-time diffusion models. Following Song et al. (2021), let $p(\mathbf{x})$ be the unknown data distribution. Diffusion models firstly construct a forward diffusion process $\{\mathbf{x}_t\}_{t=0}^{T_{\text{diff}}}$ indexed by a continuous time variable $t \in [0, T_{\text{diff}}]$, which can be modeled by a stochastic differential equation (SDE) with positive time increments:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \tag{2}$$

where $\mathbf{x}_0 := \mathbf{x} \sim p(\mathbf{x})$, $\mathbf{f}(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d$ is a vector-valued function, $g(\cdot) : \mathbb{R} \to \mathbb{R}$ is a scalar function that is independent on $\mathbf{x}$, and $\mathbf{w}$ is a standard Wiener process.

Let $p_t(\mathbf{x})$ be the the marginal distribution of $\mathbf{x}_t$ with $p_0(\mathbf{x}) = p(\mathbf{x})$, if $\mathbf{f}(\mathbf{x}, t)$ and $g(t)$ are designed well such that $p_T(\mathbf{x}) \approx \mathcal{N}(\mathbf{0}, \boldsymbol{I}_d)$, by reversing the diffusion process from $t = T_{\text{diff}}$ to $t = 0$, we can reconstruct samples $\mathbf{x}_0 \sim p_0(\mathbf{x})$. The reverse process is given by the reverse-time SDE:

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right] \mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}}, \tag{3}$$

where $\bar{\mathbf{w}}$ is a standard reverse-time Wiener process and $\mathrm{d}t$ is an infinitesimal negative time step. Throughout the paper, we consider the VP-SDE following the setting of Song et al. (2021), where $\mathbf{f}(\mathbf{x}, t) := -\frac{1}{2}\beta(t)\mathbf{x}$ and $g(t) := \sqrt{\beta(t)}$ with $\beta(t)$ being the linear noise schedule, *i.e.*, $\beta(t) := \beta_{\text{min}} + (\beta_{\text{max}} - \beta_{\text{min}})\, t/T_{\text{diff}}$ for $t \in [0, T_{\text{diff}}]$.

Reconstructing samples from the Gaussian distribution requires the score of the marginal distribution, *i.e.*, $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ in the reverse process Eq. (3). To estimate the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, one effective solution is to train a score model $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, t)$ on samples with score matching (Hyvärinen & Dayan, 2005; Song & Ermon, 2019; Vincent, 2011). The training objective function is:

$$\boldsymbol{\theta}^* = \min_{\boldsymbol{\theta}} \mathbb{E}_t \left\{\lambda(t) \mathbb{E}_{\mathbf{x}_0 \sim p_0(\mathbf{x}_0)} \mathbb{E}_{\mathbf{x}_t \sim p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)} \left\| s_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0) \right\|_2^2\right\}, \tag{4}$$

where $\lambda(t)$ is a positive weighting function, and $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$ is a transition kernel from $\mathbf{x}_0$ to $\mathbf{x}_t$, which can be derived by the forward SDE in Eq. (2).

Maximum mean discrepancy. Following Gretton et al. (2012); Borgwardt et al. (2006), let $\mathcal{X} \subset \mathbb{R}^d$ be a separable metric space and p, q be Borel probability measures on $\mathcal{X}$. Given two independent identically distributed (*IID*) observations $S_X = \{\mathbf{x}^{(i)}\}_{i=1}^n$ and $S_Y = \{\mathbf{y}^{(i)}\}_{i=1}^m$ from distributions p and q, respectively, *maximum mean discrepancy* (MMD) aims to measure the closeness between these two distributions, which is defined as:

$$\mathrm{MMD}(p, q; \mathcal{F}) := \sup_{f \in \mathcal{F}} |\mathbb{E}[f(X)] - \mathbb{E}[f(Y)]|, \tag{5}$$

where $\mathcal{F}$ is a class of functions $f : \mathcal{X} \to \mathbb{R}$, and $X \sim p, Y \sim q$ are two random variables. To better study the richness of the MMD function class $\mathcal{F}$, Borgwardt et al. (2006) propose to choose $\mathcal{F}$ to be the unit ball in a universal reproducing kernel Hilbert space and obtain the following kernel-based MMD,

$$\begin{aligned} \mathrm{MMD}\left(p, q; \mathcal{H}_k\right) &:= \sup_{f \in \mathcal{H}, \|f\|_{\mathcal{H}_k} \leq 1} |\mathbb{E}[f(X)] - \mathbb{E}[f(Y)]| \\ &= \|\mu_p - \mu_q\|_{\mathcal{H}_k} = \sqrt{\mathbb{E}\left[k\left(X, X'\right) + k\left(Y, Y'\right) - 2k(X, Y)\right]}, \end{aligned} \tag{6}$$

where $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the kernel of a reproducing kernel Hilbert space $\mathcal{H}_k$, $\mu_p := \mathbb{E}[k(\cdot, X)]$ and $\mu_q := \mathbb{E}[k(\cdot, Y)]$ are the kernel mean embeddings of p and q, respectively (Gretton et al., 2012; Borgwardt et al., 2006; Jitkrittum et al., 2017; Liu et al., 2020a; Gao et al., 2021).

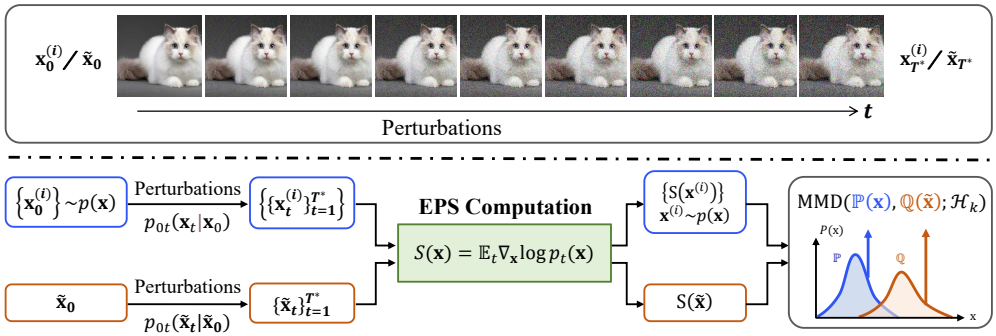


Figure 2: Overview of the proposed EPS-AD. EPS denotes the expected score after multiple perturbations of a sample using a pre-trained score model. Specifically, we simultaneously add perturbations to a set of natural images $\{\mathbf{x}_0^{(i)}\}$ and an upcoming image $\tilde{\mathbf{x}}_0$ following the diffusion process with a time step T^* to get perturbed images, from which we obtain their EPSs $S(\mathbf{x})$ via the score model and calculate the MMD between EPS of the upcoming sample and EPSs of natural samples.

3 Proposed Method

In this section, we first provide the problem definition of *adversarial detection* in Section 3.1. Then we present the definition of *expected perturbation score* (EPS) and provide the theoretical analysis that EPS is able to distinguish between natural and adversarial data well if natural data are from Gaussian distributions (Section 3.2). Motivated by this, we develop a new adversarial detection method, called EPS-AD, as shown in Figure 2. Particularly, we estimate the EPS for each sample using a pre-trained score-based diffusion model and then adopt the *maximum mean discrepancy* (MMD) between the test sample and natural samples relying on EPS as their characteristics for detection (Section 3.3). Moreover, we provide theoretical analysis that the MMD between EPSs of the natural samples is smaller than that between natural and adversarial samples. (Section 3.4).

3.1 Problem Setting

In this paper, we aim to address the following adversarial detection problem.

Problem 1. ***(Adversarial detection)*** *Let $\mathcal{X} \subset \mathbb{R}^d$ be a separable metric space and p be a Borel probability measure on $\mathcal{X}$, and IID observations $S_X = \{\mathbf{x}^{(i)}\}_{i=1}^n$ from the distribution p and a ground-truth labeling mapping $f(\cdot) := \mathbb{R}^d \to \mathcal{C}$ with $\mathcal{C} = \{1, \ldots, C\}$ being a label set. Assuming that the attacker has access to some well-trained classifier $\hat{f}$ on S_X and IID observations S'_X from the distribution p, we wish to know whether each new sample in $S_Y = \{\mathbf{y}^{(i)}\}_{i=1}^m$ that are crafted with S'_X is following the distribution p.*

Note that the definition of *adversarial detection* in Problem 1 is different from that in two-sample test (Grosse et al., 2017). Particularly, Problem 1 aims to determine whether each example in S_Y is sampled from the distribution p, while two-sample test aims to tell if two populations S_X and S_Y come from the same distribution, which focuses on the closeness between two populations.

3.2 Expected Perturbation Score

As aforementioned, the score of one sample, *e.g.*, $\nabla_{\mathbf{x}} \log p(\mathbf{x})$, is ineffective enough in identifying the adversarial samples. To capture more information from one sample, we propose a new statistic *expected perturbation score* (EPS), which is formulated as bellow.

Definition 1. ***(Expected perturbation score)*** *Let $\mathcal{X} \subset \mathbb{R}^d$ be a separable metric space and p be Borel probability measure on $\mathcal{X}$. Given a perturbation process transition distribution $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$, the expected perturbation score (EPS) of a sample $\mathbf{x}$ is given by:*

$$S(\mathbf{x}) = \mathbb{E}_{t \sim u(0,T)} \nabla_{\mathbf{x}} \log p_t(\mathbf{x}), \tag{7}$$

where $\mathbf{x}_0 := \mathbf{x}$ and T is the last perturbation time step.

Note that the perturbation process transition distribution $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$ can be any distribution, such as the commonly used Gaussian distribution or uniform distribution. Moreover, we consider multiple levels of noise in the definition of EPS, with the aim of diversifying the single sample, which enables us to capture multi-view observation data and thus fully exploit more information from the data.

Built upon Definition 1, we derive a following theorem to give a closer look on EPS $S(x)$ for the natural data and adversarial data when the perturbation process transition distribution $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$ and $p(x)$ are from Gaussian distributions.

Theorem 1. *Assuming that the distribution of natural data $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \sigma_{\mathbf{x}}^2\mathbf{I})$, given a perturbation transition kernel $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \gamma_t\mathbf{x}_0, \sigma_t^2\mathbf{I}\right)$ with γ_t and σ_t being the time-dependent noise schedule, then the following three conclusions hold for $S(\mathbf{x})$:*
1) For $\forall\, \mathbf{x}\sim p(\mathbf{x})$, $S(\mathbf{x}) \sim \mathcal{N}(\mathbf{0}, \sigma_S^2\mathbf{I})$;
2) For $\forall\, \mathbf{y}\sim p(\mathbf{x})$, adversarial sample $\hat{\mathbf{y}}=\mathbf{y}+\epsilon$ with perturbation ϵ, $S(\hat{\mathbf{y}})\sim\mathcal{N}(-\boldsymbol{\mu}_S, \sigma_S^2\mathbf{I})$;
3) For $\forall\, \mathbf{x}, \mathbf{y} \sim p(\mathbf{x})$, adversarial sample $\hat{\mathbf{y}}=\mathbf{y}+\epsilon$, we have

$$S(\mathbf{x}) - S(\mathbf{y}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, 2\sigma_S^2\mathbf{I}); \tag{8}$$

$$S(\mathbf{x}) - S(\hat{\mathbf{y}}) \xrightarrow{d} \mathcal{N}(\boldsymbol{\mu}_S, 2\sigma_S^2\mathbf{I}), \tag{9}$$

where $\boldsymbol{\mu}_S=\mathbb{E}_t\frac{\epsilon\mathbf{1}}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2}$ and $\sigma_S^2 = \mathbb{E}_t\frac{1}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2}$.

Theorem 1 tells us: 1) the first two findings show that the mean of EPS for the adversarial sample differs from that of the natural sample due to the additional perturbation term $\boldsymbol{\mu}_S$; 2) the third finding indicates that the EPS of the natural sample is closer to that of other natural samples compared to adversarial samples, and this discrepancy becomes more pronounced when the perturbation transition γ_t and σ_t are small. These findings motivate us to employ $S(\mathbf{x})$ for adversarial detection.

3.3 Adversarial Detection with Expected Perturbation Score

Estimation for expected perturbation score. Note that EPS (*i.e.*, $S(\mathbf{x})$) in Eq. (7) requires knowing the score function $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$, which can be estimated by training a neural network with score matching (Hyvärinen & Dayan, 2005; Song & Ermon, 2019; Vincent, 2011; Kingma et al., 2021). To this end, we model the perturbation process transition as a Gaussian distribution $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \gamma_t\mathbf{x}_0, \sigma_t^2\mathbf{I}\right)$, where $\gamma_t = e^{-\frac{1}{2}\int_0^t \beta(s)ds}$ and $\sigma_t^2 = 1-e^{-\int_0^t \beta(s)ds}$ with $\beta(t)$ for $t \in [0, 1000]$ being the time-dependent noise schedule. By optimizing Eq. (4), we can approximate $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ with $s_\theta(\mathbf{x}_t, t)$ (Song et al., 2021). Then we estimate EPS as $S(\mathbf{x}) \approx \mathbb{E}_t s_\theta(\mathbf{x}_t, t)$.

MMD for characterizing expected perturbation score. Using the norm of estimated EPS $S(x)$ as a characterization for adversarial detection is straightforward. Nevertheless, using the norm of $S(x)$ as the criterion can only describe the magnitude of the EPS vector, so it ignores the rich information that can be derived from its direction. It is critical that we design a distance metric to measure the distance between the EPS of an upcoming sample and the EPSs of natural samples in order to derive more useful information from $S(x)$.

Benefiting from the superior performance of maximum mean discrepancy (MMD) in measuring two given distributions (Long et al., 2015; Zhu et al., 2019; Gao et al., 2021), we resort to it for characterizing EPS. The basic idea of MMD is that two distributions would be identical if two random variables are identical for any order, and the moment that makes the largest distance between the two distributions should be the measure of the two distributions when the two distributions are not the same (Smola et al., 2007; Gong et al., 2022). Formally, we define two distributions for the natural samples $\{\mathbf{x}^{(i)}\}_{i=1}^n$ as $\mathbb{P}_X = \frac{1}{n}\sum_{i=1}^n \delta\left(\left\|\mathbf{x} - \mathbf{x}^{(i)}\right\|\right)$ and one upcoming sample $\tilde{\mathbf{x}}$ as $\mathbb{Q}_Y = \delta(\|\mathbf{y} - \tilde{\mathbf{x}}\|)$, respectively, where δ is the Dirac function (Alt, 2006). Then we estimate the distance between these two distributions as

$$\widehat{\mathrm{MMD}}_b^2\left[\mathbb{P}_X, \mathbb{Q}_Y; \mathcal{H}_k\right] = \left[\frac{1}{n^2}\sum_{i,j=1}^n k\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right) - \frac{2}{n}\sum_{i=1}^n k\left(\mathbf{x}^{(i)}, \mathbf{y}\right) + k(\mathbf{y}, \mathbf{y})\right], \tag{10}$$

where $k(\mathbf{x}, \mathbf{y}) = \kappa(S(\mathbf{x}), S(\mathbf{y}))$ with κ being the kernel of a reproducing kernel Hilbert space $\mathcal{H}_k$ such as the Gaussian kernel $\kappa(\mathbf{x}, \mathbf{y}) = \exp\left(-\|\mathbf{x} - \mathbf{y}\|^2 / \left(2\sigma^2\right)\right)$. In our practice, we consider multiple kernel MMD following Long et al. (2015).

3.4 THEORETICAL ANALYSIS FOR EPS-AD

Note that after performing the same perturbation process, the first and the third terms in Eq. (10) are the same for each upcoming sample, thus we only focus on the cross-term $J = \frac{2}{n}\sum_{i=1}^{n} k\left(\mathbf{x}^{(i)}, \mathbf{y}\right)$. Next, we analyze this term for the adversarial sample.

Corollary 1. *Considering the Gaussian kernel* $\kappa\left(\mathbf{x}, \mathbf{y}\right) = \exp\left(-\left\|\mathbf{x}-\mathbf{y}\right\|^2 / \left(2\sigma^2\right)\right)$ *and the assumption in Theorem 1, for* $\forall 0<\eta<1$*, the probability of* $P\{k(\mathbf{x}, \hat{\mathbf{y}}) > \eta\}$ *is given by*

$$P\{k(\mathbf{x}, \hat{\mathbf{y}})>\eta\} = \int_0^C \chi_d^2(\|\boldsymbol{\mu}_S\|^2)dx, \tag{11}$$

where $\boldsymbol{\mu}_S$ *denotes the mean of* $S(\mathbf{x}) - S(\hat{\mathbf{y}})$*, C is a constant for given* η *and* σ*,* χ^2 *is the probability density function of noncentral chi-squared distribution with* d *degrees of freedom (Abdel-Aty, 1954).*

Corollary 1 indicates that the cross-term J will be larger if $\|\boldsymbol{\mu}_S\|$ is close to zero given an η. Combining Eq. (8) and Eq. (9), we conclude that the natural data have larger J than the adversarial data with higher probability due to the additional term $\mathbb{E}_t \frac{\epsilon \mathbf{1}}{\gamma_t^2 \sigma_{\mathbf{x}}^2 + \sigma_t^2}$, suggesting that the MMD between EPSs of the natural samples is smaller than that between natural and adversarial samples.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Datasets and network architectures. We evaluate our method on CIFAR-10 (Krizhevsky, 2009) and ImageNet (Deng et al., 2009). We implement three widely used architectures as classifiers: WideResNet (Zagoruyko & Komodakis, 2016; Gowal et al., 2021) for CIFAR-10, ResNet (He et al., 2016) and ViT (Dosovitskiy et al., 2021) for ImageNet. For diffusion models, we consider the pre-trained diffusion models of Score SDE (Song et al., 2021) for CIFAR-10 and Guided Diffusion (Dhariwal & Nichol, 2021) for ImageNet, respectively.

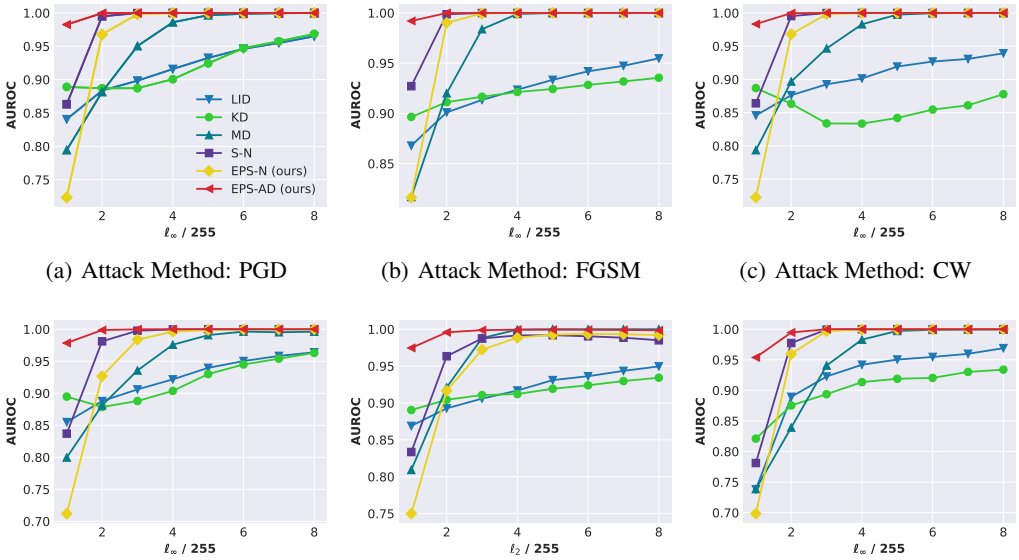
Attack methods. Following Deng et al. (2021), we evaluate our adversarial detection method under various attack methods. We consider the commonly used ℓ_2 and ℓ_∞ threat models, including PGD (Madry et al., 2018), FGSM (Goodfellow et al., 2015), BIM (Kurakin et al., 2018), MIM (Dong et al., 2018), TIM (Dong et al., 2019), CW (Carlini & Wagner, 2017), DI_MIM(Xie et al., 2019). Moreover, we apply two adaptive attack methods such as AutoAttack (AA) (Croce & Hein, 2020) and Minimum-Margin Attack (MM) (Gao et al., 2022). To show the superiority of our method, we consider the relatively low attack intensities, *i.e.*, ℓ_2-ball and ℓ_∞-ball ϵ-constrains with $\epsilon = 4/255$, and iterative attacks run for 5 steps using step size $\epsilon/5$, unless stated otherwise.

Baselines. We compare our method with several state-of-the-art adversarial detection methods, including kernel density (KD) (Feinman et al., 2017), local intrinsic dimensionality (LID) (Ma et al., 2018), mahalanobis distance (MD) (Lee et al., 2018) and LiBRe (Deng et al., 2021). Besides, we construct two new adversarial detection methods based on diffusion models: 1) S-N: using the score norm of raw images, *i.e.*, $||s_\theta(\mathbf{x}, t)||^2$. 2) EPS-N: using the norm of the EPS, *i.e.*, $||S(\mathbf{x})||^2$. Differently, our proposed EPS-AD further calculates the *maximum mean discrepancy* of EPSs.

Evaluation metric. We evaluate the performance of adversarial detection approaches with the area under the receiver operating characteristic (AUROC), which is a widely used statistic for assessing the discriminatory capacity of distribution models (Jiménez-Valverde, 2012). Considering the computational cost of applying 12 attacks to the classifier, especially for the ImageNet, following Gao et al. (2021), we randomly select two disjoint subsets as adversarial and natural samples (each containing 500 samples) and compute the AUROC value over these two subsets. Notably, our method is applicable to different data set sizes, which is verified in Appendix. Moreover, we set $T^* = 20$ in $S(x)$ on CIFAR-10 and $T^* = 50$ on ImageNet for both our EPS-AD and EPS-N, and set $t^* = 5$ in $s_\theta(\mathbf{x}, t)$ on CIFAR-10 and $t^* = 20$ on ImageNet for S-N.

Table 1: Comparison of different adversarial detection methods on CIFAR-10 and ImageNet in terms of AUROC under $\epsilon = 4/255$. The bold number indicates the best results.

Dataset	AUROC	FGSM	PGD	BIM	CW	FGSM-ℓ_2	BIM-ℓ_2	AA
CIFAR-10	KD	0.9213	0.9007	0.9037	0.8335	0.9121	0.9107	0.9135
	LID	0.9236	0.9159	0.9217	0.9014	0.9169	0.9320	0.9380
	MD	0.9990	0.9860	0.9758	0.9829	**0.9995**	0.9586	0.9830
	S-N	**1.0000**	0.9998	0.9961	0.9998	0.9885	0.9674	0.9995
	EPS-N (Ours)	**1.0000**	**1.0000**	0.9996	**1.0000**	0.9916	0.9883	**1.0000**
	EPS-AD (Ours)	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**0.9995**	**0.9989**	**1.0000**
ImageNet	KD	0.7099	0.9720	0.9797	0.9413	0.7004	0.9775	0.9319
	LID	0.8912	0.9750	0.9808	0.9528	0.8932	0.9816	0.9582
	MD	0.8786	0.9773	0.9835	0.9609	0.8715	0.9830	0.9518
	LiBRe	0.9889	0.9530	0.9269	0.9039	0.8708	0.9211	0.8724
	S-N	0.9828	0.8974	0.7208	0.8969	0.8830	0.6762	0.9151
	EPS-N (Ours)	0.9987	0.9978	0.9215	0.9978	0.8191	0.7172	0.9943
	EPS-AD (Ours)	**1.0000**	**1.0000**	**0.9999**	**1.0000**	**0.9975**	**0.9895**	**1.0000**



(a) Attack Method: PGD (b) Attack Method: FGSM (c) Attack Method: CW

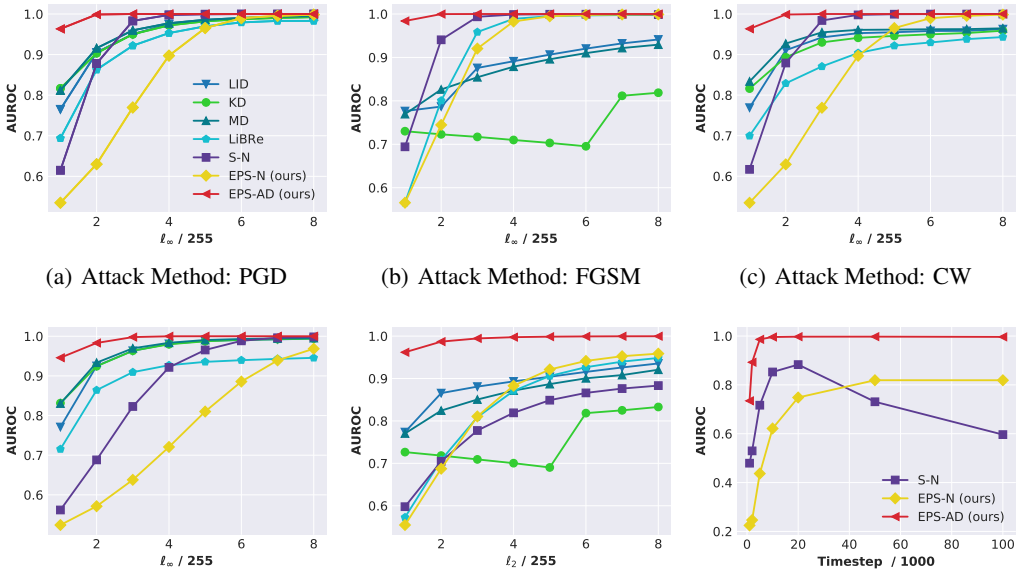
(d) Attack Method: BIM (e) Attack Method: FGSM-ℓ_2 (f) Attack Method: AA Attack

Figure 3: Comparison with adversarial detection methods on CIFAR-10 in terms of AUROC under $\epsilon \in [1/255, 8/255]$. Sub-figures (a) - (f) share the same legend presented in sub-figure (a).

4.2 Detecting on Known Attacks

We start by comparing EPS-AD with the state-of-the-art adversarial detection methods that are trained with seen adversarial examples, against ℓ_2 and ℓ_∞ threat models. Considering adversarial detection becomes more challenging when the attack intensity of adversarial samples is low, to broadly evaluate the adversarial detection performance, we compare our method with other baselines on different attacks under different attack intensities. Moreover, to show the best performance of KD, LID and MD, we test the detection performance on their corresponding adversarial examples.

Results on CIFAR-10. Figure 3 shows our adversarial detection performance against 6 attacks under different attack intensities $\epsilon \in [1/255, 8/255]$ on CIFAR-10 over WideResNet-28-10 (Zagoruyko & Komodakis, 2016) compared to other baselines. We demonstrate other attacks and more results on WideResNet-70-16 (Gowal et al., 2021) in Appendix. Obviously, our EPS-AD has much higher AUROC performance than other methods. Critically, we observe that EPS-AD pre-



(a) Attack Method: PGD (b) Attack Method: FGSM (c) Attack Method: CW

(d) Attack Method: BIM (e) Attack Method: FGSM-ℓ_2 (f) Impact of timestep

Figure 4: Results of adversarial detection on ImageNet. Sub-figures (a) - (e) report the AUROC on different attacks under $\epsilon \in [1/255, 8/255]$ and share the same legend presented in sub-figure (a). Sub-figure (f) reports the AUROCs of different timesteps in $\{1, 2, 5, 10, 20, 50, 100\}$.

serves almost non-degraded AUROC even if the attack intensity reaches a severely low value (*e.g.*, $\epsilon = 1/255$) under ℓ_2 attacks (*e.g.*, FGSM-ℓ_2), which shows the stability of EPS-AD when handling challenging adversarial samples.

In addition, we report quantitative results for adversarial detection under the attack intensity $\epsilon = 4/255$ in Table 1. The results show that by employing EPS and measuring their MMD, EPS-AD consistently outperforms existing methods against various attacks in terms of AUROC. We also see that by simply applying the norm of EPS, EPS-N achieves superior adversarial detection performance, which demonstrates the effectiveness of EPS.

Taken together, the results in Figure 3 and Table 1 demonstrate the effectiveness of our method in detecting against ℓ_2 and ℓ_∞ attacks on CIFAR-10 dataset. It is worth noting that in contrast to the methods that are trained with the specific ℓ_p attacks, our method is agnostic to the threat models.

Results on ImageNet. We report the adversarial detection performance against ℓ_2 and ℓ_∞ attacks on ImageNet over ResNet-50 (He et al., 2016) in Table 1 and Figure 4 . We defer the results of one widely-used ViT architecture, DeiT-S (Dosovitskiy et al., 2021), in Appendix. We observe that our approach consistently outperforms baselines under various attacks. Significantly, EPS-AD exceeds baselines with a large margin when detecting CW, FGSM-ℓ_2 and AA attacks . These results reveal that our proposed EPS-AD is effective even on a large-scale data set. Moreover, we observe that EPS-N exhibits very poor results compared to our EPS-AD when detecting ℓ_2 attacks (*e.g.*, FGSM-ℓ_2 and BIM-ℓ_2) since the norm of EPS method ignores rich information contained in the EPS vector, which is not effective enough when detecting the adversarial samples on a large-scale data set. This performance degradation is more pronounced in method S-N.

4.3 Detecting on Unseen Attacks and Transferable Attacks

In light of poor performance for adversarial detection baselines against unseen attacks and transferable attacks, we evaluate our method in the context of these kinds of attacks.

Detecting on unseen attacks. To detect unseen attacks, we train KD, LID, and MD detectors on CIFAR-10 using only FGSM adversarial examples and evaluate their performance under other attacks. Combining Table 2 and 1, we find that adversarial detection performance for MD and LID worsens. An explanation is that their detectors are trained with vectorized features extracted from the

Table 2: Comparison of AUROC for detecting unseen attacks on CIFAR-10, where "FGSM (seen)" denotes the seen adversarial attack used for the training of KD, LID and MD.

	FGSM(seen)	PGD	BIM	CW	FGSM-ℓ_2	BIM-ℓ_2	AA
KD	0.9213	0.9005	0.9080	0.8335	0.9144	0.9144	0.9133
LID	0.9236	0.8852	0.8891	0.8736	0.9181	0.8885	0.9092
MD	0.9990	0.9855	0.9733	0.9828	0.9994	0.9491	0.9820
S-N	**1.0000**	0.9998	0.9961	0.9998	0.9885	0.9674	0.9995
EPS-N (Ours)	**1.0000**	**1.0000**	0.9996	**1.0000**	0.9916	0.9883	**1.0000**
EPS-AD (Ours)	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**0.9995**	**0.9989**	**1.0000**

Table 3: Comparison of AUROC for detecting transferable attacks on ImageNet, where KD, LID, MD and LiBRe are trained with adversarial examples with ResNet-50 but detect the adversarial examples crafted with ResNet-101.

	FGSM	PGD	BIM	CW	FGSM-ℓ_2	BIM-ℓ_2	AA
KD	0.7754	0.5999	0.5847	0.7632	0.7906	0.7756	0.7698
LID	0.8467	0.7627	0.7663	0.7704	0.8520	0.7925	0.7967
MD	0.8467	0.7698	0.7684	0.7665	0.8067	0.7759	0.788
*L*iBRe	0.9849	0.8414	0.7161	0.8286	0.8489	0.7250	0.8485
S-N	0.7671	0.6351	0.5676	0.6346	0.7254	0.5582	0.6558
EPS-N (Ours)	0.9820	0.9997	0.9979	0.9999	0.8893	0.8537	0.9996
EPS-AD (Ours)	**1.0000**	**1.0000**	**0.9998**	**1.0000**	**0.9975**	**0.9891**	**1.0000**

seen samples through logistic regression, resulting in their limited generalization on unseen attacks. In contrast, diffusion-based detection methods show superior performance since they directly model the distribution of natural data, without requiring specific adversarial samples for training.

Detecting on transferable attacks. To validate the transferability, we train KD, LID, MD and LiBRe detectors with ResNet-50 but detect the adversarial examples from a surrogate ResNet-101 model. Comparing Table 3 and 1, the non-diffusion-based methods (*e.g.*, KD, LID, MD and LiBRe) drop significantly against transferable attacks. By contrast, our EPS-AD method achieves significantly better transferability, since it does not rely on specific classifiers, but rather models the distribution of natural data, indicating its versatility in various attack scenarios.

4.4 Ablation Study on Impact of Timestep

We conduct experiments on ImageNet over ResNet-50 to show the effect of the timestep in Figure 4 (f). We compare three methods including S-N, our EPS-N and EPS-AD. We draw two observations: 1) our EPS-AD and EPS-N are insensitive to the timestep T while S-N fluctuates greatly with the timestep t; 2) As timestep T increases, EPS-AD and EPS-N exhibit progressively better performance, however, this gain gradually decreases when T exceeds the optimal value. This is due to the larger diffusion time, the mean $\boldsymbol{\mu}_S$ in Eq.(9) gradually approaches zeros, resulting in a smaller discrepancy between the natural and adversarial distributions. Nie et al. (2022) also confirms this phenomenon (Theorem 1). We defer more results on CIFAR-10 in Appendix.

5 Conclusion

In this paper, we propose a new statistic *expected perturbation score* (EPS) to capture the information from multi-view observations of one sample, which is able to distinguish between natural and adversarial data well. Relying on EPS, we propose a novel adversarial detection method, EPS-AD. We provide theoretical analysis to demonstrate the superiority of EPS-AD. Extensive experiments on CIFAR-10 and ImageNet across different network architectures including ResNet, WideResNet and ViT show our EPS-AD successfully detects adversarial samples in various attack scenarios.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of experimental results, we provide an exhaustive implementation in Section D and the code will be released upon acceptance.

REFERENCES

SH Abdel-Aty. Approximate formulae for the percentage points and the probability integral of the non-central χ 2 distribution. *Biometrika*, 41(3/4):538–540, 1954.

Ahmed Abusnaina, Yuhang Wu, Sunpreet Arora, Yizhen Wang, Fei Wang, Hao Yang, and David Mohaisen. Adversarial example detection using latent neighborhood graph. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7687–7696, 2021.

HW Alt. Lineare funktionalanalysis. eine anwendungsorientierte einführung.. aufl. *Springer-Verlag*, 10:978–3, 2006.

Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.

Zinelabinde Boulkenafet, Jukka Komulainen, Lei Li, Xiaoyi Feng, and Abdenour Hadid. Oulu-npu: A mobile face presentation attack database with real-world variations. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 612–618. IEEE, 2017.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 2206–2216. PMLR, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Zhijie Deng, Xiao Yang, Shizhen Xu, Hang Su, and Jun Zhu. Libre: A practical bayesian approach to adversarial detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 972–982, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193, 2018.

Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4312–4321, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Reuben Feinman, Ryan R Curtin, Saurabh Shintre, and Andrew B Gardner. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*, 2017.

Ruize Gao, Feng Liu, Jingfeng Zhang, Bo Han, Tongliang Liu, Gang Niu, and Masashi Sugiyama. Maximum mean discrepancy test is aware of adversarial attacks. In *International Conference on Machine Learning*, pp. 3564–3575. PMLR, 2021.

Ruize Gao, Jiongxiao 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 *Proceedings of the 39th International Conference on Machine Learning*, pp. 7144–7163. PMLR, 2022.

Li-Hua Gong, Ling-Zhi Xiang, Si-Hang Liu, and Nan-Run Zhou. Born machine model based on matrix product state quantum circuit. *Physica A: Statistical Mechanics and its Applications*, 593: 126907, 2022.

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

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. Improving robustness using generated data. *Advances in Neural Information Processing Systems*, 34:4218–4233, 2021.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(1):723–773, 2012.

Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*, 2017.

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*, 2016.

Chin-Wei Huang, Jae Hyun Lim, and Aaron C Courville. A variational perspective on diffusion-based generative models and score matching. *Advances in Neural Information Processing Systems*, 34:22863–22876, 2021.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

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.

Wittawat Jitkrittum, Wenkai Xu, Zoltán Szabó, Kenji Fukumizu, and Arthur Gretton. A linear-time kernel goodness-of-fit test. *Advances in Neural Information Processing Systems*, 30, 2017.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in Neural Information Processing Systems*, 34:21696–21707, 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. *(Technical Report) University of Toronto.*, 2009.

Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*, pp. 99–112. Chapman and Hall/CRC, 2018.

Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. In *International Conference on Learning Representations*, 2021.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, 31, 2018.

Bo Li and Yevgeniy Vorobeychik. Feature cross-substitution in adversarial classification. *Advances in Neural Information Processing Systems*, 27, 2014.

Feng Liu, Wenkai Xu, Jie Lu, Guangquan Zhang, Arthur Gretton, and Danica J Sutherland. Learning deep kernels for non-parametric two-sample tests. In *International Conference on Machine Learning*, pp. 6316–6326. PMLR, 2020a.

Yaojie Liu, Joel Stehouwer, and Xiaoming Liu. On disentangling spoof trace for generic face anti-spoofing. In *European Conference on Computer Vision*, pp. 406–422. Springer, 2020b.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, pp. 97–105. PMLR, 2015.

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. In *International Conference on Learning Representations*, 2018.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

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

Utku Ozbulak, Arnout Van Messem, and Wesley De Neve. Impact of adversarial examples on deep learning models for biomedical image segmentation. In *MICCAI*, 2019.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

Changhao Shi, Chester Holtz, and Gal Mishne. Online adversarial purification based on self-supervision. In *International Conference on Learning Representations*, 2021.

Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. A hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*, pp. 13–31. Springer, 2007.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Gaurang Sriramanan, Sravanti Addepalli, Arya Baburaj, et al. Towards efficient and effective adversarial training. *Advances in Neural Information Processing Systems*, 34:11821–11833, 2021.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

Chien-Yi Wang, Yu-Ding Lu, Shang-Ta Yang, and Shang-Hong Lai. Patchnet: A simple face anti-spoofing framework via fine-grained patch recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20281–20290, 2022.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2020.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2730–2739, 2019.

Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, pp. 12062–12072. PMLR, 2021.

Zitong Yu, Xiaobai Li, Xuesong Niu, Jingang Shi, and Guoying Zhao. Face anti-spoofing with human material perception. In *European Conference on Computer Vision*, pp. 557–575. Springer, 2020a.

Zitong Yu, Jun Wan, Yunxiao Qin, Xiaobai Li, Stan Z Li, and Guoying Zhao. Nas-fas: Static-dynamic central difference network search for face anti-spoofing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9):3005–3023, 2020b.

Zitong Yu, Chenxu Zhao, Zezheng Wang, Yunxiao Qin, Zhuo Su, Xiaobai Li, Feng Zhou, and Guoying Zhao. Searching central difference convolutional networks for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5295–5305, 2020c.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*, 2016.

Ke-Yue Zhang, Taiping Yao, Jian Zhang, Ying Tai, Shouhong Ding, Jilin Li, Feiyue Huang, Haichuan Song, and Lizhuang Ma. Face anti-spoofing via disentangled representation learning. In *European Conference on Computer Vision*, pp. 641–657. Springer, 2020.

Yongchun Zhu, Fuzhen Zhuang, and Deqing Wang. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5989–5996, 2019.

APPENDIX OF EPS-AD

In appendix, we provide detailed proofs of the theorem and corollary, descriptions of related works, more details and more experimental results of the proposed EPS-AD. We organize the supplementary into the following sections. In Section A, we provide descriptions of related works regarding adversarial detection. In Section B, we derive the proofs of the theorem and corollary. In Section C, we demonstrate the pesudo code of our proposed EPS-AD. In Section D, we present detailed implementation of our experiments. In Section E, we show the impact of EPS in our method. In Section F, we study the impact of adding perturbations over samples in our method. In Section G, we report more comparison results, more ablation studies and more applications of our proposed EPS-AD.

A RELATED WORK

Diffusion models. Diffusion models have emerged as a powerful generative model in many synthesis tasks (Song & Ermon, 2019; Dhariwal & Nichol, 2021; Saharia et al., 2022; Rombach et al., 2022; Ramesh et al., 2022). Since then, many researchers exploit the diffusion model to adversarial purification for improving the robustness of model (Nie et al., 2022; Yoon et al., 2021), where the score becomes a powerful means. Yet, only few researchers apply them to adversarial detection. Recently, Yoon et al. (2021) employ the score model to purify the adversarial examples and use the norm of scores as the stopping condition of the purification. They also demonstrate some results about the norm of score in distinguishing adversarial samples from natural samples. However, this criterion is not effective enough for adversarial detection. In our work, we comprehensively consider multiple scores of perturbed samples, where these perturbed samples are from the same sample, and exploit rich information from these scores for adversarial detection. We empirically compare our approach to these methods and find that our approach outperforms these methods by a large margin.

Adversarial Attack. Numerous studies have been proposed to attack neural networks by slightly modifying the input data to trigger misclassifications of classifiers. We enumerate a series of such works in what follows. Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) simply adds small noise along the gradient of the loss function. To further adjust the direction of increment, Kurakin et al. (2018) propose Basic Iterative Method (BIM) that extends FGSM to iteratively take multiple steps. Madry et al. (2018) propose Projected Gradient Descent (PGD) by combining the iterative method with random initialization for the adversarial example. Meanwhile, (Dong et al., 2018) propose Momentum Iterative Method (MIM) by adding a momentum term to BIM for achieving a more stable attack. Besides, Dong et al. (2019) propose Translation-Invariant Method (TIM) by optimizing the perturbation over translated images to obtain more transferable attacks, which can be incorporated into the methods FGSM and BIM. Besides the non-targeted attack methods mentioned above, there are also many methods that perturb data to one target label. For example, Carlini & Wagner (2017) perform an attack by incorporating the iterative mechanism of BIM, called CW. Besides, there are approaches that combine multiple attacks to perturb data, such as the AutoAttack (Croce & Hein, 2020) and the Minimum-margin (MM) attack (Gao et al., 2022), the faster version of AutoAttack. The various attacks may cause serious consequences in security-critical tasks, raising an urgent requirement for advanced techniques to achieve a robust model.

Adversarial detection. To ensure the safety of machine learning system, a plethora of exploration for adversarial detection has attracted increasing sight. The most common idea is to filter out adversarial samples from test data using a trained binary classifier. Recently, statistics on hidden-layer features of DNNs are widely considered for adversarial detection. Feinman et al. (2017) train a logistic regression classifier using Kernel Density (KD) of features in the last hidden layer, as well as Bayesian Uncertainty (BU) as a basis. Ma et al. (2018) consider the local intrinsic dimensionality (LID) of the features of DNNs as the characteristics for detection. Lee et al. (2018) use a mahalanobis distance-based score to detect adversarial examples. In addition, Deng et al. (2021) train a Bayesian neural network by adding uniform noises to the samples. However, these methods train a tailored detector for some specific attacks or for a specific classifier, which largely overlooks the modeling for data distribution, leading to their limited performance under unknown attacks.

B PROOFS IN SECTION 3

B.1 PROOF OF THEOREM 1

Theorem 1 *Assuming that the distribution of natural data $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \sigma_{\mathbf{x}}^2\mathbf{I})$, given a perturbation transition kernel $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \gamma_t\mathbf{x}_0, \sigma_t^2\mathbf{I}\right)$ with γ_t and σ_t being the time-dependent noise schedule, then the following three conclusions hold for $S(\mathbf{x})$:*
1) For $\forall\, \mathbf{x}\sim p(\mathbf{x})$, $S(\mathbf{x}) \sim \mathcal{N}(\mathbf{0}, \sigma_S^2\mathbf{I})$;
2) For $\forall\, \mathbf{y}\sim p(\mathbf{x})$, adversarial sample $\hat{\mathbf{y}}=\mathbf{y}+\epsilon$ with perturbation ϵ, $S(\hat{\mathbf{y}})\sim\mathcal{N}(-\boldsymbol{\mu}_S, \sigma_S^2\mathbf{I})$;
3) For $\forall\, \mathbf{x}, \mathbf{y} \sim p(\mathbf{x})$, adversarial sample $\hat{\mathbf{y}}=\mathbf{y}+\epsilon$, we have

$$S(\mathbf{x}) - S(\mathbf{y}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, 2\sigma_S^2\mathbf{I}); \tag{12}$$

$$S(\mathbf{x}) - S(\hat{\mathbf{y}}) \xrightarrow{d} \mathcal{N}(\boldsymbol{\mu}_S, 2\sigma_S^2\mathbf{I}), \tag{13}$$

where $\boldsymbol{\mu}_S=\mathbb{E}_t\frac{\epsilon\mathbf{1}}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2}$ and $\sigma_S^2 = \mathbb{E}_t\frac{1}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2}$.

Proof. 1) Based on the distribution $p(\mathbf{x})$, *i.e.*, $\mathbf{x}_0 = \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \sigma_{\mathbf{x}}^2\mathbf{I})$, we obtain $\mathbf{x}_0 = \boldsymbol{\mu}_{\mathbf{x}} + \sigma_{\mathbf{x}}\mathbf{z}$ with $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$; based on the perturbation transition kernel $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \gamma_t\mathbf{x}_0, \sigma_t^2\mathbf{I}\right)$, we have $\mathbf{x}_t = \gamma_t\mathbf{x}_0 + \sigma_t\mathbf{z}$. Combining the distribution of $\mathbf{x}$ and $\mathbf{x}_0$, we have

$$\mathbf{x}_t = \gamma_t\boldsymbol{\mu}_{\mathbf{x}} + \sqrt{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2}\mathbf{I}, \quad \textit{i.e.}, \mathbf{x}_t \sim \mathcal{N}(\gamma_t\boldsymbol{\mu}_{\mathbf{x}}, \gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2) \tag{14}$$

For $\mathbf{x}_t \sim p_t(\mathbf{x}) = \mathcal{N}(\gamma_t\boldsymbol{\mu}_{\mathbf{x}}, \gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2)$, we calculate the derivation

$$\nabla_{\mathbf{x}} \log p_t(\mathbf{x}) = -\frac{\mathbf{x}_t - \gamma_t\boldsymbol{\mu}_{\mathbf{x}}}{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2} = -\frac{1}{\sqrt{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2}}\mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{15}$$

Taking expectation to t, we give the distribution of $S(\mathbf{x})$

$$S(\mathbf{x}) \sim \mathcal{N}(\mathbf{0}, \sigma_S^2\mathbf{I}), \text{ where } \sigma_S^2 = \mathbb{E}_t\frac{1}{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2}. \tag{16}$$

2) Based on $\mathbf{y} \sim p(\mathbf{x})$ and $\hat{\mathbf{y}} = \mathbf{y} + \epsilon$, we obtain $\hat{\mathbf{y}}_0 = \hat{\mathbf{y}} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}} + \epsilon, \sigma_{\mathbf{x}}^2\mathbf{I})$. Then, we have

$$\nabla_{\hat{\mathbf{y}}} \log p_t(\hat{\mathbf{y}}) = -\frac{\mathbf{y}_t + \epsilon - \gamma_t\boldsymbol{\mu}_{\mathbf{x}}}{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2} = -\frac{1}{\sqrt{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2}}\mathcal{N}(\mathbf{0}, \mathbf{I}) - \frac{\epsilon}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2}, \tag{17}$$

where the last equation is based on $p_t(\mathbf{y}) = p_t(\mathbf{x}) = \mathcal{N}(\gamma_t\boldsymbol{\mu}_{\mathbf{x}}, \gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2)$.

Taking expectation to t, we give the distribution of $S(\hat{\mathbf{y}})$

$$S(\hat{\mathbf{y}})\sim\mathcal{N}(-\boldsymbol{\mu}_S, \sigma_S^2\mathbf{I}), \text{ where } \boldsymbol{\mu}_S=\mathbb{E}_t\frac{\epsilon\mathbf{1}}{\gamma_t^2\sigma_{\mathbf{x}}^2+\sigma_t^2} \text{ and } \sigma_S^2 = \mathbb{E}_t\frac{1}{\gamma_t^2\sigma_{\mathbf{x}}^2 + \sigma_t^2}. \tag{18}$$

3) According to the additive property of the Gaussian distribution, combing Eq. (16) and Eq. (18), we obtain the third conclusion.

□

B.2 PROOF OF COROLLARY 1

Corollary 1 *Considering the Gaussian kernel $\kappa(\mathbf{x}, \mathbf{y}) = \exp\left(-\|\mathbf{x} - \mathbf{y}\|^2 / \left(2\sigma^2\right)\right)$ and the assumption in Theorem 1, for $\forall 0<\eta<1$, the probability of $P\{k(\mathbf{x}, \mathbf{y}) > \eta\}$ is given by*

$$P\{k(\mathbf{x}, \hat{\mathbf{y}})>\eta\} = \int_0^C \chi_d^2(\|\boldsymbol{\mu}_S\|^2)dx, \tag{19}$$

where $\boldsymbol{\mu}_S$ denotes the mean of $S(\mathbf{x}) - S(\hat{\mathbf{y}})$, C is a constant for given η and σ, χ^2 is the probability density function of noncentral chi-squared distribution with d degrees of freedom (Abdel-Aty, 1954).

Proof. Based on the Gaussian kernel $\kappa(\mathbf{x}, \mathbf{y})$, we have

$$P\{k(\mathbf{x}, \hat{\mathbf{y}}) > \eta\} = P\{\kappa(S(\mathbf{x}), S(\hat{\mathbf{y}})) > \eta\} = P\left\{\exp\left(-\|S(\mathbf{x}) - S(\hat{\mathbf{y}})\|^2 / \left(2\sigma^2\right)\right) > \eta\right\} \tag{20}$$

$$= P\left\{\|S(\mathbf{x}) - S(\hat{\mathbf{y}})\|^2 < -2\sigma^2 \ln \eta\right\}. \tag{21}$$

Let $\boldsymbol{\xi} \sim S(\mathbf{x}) - S(\hat{\mathbf{y}})$, then $\boldsymbol{\xi_i} \sim \mathcal{N}((\boldsymbol{\mu}_S)_i, \sigma_S^2)$, thus we have

$$P\{k(\mathbf{x}, \hat{\mathbf{y}}) > \eta\} = P\left\{\sum_{i=1}^{d} \boldsymbol{\xi_i}^2 < -2\sigma^2 \ln \eta\right\} = P\left\{\sum_{i=1}^{d} (\frac{\boldsymbol{\xi_i}}{\sigma_S})^2 < \frac{-2\sigma^2 \ln \eta}{\sigma_S^2}\right\} \tag{22}$$

Note that $\frac{\boldsymbol{\xi_i}}{\sigma_S} \sim \mathcal{N}((\boldsymbol{\mu}_S)_i, 1)$, based on the definition of noncentral chi-squared distribution (Abdel-Aty, 1954), we have

$$\sum_{i=1}^{d} (\frac{\boldsymbol{\xi_i}}{\sigma_S})^2 \sim \chi_d^2(\|\boldsymbol{\mu}_S\|^2) \tag{23}$$

Let $C = \frac{-2\sigma^2 \ln \eta}{\sigma_S^2}$, we obtain the conclusion

$$P\{k(\mathbf{x}, \hat{\mathbf{y}}) > \eta\} = \int_0^C \chi_d^2(\|\boldsymbol{\mu}_S\|^2) dx. \tag{24}$$

□

C PSEUDO-CODE OF EPS-AD

Algorithm 1: Expected Perturbation Score for Adversarial Detection (EPS-AD)

Input: Natural sample set $\{\mathbf{x}_0^{(i)}\}_{i=1}^n$, an upcoming sample $\tilde{\mathbf{x}}_0$, pre-trained score model $s_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ and diffusion timestep T^*.

Output: MMD between EPS of the upcoming sample $\tilde{\mathbf{x}}_0$ and EPSs of natural samples $\{\mathbf{x}_0^{(i)}\}$.

Set initial time step $t = 1$;

for $t = 1, \ldots, T^*$ **do**

  Obtaining perturbed natural samples $\{\mathbf{x}_t^{(i)}\}_{i=1}^n$ according to $p_{0t}(\mathbf{x}_t \mid \mathbf{x}_0)$;

  Obtaining perturbed upcoming sample $\tilde{\mathbf{x}}_t^{(i)}$ according to $p_{0t}(\hat{\mathbf{x}}_t \mid \tilde{\mathbf{x}}_0)$.

end

Compute EPSs of natural samples $\{S(\mathbf{x}^{(i)})\}_{i=1}^n$ using Eq. (7).

Compute EPS of the upcoming sample $S(\tilde{\mathbf{x}})$ using Eq. (7).

Compute the MMD between $\{S(\mathbf{x}^{(i)})\}_{i=1}^n$ and $S(\tilde{\mathbf{x}})$ using Eq. (10).

D MORE DETAILS FOR EXPERIMENT SETTINGS

D.1 IMPLEMENTATION DETAILS OF OUR METHOD

Our adversarial detection method is built upon diffusion models. Specifically, we consider the pre-trained diffusion model of Score SDE for CIFAR-10 following Song et al. (2021) and choose the *vp/cifar10_ddpmpp_deep_continuous* checkpoint from the *score_sde* library [1]; for ImageNet, we consider the pre-trained diffusion model of Guided Diffusion following Dhariwal & Nichol (2021) and use the 256×256 *diffusion (unconditional)* checkpoint from the *guided-diffusion* library [2].

For classifiers, we use pre-trained WiderRedNet-28-10 (Zagoruyko & Komodakis, 2016) and WiderRedNet-70-16 (Gowal et al., 2021) for CIFAR-10, and ResNet-50, ResNet-101 (He et al., 2016), and DeiT-S (Dosovitskiy et al., 2021) for ImageNet. For attacks, we consider 8 attack intensities $\epsilon \in [1/255, 8/255]$ with iterative attacks run for 5 steps using step size $\epsilon/5$ under 12 different ℓ_2 and ℓ_∞ attack methods to generate adversarial examples, including PGD, PGD-ℓ_2 (Madry et al., 2018), FGSM, FGSM-ℓ_2 (Goodfellow et al., 2015), BIM, BIM-ℓ_2 (Kurakin et al., 2018), MIM (Dong et al., 2018), TIM (Dong et al., 2019), CW (Carlini & Wagner, 2017), DI_MIM(Xie et al., 2019) and two adaptive attacks AutoAttack (AA) (Croce & Hein, 2020) and Minimum-Margin Attack (MM) (Gao et al., 2022) that is a faster version of AA. And for evaluation, we choose the area under the receiver operating characteristic curve (AUROC) as metric for adversarial detection. For the implementation of MMD, we consider multiple kernel MMD Long et al. (2015) following their codebase [3] We conduct our experiments based on Python 3.7 and Pytorch 1.7.1 on a server with 1× RTX 3090 GPU.

D.2 IMPLEMENTATION DETAILS OF BASELINES

Considering that most existing adversarial detection methods focus on training a detector using some specific attacks, we choose three standard adversarial detection approaches KD (Feinman et al., 2017), LID (Ma et al., 2018) and MD (Lee et al., 2018) as baselines for both CIFAR-10 and ImageNet, as well as a state-of-the-art detector LiBRe (Deng et al., 2021) individually on the ImageNet, which trains a Bayesian neural network by adding the uniform noises into the samples. Besides, we also construct two additional diffusion-based detection methods:1) S-N: using the score norm of raw images as the characteristics, *i.e.*, $||s_\theta(\mathbf{x}, t)||^2$; 2) EPS-N: using the norm of the EPS as the characteristics, *i.e.*, $||S(\mathbf{x})||^2$.

KD & LID & MD. We implement KD following the codebase[4], LID following the codebase[5], and MD following the codebasse[6]. These three methods train a logistic regressor to distinguish natural, noisy and adversarial samples. To show their best performance for adversarial detection, we choose the noise scale of the ℓ_2 distance in KD, LID and MD as 40, 1, 10 on CIFAR-10 and 20, 1, 10 on ImageNet. Besides, for KD, we choose bandwidth as 10 on CIFAR-10 and 20 on ImageNet.

LiBRe. We implement LiBRe on ImageNet following their codebase [7]. We evaluate its performance on ResNet-50 and ResNet-101 to make a comparison.

D.3 IMPLEMENTATION DETAILS OF FIGURE 1

For the results in Figure 1, we calculate the norms of scores of natural samples and adversarial samples at different purification timesteps using a score model pre-trained on ImageNet, where these adversarial samples are crafted by FGSM with $\epsilon = 1/255$. Before feeding these samples into the score model, we do not perturb them again. To better demonstrate these results, we normalize score norm with the maximum of scores in each timestep.

[1] https://github.com/yang-song/score_sde

[2] https://github.com/openai/guided-diffusion

[3] https://github.com/easezyc/deep-transfer-learning/tree/5e94d519b7bb7f94f0e43687aa4663aca18357de/MUDA/MFSAN

[4] https://github.com/rfeinman/detecting-adversarial-samples

[5] https://github.com/xingjunm/lid_adversarial_subspace_detection

[6] https://github.com/pokaxpoka/deep_Mahalanobis_detector

[7] https://github.com/thudzj/ScalableBDL

E IMPACT OF EPS FOR EPS-AD

In our method EPS-AD, we use diffusion-based score model to calculate the characteristics of samples, *i.e.*, EPS, which has the same dimension as this sample. In this experiment, we investigate the impact of EPS in our method. As a result, we remove the calculation of EPS from our method, instead using the raw sample as their characteristic. Table 4 shows adversarial detection performance of our method against 6 attacks under attack intensities $\epsilon \in \{2/255, 4/255\}$ on ImageNet over ResNet-50 compared to that without EPS. Obviously, EPS-AD without employing EPS demonstrates significant performance drop ($\approx 28\% \downarrow$), suggesting the superiority of our proposed EPS in distinguishing between adversarial and natural samples.

Table 4: Impact of EPS with ResNet-50 on ImageNet under $\epsilon = 2/255$ and $\epsilon = 4/255$.

Perturbation	Method	FGSM	PGD	BIM	MIM	TIM	CW
$\epsilon = 2/255$	EPS-AD (w/o EPS)	0.7132	0.7108	0.7101	0.7119	0.7120	0.7107
	EPS-AD (w/ EPS)	**0.9998**	**0.9988**	**0.9830**	**0.9977**	**0.9881**	**0.9987**
$\epsilon = 4/255$	EPS-AD (w/o EPS)	0.7176	0.7144	0.7131	0.7156	0.7156	0.7142
	EPS-AD (w/ EPS)	**1.0000**	**1.0000**	**0.9999**	**1.0000**	**0.9997**	**1.0000**
Perturbation	Method	DI_MIM	PGD-ℓ_2	FGSM-ℓ_2	BIM-ℓ_2	MM	AA
$\epsilon = 2/255$	EPS-AD (w/o EPS)	0.7117	0.7118	0.7133	0.7097	0.7066	0.7067
	EPS-AD (w/ EPS)	**0.9977**	**0.9995**	**0.9870**	**0.9656**	**0.9987**	**0.9987**
$\epsilon = 4/255$	EPS-AD (w/o EPS)	0.7156	0.7154	0.7175	0.7128	0.7101	0.7104
	EPS-AD (w/ EPS)	**1.0000**	**1.0000**	**0.9975**	**0.9895**	**1.0000**	**1.0000**

F IMPACT OF ADDING PERTURBATIONS OVER SAMPLES FOR EPS-AD

Adding perturbations into the samples is critical for our proposed EPS-AD. To investigate the impact of this operation, we conduct ablation studies against 6 adversarial attack methods on two datasets. Table 5 demonstrates adversarial detection performance of EPS-AD against 6 attacks under $\epsilon = 1/255$ both on CIFAR-10 over WideResNet-28-10 and on ImageNet over ResNet-50 compared to that without adding perturbations. We observe that the adversarial detection performance is significantly improved with adding perturbations ($5\% \sim 15\% \uparrow$), which coincides with the conclusion in Theorem 1 that adding the perturbations helps distinguish between adversarial and natural samples.

Table 5: Impact of adding perturbations with WideResNet-28-10 on CIFAR-10 and ResNet-50 on ImageNet under $\epsilon = 1/255$.

Dataset	Method	PGD	BIM	CW	FGSM-ℓ_2	BIM-ℓ_2	AA
CIFAR-10	EPS-AD (w/o perturbation)	0.9161	0.8973	0.9168	0.8971	0.8516	0.8062
	EPS-AD (w/ perturbation)	**0.9826**	**0.9785**	**0.9833**	**0.9746**	**0.9628**	**0.9539**
ImageNet	EPS-AD (w/o perturbation)	0.9173	0.8864	0.9176	0.9141	0.8723	0.9185
	EPS-AD (w/ perturbation)	**0.9633**	**0.9458**	**0.9635**	**0.9619**	**0.9357**	**0.9661**

G MORE RESULTS OF ADVERSARIAL DETECTION

To further evaluate the effectiveness of our proposed EPS-AD, in subsection G.1, we conduct more comparison experiments on detecting more adversarial attacks on ImageNet and CIFAR-10 datasets. To demonstrate the generalization of EPS-AD, we conduct more experiments on detecting unseen attacks and transferable attacks. In subsection G.2, we additionally study the mechanism of EPS-AD by reporting the impact of time steps, set size, low attack intensity, transferability across datasets and a real application detecting face anti-spoofing samples. We provide extra results of all compared experiments in Table 6, 7, 8 and Figure 5, 6, 8, as supplements of additional attack methods corresponding to the main body.

G.1 MORE COMPARISON EXPERIMENTS

More comparison results on basic setup. In Table 6, we provide additional attack results including MIM, TIM, DI, MIM, PGD-ℓ_2, MM on CIFAR-10 and ImageNet. We observe that EPS-AD keeps dominant position under these attacks, EPS-N and S-N exhibit poor performance on ImageNet due to the fact that these two methods use the norm of vectors and thus overlook the rich information that can be deviated from the vector. Moreover, we find that in Figure 5 and Figure 6 , most adversarial detection methods suffer from the extremely low attack intensity (*e.g.*, $\epsilon = 1/255$). In contrast, our method EPS-AD still has promising detection performance (refer to Table 14 for more quantitative results).

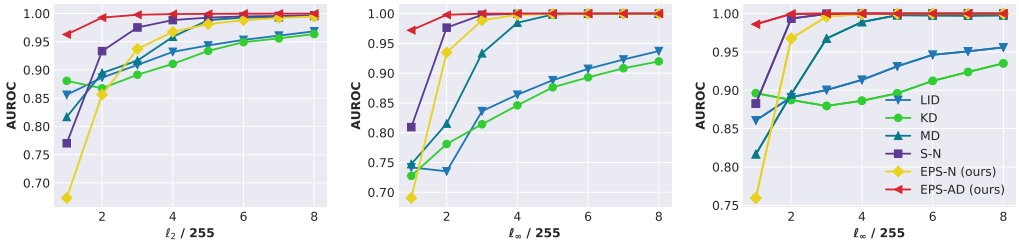
Table 6: More results of different adversarial detection methods on CIFAR-10 and ImageNet in terms of AUROC under $\epsilon = 4/255$ (MIM, TIM, DI, MIM, PGD-ℓ_2, MM).

Dataset	Method	MIM	TIM	DI_MIM	PGD-ℓ_2	MM
CIFAR-10	KD	0.8863	0.8856	0.8459	0.9296	0.9116
	LID	0.9135	0.8781	0.8638	0.9426	0.9380
	MD	0.9890	0.9998	0.9846	0.9958	0.9836
	S-N	0.9993	0.9985	0.9980	**1.0000**	0.9994
	EPS-N (Ours)	0.9999	0.9996	0.9996	**1.0000**	0.9996
	EPS-AD (Ours)	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
ImageNet	KD	0.9664	0.9317	0.7402	0.9809	0.9377
	LID	0.9665	0.9397	0.7658	0.9833	0.9575
	MD	0.9664	0.9365	0.8612	0.9858	0.9585
	LiBRe	0.9483	0.8772	0.9968	0.9532	0.8697
	S-N	0.9087	0.8310	0.9143	0.9505	0.9090
	EPS-N (Ours)	0.9778	0.9114	0.9779	0.9989	0.9939
	EPS-AD (Ours)	**1.0000**	**0.9997**	**1.0000**	**1.0000**	**1.0000**

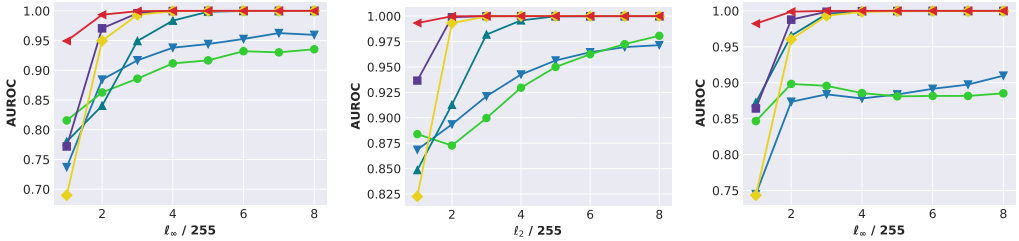
More comparison results on unseen and transferable attacks. We also compare our EPS-AD with KD, LID MD and LiBRe under 5 additional unseen or transferable attacks (MIM, TIM, DI, MIM, PGD-ℓ_2, MM) to further evaluate the effectiveness of our method. In Table 7 and Table 8, Our approach consistently exhibits superior generalization compared to other baselines.

More comparison results on CIFAR-10 over robust WideResNet-70-16 . We further compare our method with baselines on an adversarial trained classifier on CIFAR-10, *e.g.*, WideResNet-70-16 (Gowal et al., 2021) against various attacks. In Table 9, 10, We observe that diffusion-based detection methods are much better than other baselines trained with specific adversarial samples. One possible reason is that adversarial samples are difficult to deceive robust classifiers, which means that such adversarial samples are ineffective for training effective detectors.

More comparison results on ImageNet over DeiT-S. We further make an attempt on Vision-transformer-based architecture (*i.e.* DeiT-S (Dosovitskiy et al., 2021)) on ImageNet. Considering the specificity of the vision-transformer-based structure, we compare our method to three baselines

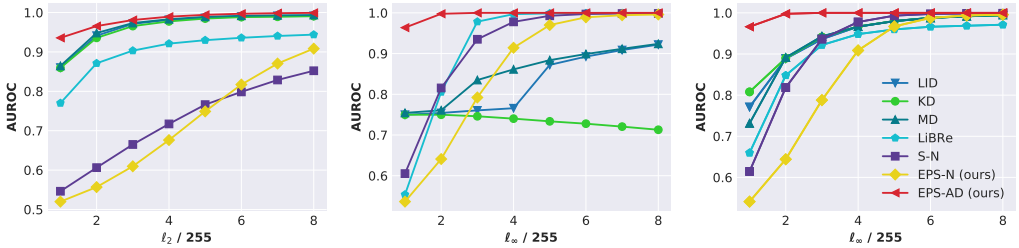


(a) Attack Method: BIM-L2 (b) Attack Method: DI-MIM (c) Attack Method: MIM

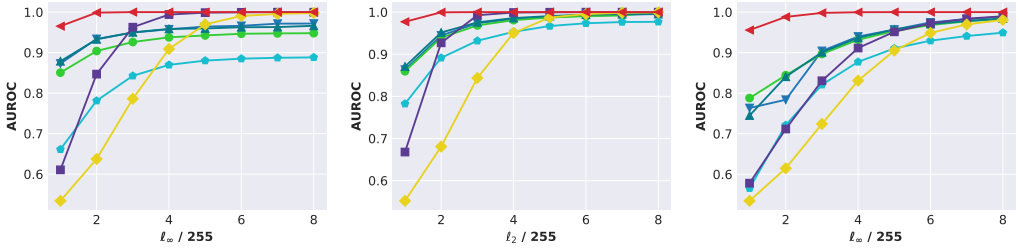


(d) Attack Method: MM-Attack (e) Attack Method: PGD-ℓ_2 (f) Attack Method: TIM

Figure 5: More Results of adversarial detection on CIFAR-10. Sub-figures (a) - (f) report the AUROC on different attacks under $\epsilon \in [1/255, 8/255]$ and share the same legend in sub-figure (c).



(a) Attack Method: BIM-L2 (b) Attack Method: DI-MIM (c) Attack Method: MIM



(d) Attack Method: MM-Attack (e) Attack Method: PGD-ℓ_2 (f) Attack Method: TIM

Figure 6: More Results of adversarial detection on ImageNet. Sub-figures (a) - (f) report the AUROC on different attacks under $\epsilon \in [1/255, 8/255]$ and share the same legend in sub-figure (c).

(LID, S-N and EPS-N). From Table 11, our method exhibits consistent superiority when compared to other baselines, suggesting the versatility of EPS-AD with different architectures.

Table 7: More results of AUROC for detecting the unseen attacks (MIM, TIM, DI, MIM, PGD-ℓ_2, MM) on CIFAR-10. "FGSM (seen)" denotes the seen adversarial attack used for the training of KD, LID and MD.

Method	FGSM(seen)	MIM	TIM	DI_MIM	PGD-ℓ_2	MM
KD	0.9213	0.8865	0.8874	0.8546	0.9302	0.9112
LID	0.9236	0.9071	0.8410	0.8701	0.9076	0.9098
MD	0.9990	0.9855	0.9995	0.9789	0.9952	0.9826
S-N	**1.0000**	0.9993	0.9985	0.9980	**1.0000**	0.9994
EPS-N (Ours)	**1.0000**	0.9999	0.9996	0.9996	**1.0000**	0.9996
EPS-AD (Ours)	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**

Table 8: More results of AUROC for detecting the transferable attacks (MIM, TIM, DI, MIM, PGD-ℓ_2, MM) on ImageNet, where KD, LID, MD and LiBRe are trained with adversarial examples with ResNet-50 but detect the adversarial examples crafted with ResNet-101.

Method	MIM	TIM	DI_MIM	PGD-ℓ_2	MM
KD	0.6355	0.7006	0.7558	0.7028	0.7381
LID	0.7780	0.7978	0.7654	0.7734	0.7913
MD	0.7756	0.7612	0.8395	0.7827	0.7864
*L*iBRe	0.8966	0.7317	0.9722	0.8749	0.8388
S-N	0.6584	0.6177	0.6620	0.6819	0.6512
EPS-N (Ours)	0.9985	0.9888	0.9983	0.9998	0.9996
EPS-AD (Ours)	**1.0000**	**0.9996**	**1.0000**	**1.0000**	**1.0000**

G.2 More Discussions of EPS-AD

Ablation of timestep. The ablation study of timestep is provided in Table 12. We observe that when the timestep T stays in $[10, 100]$, EPS-AD and EPS-N both obtain usable AUROC in detecting FGSM-ℓ_2 attack, which verifies that our method is insensitive to the timestep. Moreover, Our approach achieves optimum performance when $T^* = 20$ on CIFAR-10 and $T^* = 50$ on ImageNet.

Impact of set size. Previous adversarial detection methods usually measure the discrepancy well only with large amount of data (Gao et al., 2021). To show the effectiveness of our proposed EPS-AD, in this experiment, we further ablate the effect of set size by conducting experiments on 100 to 500 samples subset of CIFAR-10 with WideResNet-28-10 and ImageNet with ResNet-50. Performance of our three methods, S-N, EPS-N and EPS-AD, is shown in Table 13 and Figure 8. It is obvious that EPS-AD consistently outperforms EPS-N and S-N with small set size and large set size. Moreover, EPS-AD is robust to the changes of set size while EPS-N and S-N fluctuate with set size, especially on CIFAR-10 dataset.

Detecting on low attack intensity. To further reveal the superiority of our EPS-AD, we conduct an experiment under an extremely low attack intensity (*e.g.*, $\epsilon = 1/255$) on ImageNet. In Table 14, we observe that Our EPS-AD achieves a significant advantage in detecting adversarial samples crafted with extremely low attack intensity, demonstrating its significant effectiveness.

Detecting on adversarial samples across datasets. We further exploit the transferability across different datasets. To this end, we utilize a pre-trained score-based diffusion model on ImageNet to perform detecting adversarial samples from CIFAR-10. Specifically, we randomly select two disjoint subsets as adversarial and natural samples (each containing 500 samples) from CIFAR-10 and use a score model pre-trained on ImageNet to calculate the AUROC, which is named EPS-AD*. Table 15 demonstrates detection performance of 6 methods against 12 attacks under $\epsilon = 2/255$ on CIFAR-10 over WideResNet-28-10. We observe that EPS-AD* still exhibits superior detection performance compared to KD, LID, MD baselines, and achieves a comparable performance compared to other diffusion-based methods that use the score model pre-trained on CIFAR-10.

Table 9: Comparison of AUROC for using adversarial trained WideResNet-70-16 as classifier on CIFAR-10 under $\epsilon = 2/255$. Due to the constraint of memory and resources, we omit the detection results on AutoAttack for KD, LID and MD.

Method	KD	LID	MD	S-N	EPS-N (Ours)	EPS-AD (Ours)
FGSM	0.5852	0.7551	0.6924	0.9797	0.9976	**1.0000**
PGD	0.5672	0.7517	0.6846	0.9625	0.9954	**1.0000**
BIM	0.5786	0.7543	0.6787	0.9568	0.9935	**1.0000**
MIM	0.5795	0.7544	0.6804	0.9679	0.9963	**1.0000**
TIM	0.5812	0.7533	0.6783	0.8468	0.9506	**0.9997**
CW	0.5559	0.7511	0.6837	0.9600	0.9949	**1.0000**
DI_MIM	0.5763	0.7505	0.6712	0.8877	0.9728	**0.9999**
PGD-ℓ_2	0.6116	0.7632	0.7049	0.9942	0.9994	**1.0000**
FGSM-ℓ_2	0.6114	0.7619	0.7032	0.9471	0.9766	**0.9999**
BIM-ℓ_2	0.7550	0.7629	0.7060	0.9396	0.9756	**0.9999**
MM	0.6030	0.5103	0.5065	0.5149	0.5222	**0.9840**
AA	—	—	—	0.5151	0.5242	**0.9847**

Table 10: Comparison of AUROC for using adversarial trained WideResNet-70-16 as classifier on CIFAR-10 under $\epsilon = 4/255$.

Method	KD	LID	MD	S-N	EPS-N (Ours)	EPS-AD (Ours)
FGSM	0.6020	0.7628	0.7668	0.9999	**1.0000**	**1.0000**
PGD	0.5913	0.7598	0.7535	0.9998	**1.0000**	**1.0000**
BIM	0.6076	0.7617	0.7588	0.9994	**1.0000**	**1.0000**
MIM	0.7683	0.7625	0.7601	0.9997	**1.0000**	**1.0000**
TIM	0.6029	0.7605	0.7563	0.9917	0.9984	**1.0000**
CW	0.5919	0.7581	0.7524	0.9998	**1.0000**	**1.0000**
DI_MIM	0.6015	0.7562	0.7551	0.9983	0.9998	**1.0000**
PGD-ℓ_2	0.8314	0.7774	0.7544	**1.0000**	**1.0000**	**1.0000**
FGSM-ℓ_2	0.7805	0.7709	0.7489	0.9803	0.9902	**0.9999**
BIM-ℓ_2	0.8351	0.7739	0.8141	0.9835	0.9927	**1.0000**
MM	0.7475	0.6925	0.5090	0.5387	0.5451	**0.9844**
AA	—	—	—	0.5387	0.5455	**0.9854**

Detecting on face anti-spoofing samples. We further consider a real application scenario on detecting face on anti-spoofing protocol 1 of OULU-NPU data set (Boulkenafet et al., 2017) (Figure 7) with a score model pre-trained on ImageNet. Specifically, we randomly select two subsets as spoofing and anti-spoofing samples (each containing 500 samples) from the data set to specify a threshold so that a high fraction of samples are correctly classified; then we use remaining samples to test the *APCER*, *BPCER* and *ACER* value, where *APCER* $= FP/(TN+FP)$, *BPCER* $= FN/(TP+FN)$ and *ACER* $= (APCER+BPCER)/2$. In Table 16, our EPS-AD outperforms 5 baseline methods and shows comparable performance with the SOTA method PatchNet (Wang et al., 2022) even if we do not use the face data to train the score model, which demonstrates the versatility of our method.

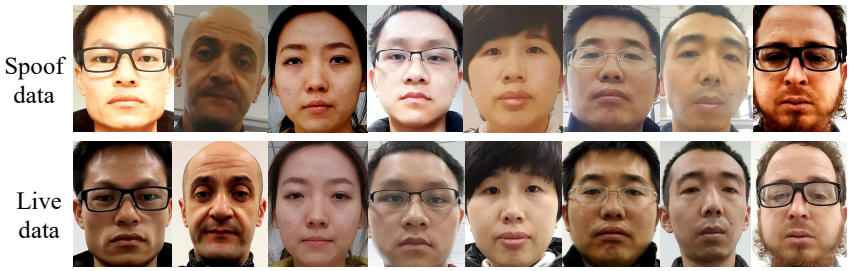


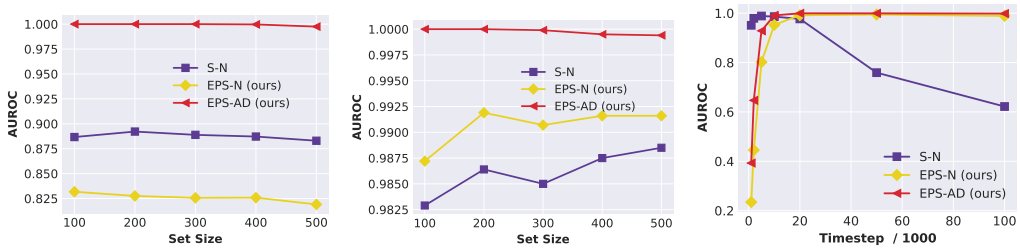
Figure 7: Some examples of OULU-NPU data set.

Table 11: Comparison of AUROC for using DeiT-S as classifier on ImageNet under $\epsilon = 4/255$.

Method	LID	S-N	EPS-N (Ours)	EPS-AD (Ours)
FGSM	0.8846	0.9789	0.9984	**1.0000**
PGD	0.9162	0.8935	0.9969	**1.0000**
BIM	0.9191	0.7331	0.9215	**0.9998**
MIM	0.9102	0.9025	0.9780	**1.0000**
TIM	0.9019	0.8091	0.8765	**0.9991**
CW	0.8742	0.8934	0.9975	**1.0000**
DI_MIM	0.7246	0.9074	0.9752	**1.0000**
PGD-ℓ_2	0.9041	0.9451	0.9987	**1.0000**
FGSM-ℓ_2	0.8698	0.7665	0.7023	**0.9874**
BIM-ℓ_2	0.9002	0.6564	0.6544	**0.9791**
MM	0.9164	0.8902	0.9886	**0.9992**
AA	0.9191	0.9023	0.9915	**1.0000**

Table 12: Impact of timestep with WideResNet-28-10 on CIFAR-10 and ResNet-50 on ImageNet against FGSM-ℓ_2 under $\epsilon = 4/255$.

Dataset	Method \ timestep	1	2	5	10	20	50	100
CIFAR-10	S-N	0.9508	0.9792	0.9885	0.9860	0.9768	0.7590	0.6218
	EPS-N (Ours)	0.2346	0.4454	0.8018	0.9526	0.9916	0.9915	0.9889
	EPS-AD (Ours)	0.3927	0.6468	0.9287	0.9911	**0.9994**	0.9993	0.9983
ImageNet	S-N	0.4794	0.5298	0.7168	0.8528	0.8830	0.7309	0.5964
	EPS-N (Ours)	0.2246	0.2466	0.4369	0.6215	0.7484	0.8191	0.8190
	EPS-AD (Ours)	0.7348	0.8926	0.9864	0.9960	0.9974	**0.9975**	0.9965



(a) Impact of set size on ImageNet (b) Impact of set size on CIFAR-10 (c) Impact of timestep

Figure 8: Impact of different set sizes and diffusion time step. Sub-figures (a) and (b) report the AUROC on FGSM-ℓ_2 attack under $\epsilon = 4/255$. Sub-figure (c) reports the AUROCs of different diffusion time in $\{1, 2, 5, 10, 20, 50, 100\}$ on CIFAR-10 dataset.

Table 13: Impact of data set size with WideResNet-28-10 on CIFAR-10 and ResNet-50 on ImageNet against FGSM-ℓ_2 under $\epsilon = 4/255$.

Dataset	Method \ size	100	200	300	400	500
CIFAR-10	S-N	0.9829	0.9864	0.9850	0.9875	0.9885
	EPS-N (Ours)	0.9872	0.9919	0.9907	0.9916	0.9916
	EPS-AD (Ours)	1.0000	1.0000	0.9999	0.9995	0.9994
ImageNet	S-N	0.8867	0.8922	0.8889	0.8872	0.8830
	EPS-N (Ours)	0.8318	0.8276	0.8257	0.8259	0.8191
	EPS-AD (Ours)	1.0000	1.0000	0.9997	1.0000	0.9975

Table 14: Comparision of different adversarial detection methods with attack intensity $\epsilon = 1/255$ over ResNet-50 on ImageNet.

Method	KD	LID	MD	LiBRe	S-N	EPS-N (Ours)	EPS-AD (Ours)
FGSM	0.7301	0.7765	0.7694	0.5653	0.5654	0.6942	**0.9839**
PGD	0.8167	0.7651	0.8116	0.6938	0.5348	0.6149	**0.9633**
BIM	0.8313	0.7711	0.8303	0.7154	0.5240	0.5619	**0.9458**
MIM	0.8079	0.7711	0.7313	0.6597	0.5411	0.6144	**0.9662**
TIM	0.7879	0.7634	0.7449	0.5655	0.5344	0.5780	**0.9556**
CW	0.8161	0.7690	0.8336	0.6996	0.5345	0.6169	**0.9635**
DI_MIM	0.7498	0.7498	0.7543	0.5538	0.5364	0.6052	**0.9637**
PGD-ℓ_2	0.8597	0.8633	0.8707	0.7824	0.5519	0.6678	**0.9770**
FGSM-ℓ_2	0.7265	0.7737	0.7704	0.5727	0.5546	0.5978	**0.9619**
BIM-ℓ_2	0.8596	0.8602	0.8644	0.7701	0.5199	0.5462	**0.9357**
MM	0.8505	0.8730	0.8786	0.6611	0.5340	0.6106	**0.9652**
AA	0.8535	0.8706	0.8733	0.6629	0.5351	0.6113	**0.9661**

Table 15: Comparison of cross-dataset EPS-AD* under attack intensity $\epsilon = 2/255$ with other methods over WideResNet-28-10 on CIFAR-10, where EPS-AD* utilizes a score model pre-trained on ImageNet.

Method	KD	LID	MD	S-N	EPS-N (Ours)	EPS-AD	EPS-AD*
PGD	0.8871	0.8836	0.8815	0.9679	0.9950	0.9997	0.9539
FGSM	0.9112	0.9010	0.9200	0.9902	0.9987	0.9999	0.9612
BIM	0.8786	0.8878	0.8811	0.9268	0.9811	0.9987	0.9364
MIM	0.8873	0.8909	0.8947	0.9676	0.9935	0.9996	0.9476
TIM	0.8983	0.8735	0.9655	0.9603	0.9878	0.9989	0.9370
CW	0.8634	0.8762	0.8968	0.9682	0.9953	0.9997	0.9450
DI_MIM	0.7810	0.7351	0.8153	0.9348	0.9764	0.9979	0.9527
PGD-ℓ_2	0.8727	0.8935	0.9128	0.9931	0.9989	0.9999	0.9652
FGSM-ℓ_2	0.9044	0.8927	0.9211	0.9166	0.9634	0.9958	0.9399
BIM-ℓ_2	0.8675	0.8864	0.8946	0.8564	0.9333	0.9927	0.9257
MM	0.8627	0.8843	0.8404	0.9497	0.9706	0.9935	0.9486
AA	0.8754	0.8894	0.8392	0.9595	0.9775	0.9947	0.9471

Table 16: Comparison of our EPS-AD using a score model pre-trained on ImageNet with the SOTA methods when detecting Face anti-spoofing on OULU-NPU P1 (Boulkenafet et al., 2017), where these results are from (Wang et al., 2022).

Method	APCER(↓ %)	BPCER(↓%)	ACER(↓%)
Disentangle (Zhang et al., 2020)	1.7	0.8	1.3
SpoofTrace (Liu et al., 2020b)	0.8	1.3	1.1
BCN (Yu et al., 2020a)	0.0	1.6	0.8
CDCN (Yu et al., 2020c)	0.4	1.7	1.0
NAS-FAS (Yu et al., 2020b)	0.4	0.0	0.2
PatchNet (Wang et al., 2022)	0.0	0.0	0.0
EPS-AD	**0.0**	**0.2**	**0.1**