ADVERSARIAL ATTACK DETECTION UNDER REALISTIC CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

While adversarial attacks are a serious threat for neural networks safety, existing defense mechanisms remain very limited regarding their applicability to real-world settings. Any industrial-driven attack detector is expected to meet three unavoidable requirements: **(R1)** being adapted to black-box scenario where the user has only access to the predicted probabilities, **(R2)** making fast inference and **(R3)** not involving any training phase. In this paper, we introduce `REFEREE`, the first detector that meets all these requirements while improving state-of-the-art performances. It leverages the concept of information projections (I-projection), which generalizes ideas coming from out-of-distribution detection and allows to extract relevant information contained in the softmax outputs of a network. Our extensive experiments demonstrates that `REFEREE` improves upon existing methods while considerably reducing the inference time: it requires less than 0.05 seconds by test input, which is up to 400 times faster than former methods. This makes `REFEREE` an excellent candidate for adversarial attacks detection in real-world applications.

1 INTRODUCTION

Advanced Deep Learning (DL) techniques have made significant improvements over previous state-of-the-art methods in computer vision. The rise of highly scalable architectures and training techniques has fueled their wide adoption in the industry. However, the impressive performances of deep neural networks often hide many failures regarding their resilience and reliability (Hendrycks et al., 2021), which is an obvious obstacle to their adoption for high-risk applications such as face recognition (Grother et al., 2014; 2018; 2019) or autonomous vehicles (Bojarski et al., 2016). This paper focuses on a specific safety issue: adversarial attacks. The latter refer to the design of malicious attackers able to craft samples that fool a given classifier. This is typically done by adding small additive perturbations to real-world examples, which are indistinguishable to human eyes but highly disruptive to network predictions.

The design of efficient attacks has resulted in a vast literature in the field of computer vision (starting with the seminal work of Szegedy et al. (2013)), but fewer works have focused on building appropriate defense mechanisms against these attacks. Protection techniques can be divided into two main tendencies, depending on whether the practitioner is intervening during the training phase or on an already deployed system. The first line of work can be assimilated to robust training techniques which consist in incorporating regularization terms that are smoothing the variability of predictions (Madry et al., 2018; Zhang et al., 2019; Carmon et al., 2019). However, this approach contains two important limitations: *(i)* in many situation, it makes the training phase unstable and *(ii)* it is not able to anticipate for future attack mechanisms which could fool the system.

The second line of work corresponds to the design of *detectors* that are able to decide, based on an already existing system, whether an input sample is a malicious attack or not. This paradigm is appealing because it does not require any change during the learning phase, making it ready-to-use for an already deployed system. However, existing methods fail to meet the requirements of real-life scenarii which can be summarized into three points.

- **(R1) Black-box scenario.** Systems already deployed in production are generally opaque to the end user, who only has access to the softmax predictions of the networks.

(R2) Low resources / computation time. In many real-world applications, AI systems are making real-time predictions at a high frequency (*e.g.* face recognition for airport security). As a result, any relevant detector should have a low inference time and require low computation resources.

(R3) No oracle on the nature of the attackers. Any relevant detector should be *unsupervized*, meaning it should not require any training phase with access to attack examples. Indeed, the landscape of existing attackers is moving fast, making the availability of adversarial examples not realistic in practice.

CONTRIBUTIONS

In this paper, we introduce the first efficient adversarial attacks detector that meets all requirements **(R1)** - **(R2)** - **(R3)** of real-life applications of computer vision. In words, our detector is only based on the sotftmax predictions of the network, makes fast predictions and is unsupervised. In addition, it improves upon existing state-of-the-art detection methods, as can be visually checked in Fig. 1. In order to ensure fair comparison with previous works, we conduct extensive experiments on various datasets (*i.e.*, CIFAR10, CIFAR100 and Tiny ImageNet) and various attack mechanisms.

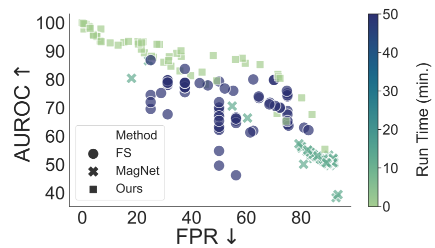


Figure 1: Performances versus testing time.

Experimental setting. We choose to evaluate our detectors on vision transformers (Dosovitskiy et al., 2021; Tolstikhin et al., 2021; Steiner et al., 2021; Chen et al., 2021; Zhai et al., 2022), contrarily to previous works on adversarial attack detection that rely on ResNets (Kherchouche et al., 2020; Xu et al., 2018; Meng & Chen, 2017; Ma et al., 2018; Feinman et al., 2017). This choice is motivated by the fact that transformers have achieved state-of-the-art results in several tasks (*e.g.*, image generation (Parmar et al., 2018), image classification (Wang et al., 2021) and image segmentation (Zheng et al., 2021)). Our extensive experiments on CIFAR10 (Krizhevsky, 2009), CIFAR100 (Krizhevsky et al.) and TinyImageNet (Jiao et al., 2019) demonstrate the superiority of `REFEREE` over existing methods.

Paper organisation. In Sec. 2, we formalise the adversarial attack detection problem and discuss related works and their limitations. Then, motivated by Information Theory considerations, we introduce our detector in Sec. 3. We present insights about our proposed detector in Sec. 4, while Sec. 5 is dedicated to the presentation and analysis of our extensive experiments. Finally, in Sec. 6, we provide concluding remarks.

2 FRAMEWORK AND RELATED WORK

2.1 PROBLEM FORMULATION

Computer vision classification. We are considering classification problems for computer vision applications and will denote by $\mathcal{X}$ the image input space and $\mathcal{Y} = \{1, \ldots, C\}$ the target space made of $C > 1$ classes. A given training set can be described as a set of i.i.d. realizations $\mathcal{D}_{\text{train}}\{(\mathbf{x}^k, y^k)\}_{k=1}^n$ of a given pair of random variables $(\mathbf{X}, Y)$ taking its values in $\mathcal{X} \times \mathcal{Y}$. We will denote by p_{XY} the p.d.f. of $(\mathbf{X}, Y)$.

Transformers and softmax-based decision. A given Vision Transformer (ViT) f_θ, parametrized by $\theta \in \Theta$, processes a given image $\mathbf{x}$ through consecutive layers in order to extract relevant information. After the application of a softmax, the last layer $\{q_\theta(i|\mathbf{x})\}_{i=1}^C$ corresponds to the probabilities of class membership of $\mathbf{x}$. The final decision of the Transformer is then $f_\theta(\mathbf{x}) = \underset{i \in \mathcal{Y}}{\operatorname{argmax}}\ q_\theta(i|\mathbf{x})$.

Adversarial attacks. The goal of an attacker consists in finding, from a standard input $\mathbf{x}$, a deformation $\mathbf{x}'$ close to $\mathbf{x}$ but leading to a change in the network prediction. Formally, one tries to solve an optimization problem (Szegedy et al., 2013) of the following form $\mathbf{x}' \in \operatorname{argmin}_{\mathbf{w}}\ d(\mathbf{w}, \mathbf{x})$, where d is a distance on $\mathcal{X}$ and under the constraint that $f_\theta(\mathbf{w}) \neq f_\theta(\mathbf{x})$ and that $\mathbf{w}$ remains an image.

Detection of adversarial attacks. The goal of an adversarial attack detector is to predict whether a new input $\mathbf{x}$ is regular or has been crafted by a malicious adversary. In full generality, it first computes an anomaly score $s(\mathbf{x})$ based on $\mathbf{x}$ and/or any of its transformations through the network f_θ. Then, depending on the magnitude of this score, the sample $\mathbf{x}$ is deemed regular or not. Denoting the detector by d, the final decision takes the following form, for a given threshold γ:

$$d(\mathbf{x}) = \mathbf{1}_{s(\mathbf{x})\geq\gamma} = \begin{cases} 1 & \text{if } s(\mathbf{x}) \geq \gamma, \\ 0 & \text{if } s(\mathbf{x}) < \gamma. \end{cases} \tag{1}$$

2.2 Existing adversarial detection methods

Let us review existing defense mechanisms that exist to protect neural networks against adversarial attacks in the context of computer vision. We will divide our review into two paragraphs whether the detector satisfy requirement **(R3)** or not, that is whether attacked training data are required to train the detector or not.

Supervised methods – not satisfying (R3). Supervised methods usually consist in training simple machine learning algorithms, such as SVMs or logistic regressions, to discriminate adversarial examples from natural ones, using examples from both classes. The features used for training these machine learning models can be extracted from the networks layers using directly the samples (Lu et al., 2017; Carrara et al., 2018; Metzen et al., 2017), or pre-process them using kernel density estimation or uncertainty measure (Feinman et al., 2017), computer vision specific characteristics such as natural scene statistics (Kherchouche et al., 2020), PCA (Li & Li, 2017) or also local intrinsic dimensionality (Ma et al., 2018). Regarding **(R2)**, one can arguably say that these methods are satisfying as the inference time of simple machine learning models is fast. However, as they do not satisfy **(R3)**, these methods need to make some assumptions on the nature of adversarial attacks to generate malicious samples, at the risk of overfitting and misgeneralizing. Moreover, most of these methods rely on the hidden layers of the networks, which makes them unrealistic for practical black-box applications **(R1)** where only softmax are available.

Unsupervised methods – satisying (R3). Unsupervised methods only rely on clean samples to build a detector, making them very attractive for real-life applications. Let us explore existing works in light of requirements **(R1)** and **(R2)**. Some detectors require access to intermediate layers representation (Ma et al., 2019; Sotgiu et al., 2020; Zheng & Hong, 2018; Aldahdooh et al., 2021), which makes them unsuitable for use in the context of **(R1)**. Two methods satisfy **(R1)** but are arguably less effective regarding **(R2)**: the Feature Squeezing (FS) of Xu et al. (2018) and the MagNet detector of Meng & Chen (2017) which relies on a denoising autoencoder. We will discuss these two methods in further details in Sec. 3.3 as we include them into our experimental setting. Let us also mention JTLA Raghuram et al. (2021), a refinement of FS which unfortunately does not satisfy **(R1)**.

Table 1: Summary of Detector's requirements meets

Detector	(R1)	(R2)	(R3)
Ma et al. (2019)	✗	✗	✓
Sotgiu et al. (2020)	✗	✗	✓
Zheng & Hong (2018)	✗	✗	✓
Aldahdooh et al. (2021)	✗	✗	✓
Xu et al. (2018)	✓	✗	✓
Meng & Chen (2017)	✓	✗	✓
Raghuram et al. (2021)	✗	✗	✓

2.3 Out-of-Distribution detection methods

Adversarial attack detection can be considered as an extreme case of the out-of-distribution (OOD) detection problem. The latter has received much attention from the ML/DL community and many techniques have been developed. In particular, a line of methods is based on the extraction of relevant information from the softmax probabilities, making them very attractive for our purpose. This line of works has been lauched by the seminal work of Hendrycks & Gimpel (2016) who proposed to focus on the Maximum Softmax Probability (MSP) to discriminate between in- and out-of-distribution samples. The underlying idea of MSP is that the more spiky the probabilities, the more confident the network is and therefore the cleaner the input. Let us also mention the DOCTOR detector, recently introduced by Granese et al. (2021), which computes the Gini coefficient of the softmax probabilities. Both methods satisfy the three requirements **(R1)** - **(R2)** - **(R3)** and shall be used as baselines in our experiments.

3 REFEREE: AN EFFICIENT, REAL-LIFE ADAPTED ADVERSARIAL DETECTOR

3.1 AN INFORMATION THEORETIC VIEW ON SOFTMAX-BASED DETECTION METHODS

Both previously mentioned methods MSP and DOCTOR make the assumption that *softmax probabilities contain relevant information regarding the input under consideration*. We think this hypothesis is relevant and introduce a softmax-based detector that is able to improve the state-of-the-art performances regarding the detection of adversarial attacks. Our idea is based on a quite simple interrogation: do existing methods leverage full information from the softmax probabilities? Information Theoretic reasoning is helpful to investigate this question as it provides many tools to measure how much a discrete probability distribution differs from a fixed reference (see for instance Basseville (2013)). The resulting notions of *divergence* between probability distributions has been extensively used by the machine learning community (Li & Turner, 2016). The most famous are probably the Bregman, Rényi and Chernoff divergences (Bregman, 1967; Rényi et al., 1961; Basu et al., 1998; Chernoff et al., 1952), which are specific instantiations of the family of f-divergences (Csiszár, 1967). In this paper, we will focus on the fruitful notion of Tsallis-α divergence.

Definition (Tsallis-α divergence). Let $C \geq 1$. Let $\mathbf{p} = (p_i)_{i=1}^C$ and $\mathbf{q} = (q_i)_{i=1}^C$ be two discrete probability distributions. Let $\alpha \in \mathbb{R} \setminus \{1\}$. The Tsallis-$\alpha$ divergence $T_\alpha(\mathbf{p} \,||\, \mathbf{q})$ between $\mathbf{p}$ and $\mathbf{q}$ is defined by

$$T_\alpha\left(\mathbf{p} \,||\, \mathbf{q}\right) := \frac{\text{sign}(\alpha)}{\alpha - 1}\left[\left(\sum_{i=1}^{C} p_i^\alpha \times q_i^{1-\alpha}\right) - 1\right]. \tag{2}$$

When $\alpha \to 1$, the definition extends by taking the natural limit, which leads to the usual Kullback-Leibler divergence: $T_1\left(\mathbf{p} \,||\, \mathbf{q}\right) = D_{\text{KL}}(\mathbf{p} \,||\, \mathbf{q})$.

This notion of Tsallis-α divergence has many links with other notions of entropy (see Villmann & Haase (2010)). Moreover, it offers a quite natural generalisation of both the DOCTOR and MSP detector as stated in the following proposition, which follows from elementary computations.

Proposition 1. Let $\mathbf{u}$ be the uniform distribution.

1. When $\alpha = -1$, $T_{-1}(\mathbf{u} \,||\, \mathbf{q})$ is proportional to the Gini coefficient of $\mathbf{q}$. As a result, $T_{-1}(\mathbf{u} \,||\, \mathbf{q})$ corresponds to the DOCTOR score.
2. It holds that $\lim\limits_{\alpha \to -\infty} \frac{1}{\alpha-1} \ln\left[(\alpha - 1)T_\alpha(\mathbf{u}||\mathbf{q}) + 1\right] = \ln \max\limits_i \frac{q_i}{u_i}$. Otherwise saying, the asymptotic behavior of the Tsallis-α divergence is governed by the MSP score.

3.2 REFEREE: OUR INFORMATION THEORETIC DETECTOR

General observations. Notice that MSP and DOCTOR are not assuming knowledge on the training data distribution, while the latter is usually at the disposal of the practitioner. This missing information should however be instrumental to discriminate between a clean and a malicious sample. A refinement of DOCTOR could be to replace the uniform distribution by the empirical frequencies of each class. Still, this aggregated version of the training distribution would not be able to capture all existing attack mechanisms. Let us be more precise. A typical, in-distribution softmax probability is spiked on the predicted class and a malicious one deviates from this typical behavior. However, this deviation can take two opposite forms: it can be "over-spiked" on a given class or it can be "over-smoothed" (see Figure 3 for an illustration of these concepts). The DOCTOR detector assumes that any deviation is of the second form: the more close it is to the uniform distribution, the more likely it is malicious. But this completely misses the first type.

Projection onto the training manifolds of softmax probabilities. Instead of aggregating over the training distribution, we will leverage the full information it contains at the level of the softmax probabilities. More precisely, given the training reference of images $\mathcal{D}_{\text{train}} = \{(\mathbf{x}^k, y^k)\}_{k=1}^{n}$, our detector computes the distance to the softmax probabilities associated to $\mathcal{D}_{\text{train}}$. Formally, the anomaly score $s_{\texttt{REFEREE}}$ of a test input $\mathbf{x}$ is defined by the following formula:

$$s_{\texttt{REFEREE}}(\mathbf{x}) = \min_{\mathbf{x^k} \in \mathcal{D}_{\text{train}}} T_\alpha\left(q_\theta(\cdot\,|\,\mathbf{x}^k)\,||\,q_\theta(\cdot\,|\,\mathbf{x})\right). \tag{3}$$

Then, the decision is taken as in Eq. 1. Our detector can therefore be divided into three steps:

REFEREE in a nutshell

1. (*Offline*) Collect the softmax probabilities of the training set $\{q_\theta(\cdot\,|\,\mathbf{x}^k)\}_{k=1}^{n}$.
2. (*Online*) For a given test input $\mathbf{x}$:
   (a) Compute the anomaly score $s_{\texttt{REFEREE}}(\mathbf{x})$,
   (b) Threshold the score: $d_{\texttt{REFEREE}}(\mathbf{x}) = \mathbf{1}_{s_{\texttt{REFEREE}}(\mathbf{x}) \geq \gamma}$.

Remark (Hyperparameters of our detector). Our detector possesses two hyperparameters: α, which controls the amount of distortion incorporated in the divergence computation T_α, and the threshold γ. In Sec. 5.3, we will discuss the choice of α. Regarding γ, a typical way to select it is to use the training set and select a proportion of "outliers" (*e.g.* by relying on a notion of data depth Tukey (1975)) and trying to detect them with REFEREE.

Notice that Eq. 3 is reminiscent of the notion of Information Projection introduced by Kullback (1997); Csiszár (1975; 1984). It finds numerous applications, for instance in statistical physics (Jaynes, 1957) or in large deviation theory (Sanov, 1958). The basic idea behind REFEREE is that a malicious sample lies outside the training manifold, at the level of the softmax probabilities. Indeed, when $n \to +\infty$, if $\mathcal{M}_{\text{train}}$ denote the limiting set of $\mathcal{D}_{\text{train}}$, the limiting form of Eq. 3 is the Information Projection onto the manifold $\mathcal{M}_{\text{train}}$. As our experiments demonstrate, REFEREE improve the state-of-the-art. We think it is quite remarkable that *the information contained at the level of softmax probabilities is actually sufficient to detect adversarial attacks*. Moreover, this notion of Information Projection offers a very natural interpretation of the score $s_{\texttt{REFEREE}}(\mathbf{x})$ which computes the similarity level between a test input and the training dataset. We investigate this aspect in Sec. 5.4.

3.3 Comparison with existing detectors

As previously announced, we are going to compare REFEREE with FS, MagNet and OOD-detection methods (MSP, ODIN and DOCTOR). Let us provide more details about FS and MagNet.

The Feature Squeezing method (FS; Xu et al. (2018)). FS is an unsupervized parameter-free method that does not involve any training. Given a pre-trained classifier, FS consists of three steps: (i) input feature compression, (ii) prediction extraction, (iii) comparison of the extracted features to the original prediction. The more the predictions differ, the more the sample is likely to be inconsistent. FS requires four different versions of the input: the original input, a low-precision version, a median-filtered version and a denoised version. At test time, the pre-trained classifier is run on all four versions of the input sample. A L_1-distance is then used to compare the predictions. FS requires a GPU to run inference on different inputs and is memory intensive as it requires both all input changes and network predictions, making it difficult to deploy in a real-world scenario.

MagNet Meng & Chen (2017). MagNet is an unsupervised adversarial detection method that involves learning two different components: a detector and a reformer. The role of the detector is to decide whether the input sample is clean, while the reformer finds the closest input on the training manifold. MagNet implements this strategy by relying on two autoencoders trained on clean samples. MagNet is computationally intensive because, during inference, the detector as well as the model must be run. The careful training of the autoencoders is an additional layer of complexity, which makes it difficult to use in practice.

REFEREE does not require any training which makes it easy to use. At the time of testing, given an input sample, REFEREE solely relies on a comparison on the predictions of the softmax method, which requires a calculation for prediction, with a set of pre-computed reference distributions. It is, therefore, *computationally efficient* and makes REFEREE a good fit for real-world scenarios.

4 BENCHMARKING OUR DETECTOR

4.1 EXPERIMENTAL DETAILS

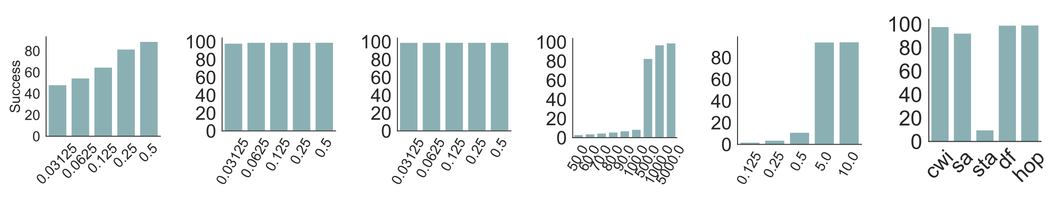
Setting. To benchmark `REFEREE`, we rely on Vision Transformers (He et al., 2016) as they outperforms ResNet models on many vision tasks (Parmar et al., 2018; Wang et al., 2021; He et al., 2021; Dosovitskiy et al., 2021; Steiner et al., 2021; Chen et al., 2021; Tolstikhin et al., 2021; Zhai et al., 2022). We test `REFEREE` on three vision datasets that have been widely used by the vision community: CIFAR10, CIFAR100 (Krizhevsky, 2009) and Tiny ImageNet (Jiao et al., 2019). On CIFAR10 and CIFAR100, we finetune the ViT-based model with 16 layers (85.8 million of parameters)[1] (Dosovitskiy et al., 2021) pretrained on ImageNet (Deng et al., 2009). During finetuning the batch size is set to 512, the learning rate of SGD (Ruder, 2016) is set to 3×10^{-2} and we use 500 warming steps with no gradient accumulation Vaswani et al. (2017). For Tiny ImageNet, we used a ViT with 16 layers, trained by Huynh (2022) and available at `https://github.com/ehuynh1106/TinyImageNet-Transformers`.

Table 2: ViT accuracy

Dataset	Acc (%)
`CIFAR10`	98.7
`CIFAR100`	92.4
`Tiny ImageNet`	86.4

Choice of the Attacks. To benchmark the evaluation methods, we rely on multiple attack mechanisms. First, we consider *Fast Gradient Sign Method (FGSM)* (Goodfellow et al., 2015) as it is the first and one of the simplest attack. FGSM consists of taking a single step in the direction of the gradient of an attack objective w.r.t. the input. We also attack our classifier using two iterative versions of FGSM, *i.e.*, *Basic Iterative Method (BIM)* (Kurakin et al., 2018) and *Projected Gradient Descent (PGD)* (Madry et al., 2018). To test against a wide range of attacks, we also consider the *Carlini & Wagner's (CW)* Carlini & Wagner (2017) attack which attempts to solve the adversarial problem by regularizing the minimization of the perturbation norm by a surrogate of the misclassification constraint, and *DeepFool (DF)* Moosavi-Dezfooli et al. (2016) which is an iterative method that solves a locally linearized version of the adversarial problem. All these methods, as they rely on the gradient of a given objective w.r.t. the input, are what we call **white-box attacks**. In the event where no knowledge about the model to attack is available, **black-box attacks** have been created. Amongst them, we chose to test our detector against *Hop Skip Jump (HOP)* Chen et al. (2020), which tries to estimate the model's gradient through queries, *Square Attack (SA)* Andriushchenko et al. (2020) which is based on random searches for a perturbation, and, *Spatial Transformation Attack (STA)* Engstrom et al. (2019) which rotates and translates the original samples to fool the model.

Attack Calibration. As most of previous studies have been conducted on ResNet models (Goodfellow et al., 2014; Moosavi-Dezfooli et al., 2016; Zhang et al., 2019; Madry et al., 2018; Xu et al., 2018; Meng & Chen, 2017), to ensure attacker's success, we need to re-calibrate the maximal allowed perturbation for each attacks. We report in Fig. 2 the chosen ε for each attack which is justified by the efficiency of the attacker.



(a) FGSM (b) BIM (c) PGD$_\infty$ (d) PGD$_1$ (e) PGD$_2$ (f) Others

Figure 2: Percentage of successful attacks depending on the L_p-norm constraint, the maximal perturbation ε and the attack algorithm on ViT.

[1] https://github.com/jeonsworld/ViT-pytorch

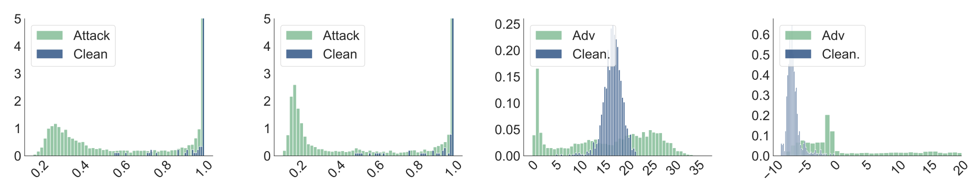
4.2 On the importance of the reference set

Setting. To understand the importance of the reference set to detect adversarial examples, we tested MSP, DOCTOR, $T_\alpha(\mathbf{u}||q_\theta(\cdot|\mathbf{x}))$ (Eq. 2) and our proposed detector REFEREE on all the previously presented attacks on all three considered datasets. In Fig. 3, we present the histograms of the scores of each detection method under the L_2-norm constraint. In Tab. 3, we present the averaged AUROC↑and FPR$\downarrow_{90\%}$on CIFAR10, CIFAR100 and Tiny ImageNet. The detailed results are presented in Tab. 7, Tab. 9, and Tab. 11.

Analysis. From Fig. 3, three different behaviors can be observed. Although DOCTOR and MSP (cf. Fig. 3b and Fig. 3a respectively) are sometimes quite effective at discarding adversarial examples, on others, it is impossible for them to distinguish between natural and adversarial samples. The detector using T_α (cf. Fig. 3c) have a different behavior. For some attacks, the scores attributed to attacked samples by each of those methods is higher than the scores of natural examples, the opposites also occurs. In other words, the method attribute sometimes over-confident and others under-confident scores to adversarial samples, it is therefore impossible to clearly distinguish between natural and attacked samples using them, as the direction of the decision is sometimes flipped. It is clear that those methods would benefit from having a reference set. REFEREE is however, able to better distinguish between natural and attacked samples as shown in Fig. 3d. All those behaviors are also clearly observable in Tab. 7, Tab. 9, and Tab. 11. From all of this, and Tab. 3, it is clear that classical OOD detection methods would benefit from having a reference set, and that REFEREE clearly outperforms them in detecting adversarial examples.

Table 3: Average AUROC and FPR for each considered softmax-based method on each considered dataset. Ours stands for REFEREE, and DOC. for DOCTOR. The best result for each attack is shown in **bold**.

		CIFAR10		CIFAR100		Tiny ImageNet	
		AUROC	FPR	AUROC	FPR	AUROC	FPR
Ours.	μ	**91.1**	**25.9**	**90.0**	**24.0**	**83.2**	**38.5**
	σ	**10.1**	31.1	**8.6**	19.1	**10.8**	**24.8**
T_α	μ	47.3	63.8	42.3	85.7	53.3	68.5
	σ	41.5	43.7	31.9	**16.3**	31.1	42.8
DOC.	μ	69.4	67.3	64.8	57.0	58.4	59.9
	σ	19.7	31.0	27.0	33.8	37.0	33.9
MSP	μ	68.6	69.5	64.6	56.4	58.8	58.8
	σ	19.0	**29.5**	16.2	33.6	37.3	34.9



(a) MSP (b) DOCTOR (c) T_α (log) (d) REFEREE (log)

Figure 3: Adversarial detection score histogram of classical OOD detection score and REFEREE.

5 Experimental results & Analysis

5.1 Global Analysis

Global Performances. We first compare the performances of REFEREE, FS and MagNet on the adversarial benchmark described in Sec. 4.1. We reported the detailed results in Tab. 6, Tab. 8, and Tab. 10, relegated in App. A, and the averaged results in Tab. 4.

Analysis. We observe substantial gains when comparing REFEREE with existing baselines such as FS or MagNet. REFEREE outperforms FS of over 15% AUROC↑on CIFAR10, CIFAR100 and Tiny. It is interesting to note that MagNet, which was originally developed for the ResNet model, does not generalize at all to ViT since it performs poorly on all datasets. The decrease in performance observed when comparing CIFAR10, CIFAR100 and Tiny shows that as the complexity of the dataset increases, the detection task becomes more difficult. Finally, from Tab. 6, Tab. 8, and Tab. 10, we can observe that the performance of REFEREE is consistently better than FS and MagNet, regardless of attacks and ε, which further validates our approach.

Table 4: Detection performance for each considered dataset. Mean (μ) and standard deviation (σ) are obtained by aggregating results by L_p-norm. AUC. stands for AUROC, and No stands for No Norm. The best result for each attack is shown in **bold**.

		REFEREE						FS						MagNet					
		CIFAR10		CIFAR100		Tiny		CIFAR10		CIFAR100		Tiny		CIFAR10		CIFAR100		Tiny	
		AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR	AUC	FPR
L_1	μ	**87.8**	**30.5**	**85.4**	**32.3**	**75.4**	**56.7**	79.5	36.0	71.2	55.5	54.2	75.1	51.3	90.1	50.1	90.2	49.4	90.9
	σ	12.3	30.7	8.6	18.0	3.0	5.2	3.3	7.9	5.1	8.0	14.0	11.0	**1.1**	**3.3**	**0.2**	**0.2**	**0.9**	1.5
L_2	μ	**89.8**	38.1	**87.0**	**29.4**	**75.5**	**55.2**	77.3	**37.2**	68.2	58.9	58.8	72.4	51.0	89.7	50.6	89.3	49.9	89.2
	σ	6.1	29.8	5.9	11.7	**1.0**	7.8	1.8	8.6	5.1	10.5	14.4	10.6	**1.2**	**2.7**	**0.7**	**2.0**	1.3	**2.6**
L_∞	μ	**93.3**	**19.6**	**93.2**	**17.5**	**91.0**	**21.1**	74.1	51.8	62.6	66.8	74.8	61.2	55.6	89.9	55.0	81.3	50.9	88.3
	σ	9.9	30.6	8.5	20.3	10.3	24.2	**4.0**	18.8	**6.8**	**11.8**	17.6	23.9	7.7	**17.4**	8.7	15.8	**2.6**	**4.5**
No.	μ	**93.3**	**6.9**	**92.5**	**20.5**	**76.9**	**51.5**	78.8	37.5	65.4	50.0	53.0	77.5	39.4	93.5	38.3	92.8	34.9	95.6
	σ	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
Avg.	μ	**91.1**	**25.9**	**89.9**	**24.0**	**83.2**	**38.5**	76.3	44.2	66.1	62.0	65.3	67.7	53.0	85.0	52.3	85.7	49.8	88.9
	σ	10.1	31.1	8.7	19.1	10.8	24.8	**4.1**	16.4	**7.0**	**11.8**	18.5	19.6	6.4	**13.5**	**7.0**	12.0	**3.7**	**4.0**

Time & Resources. To ensure the adoption of REFEREE to a real application (see **(R3)**), we investigate the resources an execution time. Tab. 5 shows a comparison of the different methods when run on NVIDIA V100 GPUs with 32Go of RAM for each considered dataset.

Analysis. REFEREE is up to two orders of magnitude faster than FS and MagNet. It should be noted that REFEREE can also be run on the CPU and takes about 0.003 sec/input.

Table 5: Execution time of each method on each dataset. Relative improvements are computed w.r.t REFEREE.

Dataset	Method	Time (min)
CIFAR 10/100	FS	51 +10200%
	MagNet	13 +2600%
	REFEREE	0.5
TINY	FS	34 +34000%
	MagNet	13 +25000%
	REFEREE	0.1

5.2 Adaptive Experiments

Adaptive Attacks. In the previous sections, we considered attacks with no knowledge about the defense. However, in the last few years, adaptive attacks (Athalye et al., 2018; Tramer et al., 2020; Carlini & Wagner, 2017), *i.e.*, attacks with full knowledge about the defense, has gain momentum. To further assess the effectiveness of our method compared to previous state-of-the-art ones, we attacked both REFEREE and FS using PGD_∞ with $\varepsilon = 0.03125$ modifying the attack objective so the attacker targets both the underlying classifier and the detection method, using an hyperparameter β to control the trade-off between the two objectives. We present the results in Fig. 4.

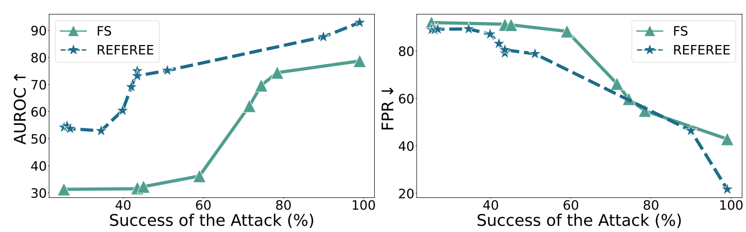


Figure 4: FS' and REFEREE's performances under adaptive attacks.

Analysis. As β increases, the effectiveness of the attack on the classifier decreases while the effectiveness on the defense increases. No matter the success of the attack on the classifier, REFEREE clearly outperforms FS, in terms of both AUROC↑and $\text{FPR}\downarrow_{90\%}$.

5.3 Ablation studies: role of α

On the importance of α. To decide whether a sample is contradictory, REFEREE relies on the Tsallis-α divergence parameterized by α. In Fig. 5, we report the performance variations when varying α. We stopped at $\alpha = 14$ due to overflow limitations. The color area of the curve correspond to the 90%-confidence region.

Analysis. Performances of REFEREE monotonically increase as α increases for both CIFAR10 and

CIFAR100. We therefore chose $\alpha = 9$. For Tiny ImageNet, we observe an optimal value for $\alpha = 3$. However, in this papers all the results are reported with $\alpha = 9$ as we wanted to provide a unified framework across datasets.

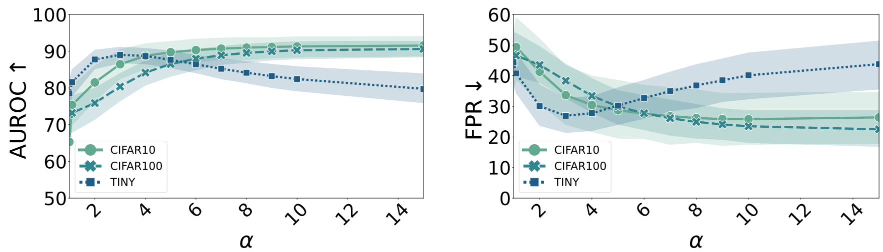


Figure 5: Study of the influence of α in `REFEREE`'s performance.

5.4 Interpreting `REFEREE`'s decisions

On the interpretability of `REFEREE`. In a practical scenario, a key ingredient to fostering adoption is the ability to monitor and verify the results of the automatic system (Montavon et al., 2018). `REFEREE` makes a step towards this ambitious objective by relying on an interpretable score. For any input sample $\mathbf{x}$, `REFEREE` computes the information projection as defined in Eq. 3. Thus one can find $\mathbf{x}^* \in \mathcal{D}_{\text{train}}$ such that:

$$\mathbf{x}^* = \arg\min_{\mathbf{x^k} \in \mathcal{D}_{\text{train}}} T_\alpha(q_\theta(\cdot|\mathbf{x^k}) \| q_\theta(\cdot|\mathbf{x}))$$

to control the decision of `REFEREE`.

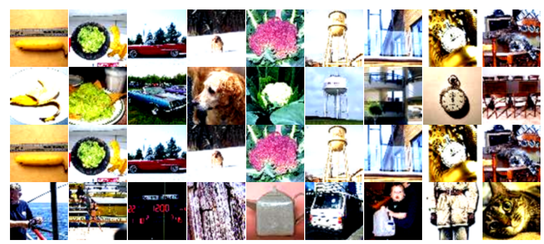


Figure 6: Example of $\mathbf{x}^*$ for different $\mathbf{x}$. First row displays clean input $\mathbf{x}$, second row its closest projection $\mathbf{x}^*$, third row displays adversarial inputs $\mathbf{x}'$, last row displays its closest projection $\mathbf{x}'^*$.

Analysis. Fig. 6 reports clean and attacked samples, along with their closest projection on the reference set. We observe that, for clean samples, the closest point in the reference set belong to the same class (row 1 and 2 of Fig. 6). However, for most of the adversarial samples, the closest reference point belongs to a different class, showing the effectiveness of the attack (row 3 and 4 of Fig. 6). Therefore, one can visually see what the prediction of the classifier is going to be, and assess its quality.

6 Conclusion

This paper revisits the problem of adversarial attack detection and approaches it under realistic constraints. The introduced detector, called `REFEREE`, is unsupervised and black-box. It is 400 times faster than previous methods (0.05s per image) and significantly outperforms existing detection methods on CIFAR10, CIFAR100 and Tiny ImageNet. We let teams with more GPUs evaluate the methods on Imagenet. The new introduced formulation opens up new avenues of research and ensures that future detectors will be ready for deployment in the real world and could benefit society.

Future Research. Our research is expected to have a positive societal impact by protecting the integrity of artificial intelligence systems, which is particularly needed in critical systems such as stock predictions (Xie et al., 2022) , autonomous cars (Morgulis et al., 2019) or healthcare systems (Newaz et al., 2020). Future work includes testing the projection of information to textual adversarial attacks where we expect to see different behaviors (Yoo et al., 2022; Le et al., 2020).

References

Ahmed Aldahdooh, Wassim Hamidouche, and Olivier Déforges. Revisiting model's uncertainty and confidences for adversarial example detection. *arXiv preprint arXiv: 2103.05354*, 2021.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European Conference on Computer Vision*, pp. 484–501. Springer, 2020.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Michéle Basseville. Divergence measures for statistical data processing—an annotated bibliography. *Signal Processing*, 93(4):621–633, 2013.

Ayanendranath Basu, Ian R Harris, Nils L Hjort, and MC Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Lev M Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR computational mathematics and mathematical physics*, 7(3), 1967.

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.

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, pp. 11192–11203, 2019.

Fabio Carrara, Rudy Becarelli, Roberto Caldelli, Fabrizio Falchi, and Giuseppe Amato. Adversarial examples detection in features distance spaces. In *Computer Vision - ECCV 2018 Workshops - Munich, Germany, Proceedings, Part II*, volume 11130, pp. 313–327. Springer, 2018.

Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)*, pp. 1277–1294. IEEE, 2020.

Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pretraining or strong data augmentations. *arXiv preprint arXiv:2106.01548*, 2021.

Herman Chernoff et al. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *The Annals of Mathematical Statistics*, 23(4), 1952.

Imre Csiszár. Information-type measures of difference of probability distributions and indirect observation. *studia scientiarum Mathematicarum Hungarica*, 2:229–318, 1967.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pp. 146–158, 1975.

Imre Csiszár. Sanov property, generalized i-projection and a conditional limit theorem. *The Annals of Probability*, pp. 768–793, 1984.

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.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness. In *International Conference on Machine Learning*, pp. 1802–1811. PMLR, 2019.

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

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

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

Federica Granese, Marco Romanelli, Daniele Gorla, Catuscia Palamidessi, and Pablo Piantanida. DOCTOR: A simple method for detecting misclassification errors. *arXiv preprint arXiv:2106.02395*, 2021.

Patrick Grother, Mei Ngan, and Kayee Hanaoka. *Face recognition vendor test (fvrt): Part 3, demographic effects*. National Institute of Standards and Technology Gaithersburg, MD, 2019.

Patrick J Grother, Patrick J Grother, and Mei Ngan. *Face recognition vendor test (frvt)*. US Department of Commerce, National Institute of Standards and Technology, 2014.

Patrick J Grother, Mei L Ngan, Kayee K Hanaoka, et al. Ongoing face recognition vendor test (frvt) part 2: Identification. 2018.

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*, pp. 770–778, 2016.

Lu He, Qianyu Zhou, Xiangtai Li, Li Niu, Guangliang Cheng, Xiao Li, Wenxuan Liu, Yunhai Tong, Lizhuang Ma, and Liqing Zhang. End-to-end video object detection with spatial-temporal transformers. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 1507–1516, 2021.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021.

Ethan Huynh. Vision transformers in 2022: An update on tiny imagenet. *arXiv preprint arXiv:2205.10660*, 2022.

Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*, 2019.

Anouar Kherchouche, Sid Ahmed Fezza, Wassim Hamidouche, and Olivier Déforges. Natural scene statistics for detecting adversarial examples in deep neural networks. In *22nd IEEE International Workshop on Multimedia Signal Processing*, pp. 1–6. IEEE, 2020.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). URL http://www.cs.toronto.edu/~kriz/cifar.html.

Solomon Kullback. *Information theory and statistics*. Courier Corporation, 1997.

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.

Thai Le, Noseong Park, and Dongwon Lee. A sweet rabbit hole by darcy: Using honeypots to detect universal trigger's adversarial attacks. *arXiv preprint arXiv:2011.10492*, 2020.

Xin Li and Fuxin Li. Adversarial examples detection in deep networks with convolutional filter statistics. In *IEEE International Conference on Computer Vision, ICCV*, pp. 5775–5783. IEEE Computer Society, 2017.

Yingzhen Li and Richard E Turner. R\'enyi divergence variational inference. *arXiv preprint arXiv:1602.02311*, 2016.

Jiajun Lu, Theerasit Issaranon, and David A. Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly. In *IEEE International Conference on Computer Vision*, pp. 446–454. IEEE Computer Society, 2017.

Shiqing Ma, Yingqi Liu, Guanhong Tao, Wen-Chuan Lee, and Xiangyu Zhang. NIC: detecting adversarial samples with neural network invariant checking. In *26th Annual Network and Distributed System Security Symposium*. The Internet Society, 2019.

Xingjun Ma, Bo Li, Yisen Wang, Sarah M. Erfani, Sudanthi N. R. Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E. Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. In *6th 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.

Dongyu Meng and Hao Chen. Magnet: A two-pronged defense against adversarial examples. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu (eds.), *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 135–147. ACM, 2017.

Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. In *5th International Conference on Learning Representations*, 2017.

Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Nir Morgulis, Alexander Kreines, Shachar Mendelowitz, and Yuval Weisglass. Fooling a real car with adversarial traffic signs. *arXiv preprint arXiv:1907.00374*, 2019.

AKM Iqtidar Newaz, Nur Imtiazul Haque, Amit Kumar Sikder, Mohammad Ashiqur Rahman, and A Selcuk Uluagac. Adversarial attacks to machine learning-based smart healthcare systems. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pp. 1–6. IEEE, 2020.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pp. 4055–4064. PMLR, 2018.

Jayaram Raghuram, Varun Chandrasekaran, Somesh Jha, and Suman Banerjee. A general framework for detecting anomalous inputs to dnn classifiers. In *International Conference on Machine Learning*, pp. 8764–8775. PMLR, 2021.

Alfréd Rényi et al. On measures of entropy and information. In *Contributions to the Theory of Statistics*, 1961.

Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

Ivan N Sanov. *On the probability of large deviations of random variables*. United States Air Force, Office of Scientific Research, 1958.

Angelo Sotgiu, Ambra Demontis, Marco Melis, Battista Biggio, Giorgio Fumera, Xiaoyi Feng, and Fabio Roli. Deep neural rejection against adversarial examples. *EURASIP J. Inf. Secur.*, 2020:5, 2020.

Andreas Steiner, Alexander Kolesnikov, , Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in Neural Information Processing Systems*, 33:1633–1645, 2020.

John W. Tukey. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians*, volume 2, pp. 523–531, 1975.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

Thomas Villmann and Sven Haase. Mathematical aspects of divergence based vector quantization using fréchet-derivatives. *University of Applied Sciences Mittweida*, 2010.

Yulin Wang, Rui Huang, Shiji Song, Zeyi Huang, and Gao Huang. Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition. *Advances in Neural Information Processing Systems*, 34, 2021.

Yong Xie, Dakuo Wang, Pin-Yu Chen, Jinjun Xiong, Sijia Liu, and Oluwasanmi Koyejo. A word is worth a thousand dollars: Adversarial attack on tweets fools stock prediction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 587–599, Seattle, United States, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.naacl-main.43.

Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *25th Annual Network and Distributed System Security Symposium*. The Internet Society, 2018.

KiYoon Yoo, Jangho Kim, Jiho Jang, and Nojun Kwak. Detection of word adversarial examples in text classification: Benchmark and baseline via robust density estimation. *arXiv preprint arXiv:2203.01677*, 2022.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. *CVPR*, 2022.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pp. 1–11, 2019.

Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6881–6890, 2021.

Zhihao Zheng and Pengyu Hong. Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. In *Advances in Neural Information Processing Systems 31*, pp. 7924–7933, 2018.

A DETAILED RESULTS

Table 6: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on CIFAR10 for REFEREE, FS and MagNet on ViT. The best result for each attack is shown in **bold**.

CIFAR10						
Norm L1	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]						
ε = 50	**95.4**	**4.8**	77.6	37.5	53.3	90.1
ε = 60	**95.2**	**5.4**	77.4	37.5	51.6	92.1
ε = 70	**93.6**	**6.5**	78.0	31.2	51.9	92.0
ε = 80	**92.1**	**13.1**	78.1	31.2	51.3	91.9
ε = 90	**90.1**	36.1	78.7	**31.2**	52.0	91.6
ε = 100	**88.4**	47.9	79.0	**37.5**	51.6	91.6
ε = 500	55.4	88.7	**86.8**	**25.0**	49.6	90.5
ε = 1000	81.8	71.5	**83.7**	**37.5**	49.9	90.0
ε = 5000	**97.8**	**0.6**	76.0	55.2	50.1	89.9
Norm L2	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]						
ε = 0.125	**95.9**	**3.9**	75.5	37.5	50.6	92.1
ε = 0.25	**95.1**	**5.3**	77.2	37.5	52.2	91.7
ε = 0.5	**85.7**	59.6	79.8	**31.2**	50.6	91.6
ε = 5	**85.2**	65.0	77.0	**45.9**	50.0	89.8
ε = 10	**87.5**	58.5	76.8	**52.1**	50.1	89.8
HOP						
ε = 0.1	**98.5**	**2.7**	74.5	25.0	53.4	83.6
DeepFool						
No ε	**80.9**	71.7	79.7	**31.2**	50.3	89.7
Norm L_∞	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD^∞						
ε = 0.03125	**92.9**	**21.7**	78.7	42.9	50.3	89.6
ε = 0.0625	**99.9**	**0.0**	73.4	64.7	51.0	88.4
ε = 0.125	**100**	**0.0**	71.8	68.6	52.9	85.5
ε = 0.25	**100**	**0.0**	70.9	70.0	54.3	83.4
ε = 0.5	**100**	**0.0**	70.8	70.1	54.4	83.3
BIM						
ε = 0.03125	67.6	84.0	**74.0**	**64.5**	50.3	89.6
ε = 0.0625	**95.6**	**4.1**	70.2	72.3	50.7	88.9
ε = 0.125	**99.9**	**0.0**	70.0	72.2	51.8	87.2
ε = 0.25	**100**	**0.0**	70.7	70.5	53.6	84.4
ε = 0.5	**100**	**0.0**	71.2	68.4	56.4	80.1
FGSM						
ε = 0.03125	73.5	80.2	**75.2**	**38.8**	51.9	88.1
ε = 0.0625	**80.4**	72.3	77.2	**37.5**	53.0	86.1
ε = 0.125	**92.9**	**10.2**	78.9	31.2	57.3	79.2
ε = 0.25	**99.5**	**0.9**	69.6	25.0	70.6	54.8
ε = 0.5	**99.7**	**0.5**	67.7	31.2	80.4	18.0
SA						
ε = 0.125	**98.0**	**2.9**	72.0	25.0	55.1	82.4
CW^∞						
ε = 0.3125	**87.0**	56.0	78.8	**37.5**	50.6	89.3
No Norm	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA						
No ε	**93.3**	**6.9**	78.8	37.5	39.4	93.5

Table 7: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on CIFAR10 for REFEREE, T_α, DOCTOR, and MSP. The best result for each attack is shown in **bold**.

CIFAR10								
Norm L1	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]								
ε = 50	**95.4**	**4.8**	12.3	99.4	89.7	43.6	88.8	49.3
ε = 60	**95.2**	**5.4**	12.8	99.4	89.0	47.2	87.1	56.7
ε = 70	**93.6**	**6.5**	14.2	99.4	87.0	56.2	84.7	63.9
ε = 80	**92.1**	**13.1**	15.6	99.2	83.8	65.8	81.0	71.7
ε = 90	**90.1**	**36.1**	17.2	99.2	80.6	72.1	78.4	75.3
ε = 100	**88.4**	**47.9**	18.5	99.1	78.8	74.7	77.0	77.0
ε = 500	55.4	88.7	**67.0**	**82.1**	50.8	90.1	50.7	90.1
ε = 1000	81.8	71.5	**89.5**	**30.0**	48.1	91.2	48.3	91.2
ε = 5000	97.8	**0.6**	**98.3**	2.4	47.8	91.1	48.1	91.1
Norm L2	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]								
ε = 0.125	**95.9**	**3.9**	10.9	99.7	93.3	4.7	92.6	17.3
ε = 0.25	**95.1**	**5.3**	12.4	99.4	89.4	44.7	86.1	60.3
ε = 0.5	**85.7**	**59.6**	20.4	98.9	76.4	77.5	74.1	79.7
ε = 5	85.2	65.0	**87.5**	**40.5**	49.0	90.7	49.2	90.7
ε = 10	87.5	58.5	**88.6**	**36.6**	48.9	90.8	49.1	90.8
HOP								
ε = 0.1	**98.5**	2.7	4.9	99.7	96.8	2.7	95.8	**2.6**
DeepFool								
No ε	**80.9**	**71.7**	17.7	100	73.9	79.9	72.4	81.1
Norm L_∞	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD$^\infty$								
ε = 0.03125	92.9	21.7	**94.6**	**13.8**	48.7	91.1	48.8	91.1
ε = 0.0625	**99.9**	**0.0**	99.8	0.1	48.2	91.0	48.3	91.0
ε = 0.125	**100**	**0.0**	**100**	**0.0**	48.1	90.7	48.2	90.6
ε = 0.25	**100**	**0.0**	**100**	**0.0**	48.1	90.7	48.2	90.6
ε = 0.5	**100**	**0.0**	**100**	**0.0**	48.1	90.6	48.3	90.6
BIM								
ε = 0.03125	67.6	84.0	**76.7**	**61.1**	49.3	90.5	49.4	90.5
ε = 0.0625	**95.6**	**4.1**	**95.6**	10.2	49.0	90.8	49.0	90.8
ε = 0.125	**99.9**	**0.0**	99.8	0.1	48.4	90.8	48.5	90.8
ε = 0.25	**100**	**0.0**	**100**	**0.0**	48.2	90.6	48.3	90.6
ε = 0.5	**100**	**0.0**	**100**	**0.0**	48.1	90.5	48.3	90.5
FGSM								
ε = 0.03125	**73.5**	**80.2**	21.4	99.9	67.3	84.1	66.1	84.7
ε = 0.0625	**80.4**	**72.3**	10.4	100	73.7	80.1	72.1	81.3
ε = 0.125	**92.9**	**10.2**	1.8	100	87.6	55.6	86.0	61.5
ε = 0.25	**99.5**	0.9	0.0	100	99.1	**0.7**	98.8	**0.7**
ε = 0.5	99.7	0.5	0.0	100	99.7	**0.0**	**99.8**	**0.0**
SA								
ε = 0.125	**98.0**	2.9	4.5	100	96.1	**2.5**	95.4	**2.5**
CW$^\infty$								
ε = 0.3125	**87.0**	**56.0**	15.4	100	80.8	72.1	78.8	75.1
No Norm	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA								
No ε	**93.3**	**6.9**	5.4	100	88.3	53.6	86.2	61.6

Table 8: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on CIFAR100 for REFEREE, FS and MagNet. The best result for each attack is shown in **bold**.

CIFAR100						
Norm L1	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]						
ε = 50	**89.4**	**22.7**	65.5	56.2	50.5	90.5
ε = 60	**87.8**	**26.0**	66.6	56.2	50.5	90.3
ε = 70	**86.1**	**29.8**	67.4	50.0	50.0	90.4
ε = 80	**84.7**	**32.4**	68.3	50.0	50.0	90.4
ε = 90	**83.0**	**35.7**	69.2	50.0	50.2	90.3
ε = 100	**81.3**	**39.6**	70.1	50.0	50.1	90.4
ε = 500	65.3	74.5	**79.3**	**50.0**	50.0	90.0
ε = 1000	**92.3**	**28.2**	80.0	62.5	50.0	89.9
ε = 5000	**98.5**	**1.9**	74.0	75.0	50.0	89.8
Norm L2	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]						
ε = 0.125	**90.7**	**21.6**	64.6	56.2	50.8	90.8
ε = 0.25	**88.2**	**25.3**	66.2	56.2	50.8	90.1
ε = 0.5	**78.3**	**45.0**	72.0	50.0	50.3	90.0
ε = 5	**92.4**	**27.4**	75.1	75.0	50.0	89.9
ε = 10	**93.3**	**22.2**	74.4	75.0	50.0	89.9
HOP						
ε = 0.1	**93.0**	**15.2**	62.7	50.0	52.1	84.5
DeepFool						
No ε	**79.5**	**49.1**	62.2	50.0	50.0	89.9
Norm L_∞	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD^∞						
ε = 0.03125	**88.1**	**36.2**	76.0	74.8	50.2	89.7
ε = 0.0625	**99.1**	**1.5**	68.9	75.0	50.6	88.9
ε = 0.125	**99.9**	**0.0**	65.5	75.0	52.1	86.5
ε = 0.25	**100**	**0.0**	64.3	75.0	53.0	84.9
ε = 0.5	**100**	**0.0**	64.2	75.0	53.1	84.8
BIM						
ε = 0.03125	**68.0**	**71.2**	67.6	75.0	50.2	89.7
ε = 0.0625	**89.2**	**34.5**	63.0	81.1	50.5	89.2
ε = 0.125	**98.8**	**2.3**	62.1	82.7	51.3	87.8
ε = 0.25	**99.9**	**0.0**	63.7	75.4	52.5	85.7
ε = 0.5	**100**	**0.0**	65.3	75.0	54.6	82.2
FGSM						
ε = 0.03125	**82.6**	**43.8**	61.9	62.5	51.0	88.8
ε = 0.0625	**88.0**	**31.8**	61.3	61.4	52.1	86.8
ε = 0.125	**93.3**	**20.3**	54.8	50.0	55.8	80.2
ε = 0.25	**97.8**	**6.8**	49.6	50.0	66.4	60.4
ε = 0.5	**99.4**	**1.4**	46.2	56.2	86.6	24.2
SA						
ε = 0.125	**93.4**	**16.9**	63.3	50.0	54.9	82.6
CW^∞						
ε = 0.3125	**86.6**	**31.5**	67.0	50.0	50.0	89.8
No Norm	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA						
No ε	**92.5**	**20.5**	65.4	50.0	38.3	92.8

Table 9: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on CIFAR100 for REFEREE, T_α, DOCTOR, and MSP. The best result for each attack is shown in **bold**.

CIFAR100								
Norm L1	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]								
ε = 50	**98.4**	22.7	19.1	98.3	89.7	**22.0**	89.7	22.1
ε = 60	**87.8**	26.0	20.8	97.8	87.7	25.9	87.6	**25.8**
ε = 70	**86.1**	29.8	22.3	97.4	85.8	**29.3**	85.6	29.4
ε = 80	**84.7**	**32.4**	24.0	96.9	83.8	33.6	83.6	33.4
ε = 90	**83.0**	**35.7**	25.9	96.5	81.8	37.9	81.6	38.0
ε = 100	**81.3**	**39.6**	27.4	96.1	79.9	42.2	79.5	42.8
ε = 500	65.3	74.5	**70.1**	72.9	33.2	93.9	34.1	93.5
ε = 1000	**92.3**	**28.2**	73.0	74.4	30.9	94.3	32.2	93.8
ε = 5000	**98.5**	**1.9**	71.6	78.3	32.7	93.9	33.5	93.5
Norm L2	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]								
ε = 0.125	90.7	21.6	17.6	98.4	**92.0**	**16.5**	**92.1**	**16.4**
ε = 0.25	**88.2**	25.3	20.3	97.8	**88.2**	**24.4**	**88.2**	24.5
ε = 0.5	**78.3**	**45.0**	30.8	95.2	76.5	48.7	75.8	49.2
ε = 5	**92.4**	**27.4**	71.3	76.0	31.0	94.4	32.5	93.8
ε = 10	**93.3**	**22.2**	70.8	76.7	31.2	94.3	32.6	93.7
HOP								
ε = 0.1	**93.0**	**15.2**	8.1	99.5	91.8	17.0	91.6	17.1
DeepFool								
No ε	**79.5**	**49.1**	22.9	98.9	78.8	51.6	78.1	52.6
Norm L_∞	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD^∞								
ε = 0.03125	**88.1**	**36.2**	74.3	76.6	33.3	94.2	34.9	93.4
ε = 0.0625	**99.1**	**1.5**	78.8	74.9	35.1	93.6	36.0	93.1
ε = 0.125	**99.9**	**0.0**	85.3	63.9	36.9	92.8	37.1	92.7
ε = 0.25	**100**	**0.0**	87.4	58.8	37.9	92.5	38.1	82.4
ε = 0.5	**100**	**0.0**	87.5	58.5	38.3	92.3	38.3	92.4
BIM								
ε = 0.03125	**68.0**	**71.2**	62.4	85.2	36.0	93.7	37.2	93.0
ε = 0.0625	**89.2**	**34.5**	69.4	82.5	35.1	93.8	36.5	93.1
ε = 0.125	**98.8**	**2.3**	77.1	76.5	34.7	93.7	35.8	93.1
ε = 0.25	**99.9**	**0.0**	85.1	64.1	36.6	93.0	37.0	92.8
ε = 0.5	**100**	**0.0**	92.2	36.7	41.2	91.0	40.4	91.7
FGSM								
ε = 0.03125	**82.6**	**43.8**	8.7	99.9	81.6	45.6	80.6	46.0
ε = 0.0625	**88.0**	**31.8**	4.0	100	86.9	34.2	86.2	34.5
ε = 0.125	**93.3**	**20.3**	1.6	100	92.6	22.7	92.1	22.9
ε = 0.25	**97.8**	**6.8**	0.3	100	97.6	9.8	97.2	10.4
ε = 0.5	**99.4**	1.4	0.0	100	99.3	**1.1**	99.0	2.5
SA								
ε = 0.125	93.4	16.9	10.6	99.6	94.2	**13.3**	**94.3**	13.4
CW^∞								
ε = 0.3125	86.6	31.5	20.0	98.8	**86.8**	30.5	86.7	**30.4**
No Norm	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA								
No ε	92.5	20.5	4.6	99.9	**92.9**	**19.4**	92.5	19.5

Table 10: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on Tiny ImageNet for REFEREE, FS and MagNet. The best result for each attack is shown in **bold**.

Tiny						
Norm L1	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]						
ε = 50	**74.6**	**56.3**	44.8	81.6	50.4	88.9
ε = 60	**74.9**	**56.6**	45.0	81.8	50.3	88.9
ε = 70	**75.4**	**56.6**	45.1	82.0	50.0	89.0
ε = 80	**75.6**	**54.6**	45.1	82.3	49.6	88.9
ε = 90	**76.0**	**51.8**	45.0	82.2	49.7	89.3
ε = 100	**76.2**	**50.9**	44.9	82.0	49.6	89.0
ε = 500	**73.0**	**61.8**	60.7	71.7	48.0	93.1
ε = 1000	70.2	68.8	**73.7**	**62.4**	47.6	92.0
ε = 5000	82.4	53.1	**83.2**	**50.0**	49.1	90.3
Norm L2	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]						
ε = 0.125	**74.7**	**57.0**	45.2	81.4	50.2	88.7
ε = 0.25	**76.2**	**50.9**	45.2	81.8	49.3	89.7
ε = 0.5	**77.0**	**49.2**	47.1	79.5	49.6	91.0
ε = 5	74.1	65.4	**77.9**	**57.5**	48.7	91.0
ε = 10	74.9	64.4	**78.1**	**57.7**	48.8	90.9
HOP						
ε = 0.1	**75.9**	**44.5**	59.1	76.3	52.7	83.8
Norm L_∞	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD^∞						
ε = 0.03125	**97.9**	**2.0**	96.0	8.2	49.7	90.0
ε = 0.0625	**99.9**	**0.0**	93.8	11.9	49.8	89.9
ε = 0.125	**99.9**	**0.0**	89.2	47.1	49.9	89.6
ε = 0.25	**100**	**0.0**	85.5	73.6	50.0	89.5
ε = 0.5	**100**	**0.0**	83.6	82.2	50.1	89.4
BIM						
ε = 0.03125	**86.8**	**42.6**	86.0	44.8	49.5	90.1
ε = 0.0625	**99.4**	**0.1**	90.3	33.4	49.9	89.9
ε = 0.125	**99.9**	**0.0**	87.4	61.4	49.9	89.8
ε = 0.25	**100**	**0.0**	84.9	79.9	50.0	89.5
ε = 0.5	**100**	**0.0**	83.9	82.5	50.2	89.1
FGSM						
ε = 0.03125	**74.3**	**60.1**	56.3	75.5	49.7	90.2
ε = 0.0625	**76.8**	**55.6**	58.0	71.8	50.4	89.6
ε = 0.125	**79.0**	**51.0**	53.6	75.1	50.9	88.7
ε = 0.25	**82.1**	**43.5**	48.1	78.8	52.6	86.2
ε = 0.5	**84.9**	**37.0**	50.9	74.2	60.7	72.1
SA						
ε = 0.125	**74.4**	**46.2**	48.7	78.5	50.6	89.4
No Norm	REFEREE		FS		MagNet	
	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA						
No ε	**76.9**	**51.5**	53.0	77.5	34.9	95.6

Table 11: AUROC and FPR for each considered attack mechanisms, L_p-norm constraint and ε on Tiny ImageNet for REFEREE, T_α, DOCTOR, and MSP. The best result for each attack is shown in **bold**.

Tiny ImageNet								
Norm L1	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[1]								
ε = 50	74.6	56.3	27.8	97.2	86.7	30.9	**87.8**	**28.6**
ε = 60	74.9	56.6	27.6	97.3	86.9	29.6	**88.0**	**27.3**
ε = 70	75.4	56.6	27.7	97.2	87.2	29.2	**88.3**	**26.2**
ε = 80	75.6	54.6	27.7	97.3	87.3	28.9	**88.5**	**25.6**
ε = 90	76.0	51.8	28.0	97.3	87.4	28.6	**88.7**	**25.3**
ε = 100	76.2	50.9	27.5	97.4	87.6	28.1	**88.8**	**25.0**
ε = 500	73.0	61.8	21.4	96.9	82.5	**39.1**	**82.8**	39.3
ε = 1000	70.2	**68.8**	40.5	96.4	**71.6**	79.7	**71.6**	80.0
ε = 5000	**82.4**	**53.1**	64.3	94.3	49.0	99.6	48.9	99.6
Norm L2	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[2]								
ε = 0.125	74.7	57.0	27.5	97.3	86.8	30.8	**87.8**	**27.9**
ε = 0.25	76.2	50.9	27.6	97.2	87.6	28.1	**88.8**	**24.8**
ε = 0.5	77.0	49.2	26.8	97.6	88.0	28.2	**89.1**	**25.0**
ε = 5	**74.1**	**65.4**	48.8	96.0	62.6	95.6	62.5	95.7
ε = 10	**74.9**	**64.4**	50.0	96.1	61.6	96.6	61.4	96.7
HOP								
ε = 0.1	75.9	44.5	39.7	91.4	86.1	28.3	**87.3**	**25.1**
Norm L_∞	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
PGD[∞]								
ε = 0.03125	**97.9**	**2.0**	97.2	2.2	7.0	100	7.0	100
ε = 0.0625	**99.9**	**0.0**	**99.9**	**0.0**	0.5	100	0.6	100
ε = 0.125	99.9	**0.0**	**100**	**0.0**	0.0	100	0.1	100
ε = 0.25	**100**	**0.0**	**100**	**0.0**	0.0	100	0.0	100
ε = 0.5	**100**	**0.0**	**100**	**0.0**	0.0	100	0.0	100
BIM								
ε = 0.03125	**86.8**	**42.6**	78.5	81.0	40.0	99.8	39.9	99.8
ε = 0.0625	**99.4**	0.1	98.7	**0.0**	10.9	100	10.9	100
ε = 0.125	**99.9**	**0.0**	**99.9**	**0.0**	1.2	100	1.3	100
ε = 0.25	**100**	**0.0**	**100**	**0.0**	0.0	100	0.1	100
ε = 0.5	**100**	**0.0**	**100**	**0.0**	0.0	100	0.0	100
FGSM								
ε = 0.03125	74.3	60.1	24.0	98.0	**85.5**	36.0	85.2	**35.8**
ε = 0.0625	76.8	55.6	28.1	96.9	**85.7**	**36.1**	85.4	36.9
ε = 0.125	79.0	51.0	30.7	95.2	**87.0**	**32.7**	86.7	**32.7**
ε = 0.25	82.1	43.5	32.6	91.9	**89.2**	**26.1**	88.8	27.1
ε = 0.5	84.9	37.0	36.3	89.5	**91.1**	**21.3**	90.7	22.7
SA								
ε = 0.125	74.4	46.2	33.1	94.9	85.1	29.3	**87.8**	**22.2**
No Norm	REFEREE		T_α		DOCTOR		MSP	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
STA								
No ε	76.9	51.5	32.6	94.3	86.5	33.7	**87.0**	**32.2**