Your Out-of-Distribution Detection Method is Not Robust!

Mohammad Azizmalayeri, Arshia Soltani Moakhar, Arman Zarei, Reihaneh Zohrabi, Mohammad Taghi Manzuri, Mohammad Hossein Rohban

Department of Computer Engineering
Sharif University of Technology

{m.azizmalayeri, arshia.soltani, arman.zarei, zohrabi, manzuri, rohban}@sharif.edu

Abstract

Out-of-distribution (OOD) detection has recently gained substantial attention due to the importance of identifying out-of-domain samples in reliability and safety. Although OOD detection methods have advanced by a great deal, they are still susceptible to adversarial examples, which is a violation of their purpose. To mitigate this issue, several defenses have recently been proposed. Nevertheless, these efforts remained ineffective, as their evaluations are based on either small perturbation sizes, or weak attacks. In this work, we re-examine these defenses against an end-to-end PGD attack on in/out data with larger perturbation sizes, e.g. up to commonly used $\epsilon = 8/255$ for the CIFAR-10 dataset. Surprisingly, almost all of these defenses perform worse than a random detection under the adversarial setting. Next, we aim to provide a robust OOD detection method. In an ideal defense, the training should expose the model to almost *all* possible adversarial perturbations, which can be achieved through adversarial training. That is, such training perturbations should based on both in- and out-of-distribution samples. Therefore, unlike OOD detection in the standard setting, access to OOD, as well as in-distribution, samples sounds necessary in the adversarial training setup. These tips lead us to adopt generative OOD detection methods, such as OpenGAN, as a baseline. We subsequently propose the Adversarially Trained Discriminator (ATD), which utilizes a pre-trained robust model to extract robust features, and a generator model to create OOD samples. We noted that, for the sake of training stability, in the adversarial training of the discriminator, one should attack real in-distribution as well as real outliers, but not generated outliers. Using ATD with CIFAR-10 and CIFAR-100 as the in-distribution data, we could significantly outperform all previous methods in the robust AUROC while maintaining high standard AUROC and classification accuracy. The code repository is available at `https://github.com/rohban-lab/ATD`.

1 Introduction

Advances in the deep neural networks have led to their widespread use in real-world applications, such as object detection and image classification [1, 2]. These models have a high degree of generalizability to the extent that they assign an arbitrarily high probability even to the samples not belonging to the training set [3]. This phenomenon causes problems in safety-critical applications like medical diagnosis or autonomous driving that should treat anomalous data differently.

The problem of identifying out-of-distribution data has been widely explored in different categories such as Novelty Detection, Open-Set Recognition and Out-of-Distribution (OOD) detection [4, 5, 6].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

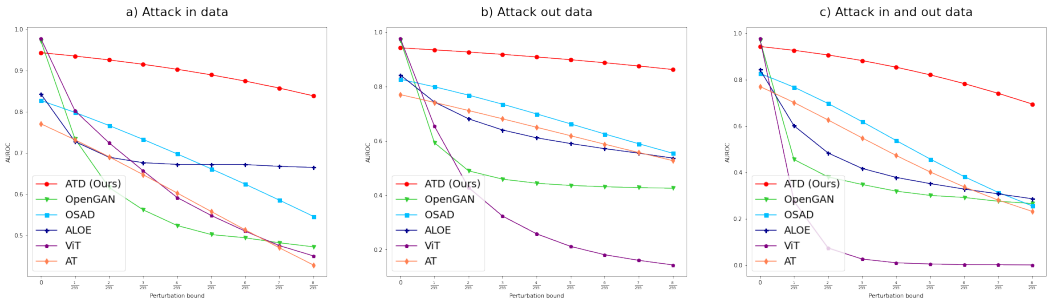

Figure 1: OOD detection scores for several models against the perturbation bound. The perturbations are designed using an end-to-end ℓ_∞ PGD attack. CIFAR-10 is used as the in-distribution dataset. a) Only the in-distribution dataset is attacked in the evaluation. b) Only out-distribution datasets (e.g., MNIST, Places, etc.) are attacked in the evaluation. c) Both in- and out-distribution datasets are attacked. ATD (Our method) outperforms others by a significant margin.

These categories are almost similar since they all consider two disjoint sets called the closed (or normal) and open (or anomalous) sets, and the model should discriminate them [7]. So far, outstanding models have been proposed in this research field [8, 9, 10, 11]. Still, these models need to be examined from other aspects, specifically robustness against adversarial perturbations added to the input data.

Deep networks are vulnerable against adversarial examples. This phenomenon was first noticed in the image classification, but it also extends to other domains [12, 13, 14]. Adversarial examples are the inputs that are slightly perturbed with a bounded perturbation (e.g. through bounding the ℓ_p-norm) to cause a high prediction error in the model. Several defenses have been proposed to make the image classification robust, none of which is as effective as adversarial training and its variants [15, 16, 17, 18]. Similar to other learning domains, OOD detection methods also suffer from the existence of adversarial examples. A small perturbation can cause a sample from the closed set to be classified as anomaly and vice-versa [19, 20].

Robust OOD detection is the connection between these two safety-critical issues. Since a sample does not change semantically under appropriately bounded perturbation in the adversarial attacks, the OOD detector model is expected to be able to label them as open/closed correctly. As a first thought, one might envision solving the issue by using the image classification defenses. Despite advances in these defenses, they have no intuition of the OOD data and not seen them in the training, making them less efficient against end-to-end attacks on the OOD detection method.

Two main approaches are introduced in previous works for the robust OOD detection. The first one uses some OOD data, uniformly sampled from the open set classes and disjoint from the closed set, and employs adversarial training to encourage the model to assign a uniform label, as opposed to one-hot labels, to anomalous samples even under adversarial perturbations [21, 22, 23]. The second approach conducts adversarial training only on the closed set, but encourages the model to learn more semantic features using representation learning techniques such as auto-encoders, self-supervised learning, and denoising methods [24, 25]. Despite the benefits of these methods, they still have fundamental issues such as:

- They did not evaluate their method with an end-to-end attack on the detection method, e.g. they attack the *classification model*, and not the detector, to evaluate the detection method [24, 25]. We show that this is not an effective and strong attack on the detector.
- They consider small perturbation sizes (e.g. $\epsilon = \frac{1}{255}$ in the CIFAR-10 dataset [23], or not attacking the in-distribution [19, 22]) to protect the standard classification accuracy, while we show that images do not change perceptually with larger perturbation sizes (e.g. $\epsilon = \frac{8}{255}$ in the CIFAR-10 dataset). Thus, these methods should be trained and evaluated against larger perturbations.
- The second approach is still unaware of OOD samples during training. Furthermore, the first approach also cannot cover various aspects of OOD data since they can only use a small number of OOD training samples.

In this work, we first evaluate previous defenses as well as standard OOD detection methods against an end-to-end PGD attack [15]. As shown in Fig. 1, almost all of the earlier methods perform worse than random in the binary OOD detection with $\epsilon = \frac{8}{255}$ in CIFAR-10. As a result, the robust

OOD detection method is still far from being solved at present. In spite of this, it is possible to design a robust model by addressing the drawbacks of previous methods. To this end, we propose Adversarialy Trained Discriminator (ATD) that uses a GAN-based method to generate OOD samples and discriminate them from the closed set. An ideal generator naturally deceives the discriminator, and therefore produces adversarial outputs from the open set [26, 27, 28]. This leads to the implicit robustness of the discriminator against adversarial attacks on the open set. In addition, we also conduct adversarial training on the closed set and the tiny real open set (known as outlier exposure) to achieve robustness on them as well. Our results (e.g., Fig. 1) show that ATD outperforms previous methods by a large margin. Therefore, this article has taken a significant step in this area by revealing the vulnerability of all previous methods and providing a solution.

2 Background

2.1 Out-of-Distribution Detection

Probabilities: A simple but effective method for anomaly detection is the Maximum Softmax Probability (MSP) [29, 30]. This method is applied on a K-class classifier and returns $\max_{c\in\{1,2,...,k\}} f_c(x)$ as the closed-set membership score of the sample x. Currently, it is shown that using the MSP with Vision Transformers (ViT) [31] leads to SOTA results in cross-dataset open set recognition [11, 32]. Also, OpenMax [33] replaces the softmax layer with a layer that calibrates the logits by fitting a class-wise probability model such as the Weibull distribution [34].

Distances: The anomaly sample can be detected by its distance to the class conditional distributions. Mahalanobis distance (MD) [35] and Relative MD (RMD) [36] are the main methods in this regard. For an in-distribution with K classes, these methods fit a class conditional Gaussian distribution $\mathcal{N}(\mu_k,\, \Sigma)$ to the pre-logit features z. The mean vector and covariance matrix are calculated as:

$$\mu_k = \frac{1}{N}\sum_{i:y_i=k} z_i, \quad \Sigma = \frac{1}{N}\sum_{k=1}^{K}\sum_{i:y_i=k}(z_i-\mu_k)(z_i-\mu_k)^T, \quad k=1,2,...,K. \tag{1}$$

In addition, to use RMD, one has to fit a $\mathcal{N}(\mu_0,\, \Sigma_0)$ to the whole in-distribution. Next, the distances and anomaly score for the input x' with pre-logits z' are computed as:

$$MD_k(z') = (z'-\mu_k)^T\Sigma^{-1}(z'-\mu_k), \quad RMD_k(z') = MD_k(z') - MD_0(z'), \tag{2}$$

$$score_{MD}(x') = -\min_k\{MD_k(z')\}, \quad score_{RMD}(x') = -\min_k\{RMD_k(z')\}. \tag{3}$$

Discriminators: Previous categories define the score function based on a K-way classifier, but one can directly train a binary discriminator for this purpose. Outlier Exposure (OE) [37] exploits some outlier data to learn a binary discriminator for the open set discrimination. GOpenMax [38], OSRCI [39], and Confident Classifier [40] augment the closed set with fake images through a GAN-based generator. Also, OpenGAN [28] selects the best discriminator model with an open validation set during training due to unstable training of GANs.

2.2 Adversarial Attacks

For an input x with the ground-truth label y, an adversarial example x^* is crafted by adding small noise to x such that the predictor model loss $J(x^*, y)$ is maximized. The ℓ_p norm of the adversarial noise should be less than a specified value ϵ, i.e. $\|x - x^*\|_p \leq \epsilon$, to ensure that the image does not change semantically. Fast Gradient Sign Method (FGSM) [13] maximizes the loss function with a single step toward the sign of the gradient of $J(x, y)$ with respect to the x:

$$x^* = x + \epsilon \,.\, sign(\nabla_x J(x, y)), \tag{4}$$

where the noise meets the ℓ_∞ norm bound ϵ. Moreover, this method can be conducted iteratively [41] with a smaller step size α:

$$x_0^* = x, \quad x_{t+1}^* = x_t^* + \alpha \,.\, sign(\nabla_x J(x_t^*, y)), \tag{5}$$

where the noise should be projected to the ℓ_∞-ball with ϵ radius at each step, which is called Projected Gradient Descent (PGD) attack [15]. There are a number of other attacks for evaluating robustness of the models [42, 43, 44], but PGD is regarded as a standard and powerful attack.

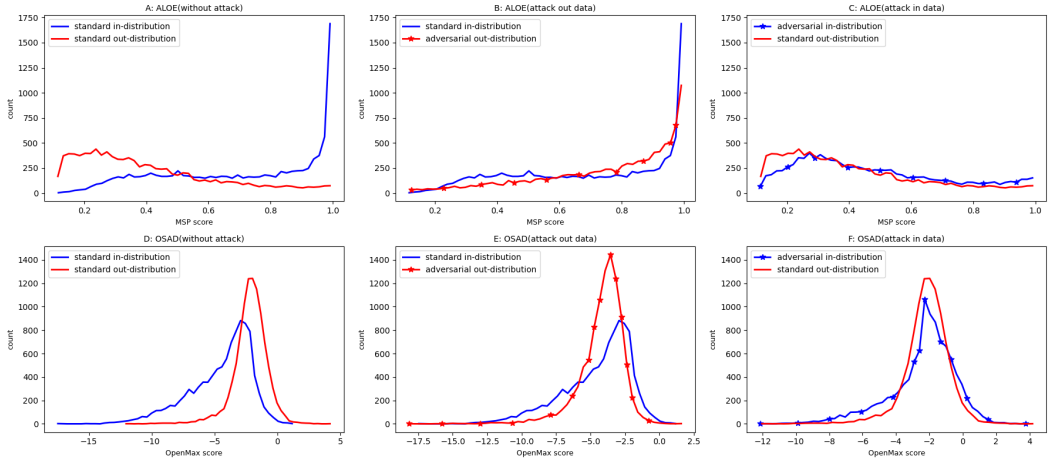

Figure 2: OOD score distribution shift after an end-to-end PGD attack with $\epsilon = \frac{8}{255}$ using CIFAR-10 as the in-distribution set. The first and second row correspond to the ALOE and OSAD methods, respectively. In each row, the left column represents the standard score distributions of in- and out-distributions. In each of the next columns, one of the in or out sets is attacked. The plots show that the standard distribution scores have drastically changed after the attack.

2.3 Adversarial Defenses

The most effective defenses for image classification and OOD detection are variants of adversarial training, which are described in the following.

Adversarial Training (AT) [15]: Through the use of adversarial examples during training, the model can learn robust features to withstand adversarial perturbations. This method is called AT and can be used to optimize model f_θ as:

$$\arg\min_{\theta} \; \mathbb{E}_{(x,y)\in D_{in}} \max_{\|x-x^*\|_\infty \leq \epsilon} J(x^*, y; f_\theta), \tag{6}$$

where the inner maximization can be approximated with PGD and the outer minimization with SGD. AT naturally reduces the standard accuracy, i.e. the accuracy on unperturbed samples [45], which can harm the anomaly detection score [11]. Therefore, we also propose Helper-Based Adversarial Training (HAT) [46] as a baseline that achieves a better trade-off between accuracy and robustness.

Adversarial Learning with inliner and Outlier Exposure (ALOE) [23]: AT lacks information about the outlier data. Therefore, it is only robust on the closed set and not on outliers. To mitigate this issue, ALOE includes some outliers in the AT similar to OE [37]. Outliers are used with uniform label $\mathcal{U}_K$ and attacked during training to obtain a model that is robust on both in- and out-distribution. The objective function of ALOE is summarized as:

$$\arg\min_{\theta} \; \mathbb{E}_{(x,y)\in D_{in}} \max_{\|x-x^*\|_\infty \leq \epsilon} J(x^*, y; f_\theta) \; + \; \lambda \, . \, \mathbb{E}_{x\in D_{out}} \max_{\|x-x^*\|_\infty \leq \epsilon} J(x^*, \, \mathcal{U}_K; f_\theta). \tag{7}$$

An identical objective function is also used in RATIO [21] with a ℓ_2 perturbation bound. Alternatively, one could consider using clean outliers samples in the training. In addition to ALOE, we evaluate this alternative method as a baseline, and refer to it as Adversarial OE (AOE).

Open-Set Adversarial Defense (OSAD) [24, 25]: Instead of including outlier data in the AT, OSAD tries to learn more semantic features in the training. To this end, OSAD adds dual-attentive denoising layers to the model architecture, and combines the AT loss with an auto-encoder loss, whose aim is to reconstruct the clean image from its adversarial version. A self-supervision loss is also added by applying transformations on the input image. They claim that this combination improves OSR robustness, but they did not evaluate the model against an end-to-end attack.

3 Method

An OOD detection method should be robust against adversarial perturbations that are added to the closed or open datasets. Despite previous efforts to provide robust OOD detection methods, the

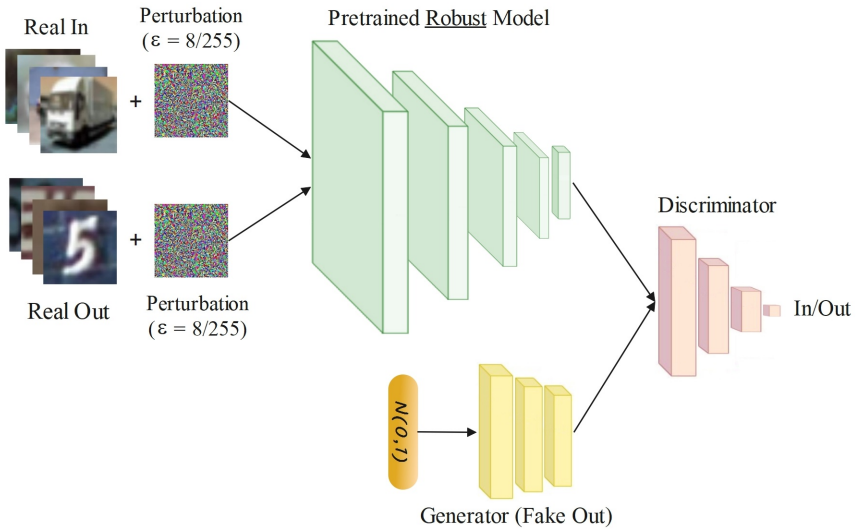

Figure 3: ATD schematic architecture. Generator, discriminator, and robust feature extractor are represented with yellow, pink, and green colors, respectively.

evaluations have yielded a false sense of robustness due to the weak evaluation attacks. In Fig. 2, we have evaluated ALOE and OSAD methods using an end-to-end PGD attack with $\epsilon = \frac{8}{255}$. According to the results, both the in and out detection score distributions remarkably change with this attack. So, these methods are not sufficiently robust. In the following, we aim to provide a more robust solution by addressing the drawbacks of previous works. We then use a toy example to provide more insights into the problem and our solution.

3.1 ATD: Adversarially Trained Discriminator

A robust OOD detection method must consider several factors to achieve robustness on both open and closed sets. There has been an extensive study of closed set robustness, which shows that variants of AT are the most effective defenses. Still, pure AT cannot provide robustness in OOD detection for two primary reasons. First, an AT model achieves a lower accuracy due to the trade-off between accuracy and robustness [16, 45]. Note that accuracy plays an important role in the OOD detection [11]. Second, the AT does not consider samples from the open set during training. ALOE tried to mitigate this issue using open datasets in training. However, this strategy works only when the open training data is a close proxy for the OOD data. Unfortunately, this is not true in most cases.

To resolve these issues, we consider OpenGAN as our baseline. Here, the discriminator performs the in/out binary classification. In this setting, a generator crafts OOD images to deceive the discriminator by images similar to the closed set. An ideal generator would cover a broader range of OOD distributions, which is a perfect setting for our problem. Furthermore, as the generator is trained to deceive the discriminator, the generated samples are naturally adversarial examples to the discriminator, which meets the needs in the AT. Hence no adversarial attack is need on such samples. Thus, we only need to perturb the closed set to adversarially train the discriminator. We call this method Adverarially Trained Discriminator (ATD), which can be summarized as:

$$\min_{G} \max_{\mathcal{D}} \mathbb{E}_{x \in D_{in}} \log \min_{\|x - x^*\|_\infty \leq \epsilon} \mathcal{D}(x^*) + \mathbb{E}_{z \in \mathcal{N}} \log(1 - \mathcal{D}(G(z))), \tag{8}$$

where the $\mathcal{D}$ and G are the discriminator and generator, respectively. Moreover, the inner minimization is approximated with the PGD attack and the outer minimization and maximization with SGD. Using adversarial perturbations to train GANs makes them more robust to adversarial examples, just like other architectures [47, 48]. On the other hand, adding perturbations to the closed set makes it difficult to optimize the discriminator. As a result, the generator would not be trained to cover a broad range since it could easily fool the discriminator in the min-max game for training GANs.

To get closer to the ideal generator that can cover a broad range of OOD distributions, unlike previously proposed robust GANs [47], we make the generator craft features instead of images. It is

evident that generating and discriminating the low-dimensional features is an easier problem than the high-dimensional images, which brings the generator closer to the ideal case. As a result, we also need a model that can extract robust features from the closed set. For this purpose, we use a pre-trained model with the HAT [46] method that achieves a satisfactory accuracy along with robustness on the closed set, considering the trade-off between accuracy and robustness [45]. The last layer, which could be thought of as a linear classifier, is excluded to obtain features instead of logits.

Finally, similar to the OpenGAN and ALOE, we utilize an open dataset in the training to stabilize the training. This data is also attacked during training since it is neither originally adversarial, nor is crafted by the generator. With these in mind, the objective function of the ATD can be reformulated as follows:

$$\begin{aligned} \min_{G} \max_{\mathcal{D}} \; & \mathbb{E}_{x \in D_{in}} \; \log \min_{\|x - x^*\|_\infty \leq \epsilon} \mathcal{D}(f_\theta(x^*)) + \\ & \alpha \, . \, \mathbb{E}_{x \in D_{out}} \; \log(1 - \max_{\|x - x^*\|_\infty \leq \epsilon} \mathcal{D}(f_\theta(x^*))) + \; (1 - \alpha) \, . \, \mathbb{E}_{z \in \mathcal{N}} \; \log(1 - \mathcal{D}(G(z))), \end{aligned} \tag{9}$$

where f_θ is the pre-trained robust feature extractor and α controls the weight of real open dataset in the optimization. Fig. 3 shows a schematic representation of ATD.

3.2 Toy Example

Here, we provide an insightful visualization of the ATD using a 2D toy example. To this end, we represent open and closed sets with disjoint distributions in a 2D space. Also, some samples are selected randomly in this space as the generated data. Fig. 4(a) shows the distributions used for this purpose, which the blue, orange, and green samples are closed, open, and generated data, respectively. Next, we train a multi-layer feed-forward neural network as the discriminator to perform the binary classification of in/out on these data in different cases. It should be noted that the open and closed data are fixed during the training, but the generated data is resampled in every epoch to simulate ATD training. Finally, the discriminator output is displayed all over the space with a blue (in) and orange (out) background in each case.

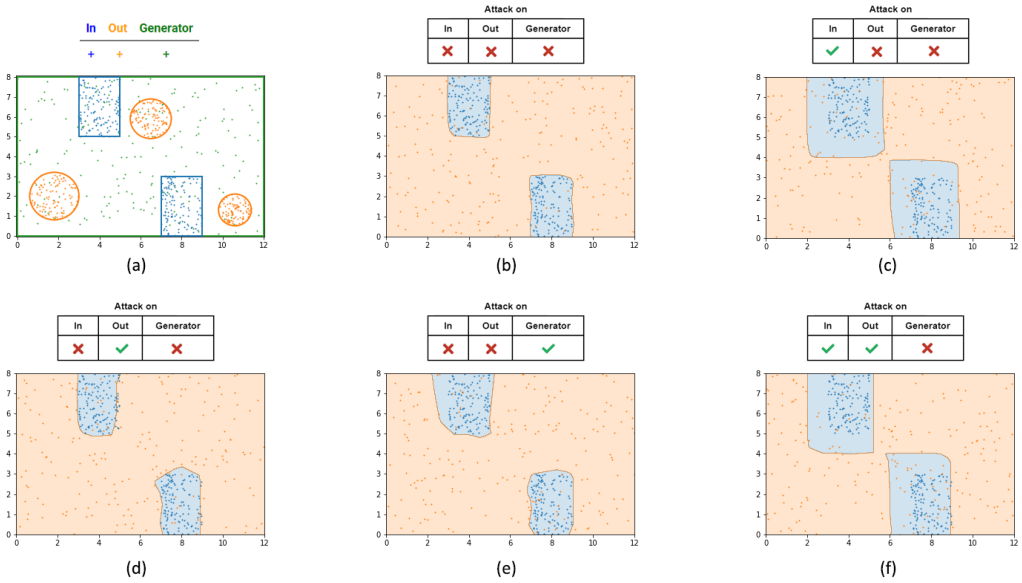

Figure 4: Two-dimensional example to illustrate the effect of ATD method. a) The in- and out-distributions samples that are considered in this example, which are represented with blue and orange colors, respecitvely. In and out samples are fixed, while the generated data colored in green is resampled at each epoch. b to f) Using these distributions, OOD classification has been trained with different attack settings and the classifier output is represented with blue (in) and orange (out) background all over the feature space. Above each plot, green ticks and red crosses indicate which data distributions are attacked.

In our analysis, different conditions are taken into account. First, standard training is conducted on the data without attacking any samples. The results in Fig. 4(b) show that the model has learned the classification decision boundaries very well. In each of the next three cases, one of the closed, open and generated sets is attacked during training with $\epsilon = 1$. Fig. 4(c) shows that attacking the closed set leads to a larger margin in the decision boundaries around them. Having this margin ensures robustness against attacks that seek to misclassify the in-distribution samples. Additionally, note that this margin is smaller in the right side that an open set is present, which shows the impact of using open sets during training. In Fig. 4(d), only the open set is attacked during the training. This has caused a tighter decision boundaries around the closed set where the open set is present. Therefore, using attacked open set in the training can slightly improve the robustness on OOD data, but it can simultaneously reduce the robustness on the closed set. As a result, α in Eq. 9 that controls the optimization weight of open set should be selected carefully. In contrast with the ATD method, we also attack the generated data to see how it affects the result. According to Fig 4(e), this does not cause robustness in the OOD data in contrast to Fig 4(d), which is another reason that ATD do not attack the generated data during training. It is noteworthy that the main cause for this observation may be the instability of the generated samples during the training, which is inevitable and makes the adversarial optimization goal infeasible.

Eventually, we attack both the open and closed sets during training as in ATD. The results in Fig. 4(f) show that the margin around the closed set increases in the directions that an open set is not present to provide robustness on both in and out samples, which can be considered as a mixture of results in Fig. 4(c) and 4(d). Also, this plot demonstrates that the robustness of the open and closed sets may be at odds with each other.

4 Experiments

In this section, we perform extensive experiments to evaluate existing OOD detection methods, including the standard and adversarially trained ones, and our ATD method against an end-to-end PGD attack. To this end, we first give details about the setting of the experiments. Next, we compare all the methods, which shows that ATD significantly outperforms the other methods. Toward the end, we conduct some additional experiments to investigate some aspects of our solution.

4.1 Setup

Detection Methods: Apart from the methods that use discriminators for the detection, other models are evaluated using MSP, OpenMax, MD and RMD as probability and distance based methods.

In-distribution Datasets: CIFAR-10 and CIFAR-100 [49] are used as the in-distribution datasets. The images pixel values are normalized to be in the range of 0 to 1.

Out-of-distribution Datasets: Following the setting in earlier works [23, 37], we use eight different datasets that are disjoint from the in-distribution sets, including MNIST [50], TinyImageNet [51], Places365 [52], LSUN [53], iSUN [54], Birds [55], Flowers [56], and COIL-100 [57] as the OOD test sets. The results are averaged over these datasets to peform a comprehensive evaluation on different OOD datasets. Also, the SVHN [58] dataset is used as the OOD validation set to select the best discriminator during the training, and Food-101 [59] is used as the open training set.

Baselines: Various methods have been considered in our comparisons. The ViT architecture and OpenGAN are used as the SOTA methods in the standard OOD detection. AT and HAT are considered as the effective defenses in the image classification, and AOE, ALOE, and OSAD as effective defenses in the OOD detection. Note that we are the first work that uses HAT for the OOD detection. All the defenses are trained with $\epsilon = \frac{8}{255}$ to have the best results against attack with this perturbation budget.

These baseline methods are trained based on the guidelines in their original work. It should be noted that OpenGAN evaluates the model in the training mode. This causes a kind of information leakage among the batch of images used to predict the OOD scores. Nevertheless, the evaluations of OpenGAN is done according to their guidelines. This problem has been solved in our implementations for ATD method and ATD is evaluated in the test mode.

ATD Hyperparameters: A simple DCGAN [60] is used for the generator and discriminator architecture in the ATD. Furthermore, ATD is trained for 20 epochs with $\alpha = 0.5$ using Adam [61] optimizer with learning rate of $1e-4$. Details of the ATD method is available in section 3.1.

Table 1: OOD detection AUROC under attack with $\epsilon = \frac{8}{255}$ for various methods trained with CIFAR-10 or CIFAR-100 as the closed set. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked. The best and second-best results are distinguished with bold and underlined text for each column.

Method	In-Distribution Dataset							
	CIFAR-10				CIFAR-100			
	Clean	In	Out	In and Out	Clean	In	Out	In and Out
OpenGAN-fea	0.971	0.473	0.425	0.266	**0.958**	0.198	0.324	0.088
OpenGAN-pixel	0.818	0.000	0.008	0.000	0.767	0.000	0.004	0.000
ViT (MSP)	0.975	0.448	0.172	0.002	0.879	0.269	0.129	0.002
ViT (MD)	**0.995**	0.136	0.495	0.000	<u>0.951</u>	0.053	0.279	0.000
ViT (RMD)	0.951	0.427	0.446	0.025	0.915	0.365	0.361	0.037
ViT (OpenMax)	<u>0.984</u>	0.346	0.291	0.004	0.907	0.086	0.166	0.001
AT (MSP)	0.735	0.462	0.442	0.174	0.603	0.324	0.250	0.085
AT (MD)	0.771	0.429	0.527	0.232	0.649	0.278	0.357	0.108
AT (RMD)	0.836	0.436	0.523	0.151	0.700	0.366	0.363	0.136
AT (OpenMax)	0.805	0.468	0.508	0.208	0.650	0.319	0.350	0.132
HAT (MSP)	0.770	0.560	0.548	0.325	0.612	0.393	0.335	0.176
HAT (MD)	0.789	0.572	0.586	0.369	0.810	<u>0.587</u>	<u>0.603</u>	<u>0.363</u>
HAT (RMD)	0.878	0.602	0.606	0.258	0.730	0.443	0.416	0.191
HAT (OpenMax)	0.821	0.613	<u>0.648</u>	<u>0.415</u>	0.703	0.462	0.462	0.263
OSAD (MSP)	0.698	0.411	0.407	0.154	0.557	0.285	0.194	0.055
OSAD (MD)	0.626	0.375	0.432	0.231	0.615	0.368	0.416	0.216
OSAD (RMD)	0.776	0.421	0.456	0.123	0.680	0.369	0.353	0.140
OSAD (OpenMax)	0.827	0.544	0.554	0.251	0.647	0.325	0.330	0.123
AOE (MSP)	0.780	0.544	0.527	0.285	0.566	0.332	0.324	0.157
AOE (MD)	0.709	0.361	0.484	0.215	0.743	0.406	0.539	0.255
AOE (RMD)	0.780	0.382	0.421	0.075	0.682	0.355	0.313	0.121
AOE (OpenMax)	0.797	0.528	0.586	0.298	0.591	0.282	0.356	0.143
ALOE (MSP)	0.843	<u>0.664</u>	0.538	0.287	0.701	0.438	0.317	0.127
ALOE (MD)	0.827	0.369	0.479	0.132	0.793	0.516	0.543	0.264
ALOE (RMD)	0.815	0.293	0.364	0.022	0.632	0.275	0.283	0.078
ALOE (OpenMax)	0.868	0.584	0.606	0.261	0.731	0.399	0.389	0.125
ATD (Ours)	0.943	**0.837**	**0.862**	**0.693**	0.877	**0.734**	**0.739**	**0.553**

Evaluation Attack: All the models are evaluated against an end-to-end PGD attack with $\epsilon = \frac{8}{255}$. For the baseline methods, we only use a 10-step attack, but ATD is evaluated with 100 steps to ensure its robustness. We show that this perturbation budget does not change the images semantically and 100 steps is more than enough to ensure the robustness of ATD in Appendix C and E. Also, the attack is performed with a single random restart and the random initialization in the range $(-\epsilon, \epsilon)$. Moreover, the attack step size is selected as $\alpha = 2.5 \times \frac{\epsilon}{N}$.

Evaluation Metric: We use AUROC as a well-known classification criterion. The AUROC value is in the range $[0, 1]$, and the closer it is to 1, the better the classifier performance.

4.2 Results

OOD detection under adversarial attack: To perform a comprehensive study, AUROC is computed in four different settings for each method. First, the standard OOD detection without any attack is conducted (Clean). Next, either in- or out-datasets are attacked (In/Out). Finally, both the in- and out-sets are attacked (In and Out). Note that resistance against the attack to both in- and out-sets is much harder than other cases since the perturbation budget has effectively been doubled. Based on the results in Table 1, ViT+MD and OpenGAN-fea have the best performance in the standard OOD

Table 2: ATD is trained on CIFAR-10 and is evaluated with the transferred attack on in/out data generated by baseline methods (columns). Results show that ATD is sufficiently robust under the transferred black-box attacks.

Attack from	OpenGAN-fea	ViT (MD)	AT (MD)	HAT (OM)	OSAD (OM)	AOE (OM)	ALOE (OM)
In	0.930	0.927	0.923	0.894	0.907	0.907	0.921
Out	0.940	0.940	0.933	0.914	0.927	0.925	0.933
In and Out	0.928	0.925	0.920	0.865	0.895	0.892	0.917

Table 3: Ablation study on our method. Other choices for the feature extractor, training method, and generated data attacking are tested on CIFAR-10, but none of them is as effective as the setting used in the ATD.

Config			Attack			
Feature Extractor	Adversarial Train	Attack Generator	Clean	In	Out	In and Out
Standard Trained	✓	✗	0.706	0.263	0.147	0.029
HAT	✗	✗	**0.947**	0.741	0.720	0.511
Not Used	✓	✓	0.916	0.776	0.758	0.546
Not Used	✓	✗	0.923	0.789	0.771	0.560
HAT	✓	✗	0.943	**0.837**	**0.862**	**0.693**

detection, but they completely fail under the adversarial setting. Early defenses improved adversarial performance at the cost of a decrease in the clean detection rate. Among them, HAT+OpenMax and HAT+MD are the most effective defenses in CIFAR-10 and CIFAR-100, respectively. Still, they are not as effective as our method ATD, which significantly outperforms them in both datasets while preserving the clean detection performance.

Black-box evaluation: To evaluate ATD under the black-box setting, we generate adversarial perturbations by attacking the introduced baselines as the source models and transferring them to the ATD as the target model. This test can also be considered as a sanity check for detecting gradient obfuscation in the model [62]. Since the single-step attack enjoys better transferability than the multi-step ones [41], we use FGSM for attacking the source models. Based on the results in Table 2, ATD is sufficiently robust against all the transferred attacks.

Ablation study: ATD uses HAT as the feature extractor and performs adversarial training on open and closed sets to robustify the discriminator. As an ablation study using CIFAR-10 as the in-distribution data, we replace HAT with a standard trained model to check its effectiveness. Also, in another experiment, we replace the discriminator adversarial training with the standard training. The results in Table 3 demonstrate that these settings are not as effective as ATD in achieving a robust detection model. As another ablation, we consider removing the feature extractor and generating images instead of features. Based on the results, the trained discriminator with this method is also not as robust as the ATD. Moreover, we check the effect of attacking the generated images, which leads to a lower AUROC score. This confirms that attacking generated data is not helpful in the training.

Classification attack instead of end-to-end attack: OSAD method is evaluated against an attack on the classification, and not the OOD detection, in their original work. Here, we show that this is not a strong attack on the detection method. This is done by checking whether an attack on classification or detection method is effective on the other or not. Results in Table 4 for the OSAD and ATD methods demonstrate that attacking the classification and detection is significantly more effective on their self than the other. Thus, an end-to-end attack is a better basis for the evaluation of the classification or detection defenses.

Comparison with ATOM: Our baselines include all the previous methods that consider robustness on both in- and out-of-distribution data to make a fair and comprehensive evaluation of our methods. ATOM method [19] is a defense that considers robustness only on the out-of-distribution data. Therefore, this method would not be robust against attacks on the in-distribution data. In addition, we have compared our results against ALOE, which can be regarded as an extension of ATOM that accounts for both in- and out-of-distribution robustness. Still, a comparison with the ATOM method is performed in the Table 5 using PGD-100 with $\epsilon = \frac{8}{255}$ that supports our arguments.

Table 4: Classification accuracy and OOD detection AUROC under attack to classification and detection methods for OSAD and ATD methods trained on CIFAR-10 dataset. An end-to-end attack is more effective in both classification and detection.

Evaluation	OSAD		ATD (Ours)	
	Attack Classification	Attack Detection	Attack Classification	Attack Detection
Classification Accuracy	0.419	0.777	0.622	0.847
Detection AUROC	0.813	0.544	0.918	0.837

Table 5: Comparison of AUROC score for OOD detection with ATD and ATOM methods under PGD-100 attack with $\epsilon = \frac{8}{255}$. CIFAR-10 and CIFAR-100 datasets are used as the in-distribution dataset. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Method	In-Distribution Dataset							
	CIFAR-10				CIFAR-100			
	Clean	In	Out	In and Out	Clean	In	Out	In and Out
ATOM	**0.983**	0.156	0.447	0.067	**0.925**	0.089	0.728	0.085
ATD (Ours)	0.943	**0.837**	**0.862**	**0.693**	0.877	**0.734**	**0.739**	**0.553**

5 Conclusion

The existing OOD detection methods are far from being robust against strong attacks, contrary to what has been claimed previously for methods such as ALOE and OSAD. To mitigate this issue, we proposed ATD that uses an adversarially trained discriminator to classify in and out samples. Moreover, ATD utilizes HAT to extract robust features from the real samples, and a generator to craft a broad range of OOD data features. This method could significantly outperform earlier methods against an end-to-end PGD attack. Also, it is sufficiently robust against black-box attacks. The primary advantage of ATD is that it preserves the standard OOD detection AUROC along with the robustness, which is needed in real-world situations.

Broader Impact

Detecting out-of-distribution inputs is a safety issue in many machine learning models. In spite of advances in this area, the existing models are susceptible to adversarial examples as another important safety problem. This work aims at overcoming this issue by detecting OOD inputs even when they have been perturbed by adversarial threat models. We believe that this is important in most machine learning systems with safety-critical concerns. For instance, it is required to diagnose an unseen disease in health care systems or to detect anomalous patterns in financial services even if an adversary has perturbed their inputs. Therefore, this work can benefit a wide range of machine learning researchers. Also, we do not expect our efforts to have any negative consequences.

Acknowledgments

We thank Mahdi Amiri, Hossein Mirzaei, Zeinab Golgooni, and the anonymous reviewers for their helpful discussions and feedbacks on this work.

References

[1] Iqbal H Sarker. Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, 2(3):1–21, 2021.

[2] Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232, 2019.

[3] Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Josh Tenenbaum, Bill Freeman, and Jiajun Wu. Learning to reconstruct shapes from unseen classes. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[4] Mohammadreza Salehi, Atrin Arya, Barbod Pajoum, Mohammad Otoofi, Amirreza Shaeiri, Mohammad Hossein Rohban, and Hamid R Rabiee. Arae: Adversarially robust training of autoencoders improves novelty detection. *Neural Networks*, 144:726–736, 2021.

[5] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21464–21475. Curran Associates, Inc., 2020.

[6] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning placeholders for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2021.

[7] Mohammadreza Salehi, Hossein Mirzaei, Dan Hendrycks, Yixuan Li, Mohammad Hossein Rohban, and Mohammad Sabokrou. A unified survey on anomaly, novelty, open-set, and out-of-distribution detection: Solutions and future challenges. *arXiv preprint arXiv:2110.14051*, 2021.

[8] Yifei Ming, Ying Fan, and Yixuan Li. Poem: Out-of-distribution detection with posterior sampling. In *International Conference on Machine Learning*, pages 15650–15665. PMLR, 2022.

[9] Hongjie Zhang, Ang Li, Jie Guo, and Yanwen Guo. Hybrid models for open set recognition. In *European Conference on Computer Vision*, pages 102–117. Springer, 2020.

[10] Guangyao Chen, Limeng Qiao, Yemin Shi, Peixi Peng, Jia Li, Tiejun Huang, Shiliang Pu, and Yonghong Tian. Learning open set network with discriminative reciprocal points. In *European Conference on Computer Vision*, pages 507–522. Springer, 2020.

[11] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *International Conference on Learning Representations*, 2022.

[12] 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.

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

[14] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018.

[15] 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.

[16] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.

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

[18] Mohammad Azizmalayeri and Mohammad Hossein Rohban. Lagrangian objective function leads to improved unforeseen attack generalization in adversarial training. *arXiv preprint arXiv:2103.15385*, 2021.

[19] Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Atom: Robustifying out-of-distribution detection using outlier mining. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 430–445. Springer, 2021.

[20] Stanislav Fort. Adversarial vulnerability of powerful near out-of-distribution detection. *arXiv preprint arXiv:2201.07012*, 2022.

[21] Maximilian Augustin, Alexander Meinke, and Matthias Hein. Adversarial robustness on in-and out-distribution improves explainability. In *European Conference on Computer Vision*, pages 228–245. Springer, 2020.

[22] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

[23] Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Robust out-of-distribution detection for neural networks. In *The AAAI-22 Workshop on Adversarial Machine Learning and Beyond*, 2022.

[24] Rui Shao, Pramuditha Perera, Pong C Yuen, and Vishal M Patel. Open-set adversarial defense. In *European Conference on Computer Vision*, pages 682–698. Springer, 2020.

[25] Rui Shao, Pramuditha Perera, Pong C Yuen, and Vishal M Patel. Open-set adversarial defense with clean-adversarial mutual learning. *International Journal of Computer Vision*, 130(4):1070–1087, 2022.

[26] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer, 2017.

[27] Yezheng Liu, Zhe Li, Chong Zhou, Yuanchun Jiang, Jianshan Sun, Meng Wang, and Xiangnan He. Generative adversarial active learning for unsupervised outlier detection. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1517–1528, 2019.

[28] Shu Kong and Deva Ramanan. OpenGAN: Open-set recognition via open data generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 813–822, 2021.

[29] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

[30] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018.

[31] 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. In *International Conference on Learning Representations*, 2021.

[32] Mohammad Azizmalayeri and Mohammad Hossein Rohban. OOD augmentation may be at odds with open-set recognition. *arXiv preprint arXiv:2206.04242*, 2022.

[33] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016.

[34] Walter J Scheirer, Anderson Rocha, Ross J Micheals, and Terrance E Boult. Meta-recognition: The theory and practice of recognition score analysis. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1689–1695, 2011.

[35] 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.

[36] Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*, 2021.

[37] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

[38] Zongyuan Ge, Sergey Demyanov, and Rahil Garnavi. Generative openmax for multi-class open set classification. In *British Machine Vision Conference (BMVC)*, 2017.

[39] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 613–628, 2018.

[40] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018.

[41] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018.

[42] 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*, pages 2574–2582, 2016.

[43] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[44] 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*, pages 484–501. Springer, 2020.

[45] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

[46] Rahul Rade and Seyed-Mohsen Moosavi-Dezfooli. Reducing excessive margin to achieve a better accuracy vs. robustness trade-off. In *International Conference on Learning Representations*, 2022.

[47] Xuanqing Liu and Cho-Jui Hsieh. Rob-gan: Generator, discriminator, and adversarial attacker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11234–11243, 2019.

[48] Desheng Wang, Weidong Jin, Yunpu Wu, and Aamir Khan. Improving global adversarial robustness generalization with adversarially trained gan. *arXiv preprint arXiv:2103.04513*, 2021.

[49] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[50] Yann LeCun and Corinna Cortes. Mnist handwritten digit database. `http://yann.lecun.com/exdb/mnist/`, 2010.

[51] 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*, pages 248–255. Ieee, 2009.

[52] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

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

[54] Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R Kulkarni, and Jianxiong Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755*, 2015.

[55] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010.

[56] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[57] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al. Columbia object image library (coil-100). 1996.

[58] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[59] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014.

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

[61] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[62] 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*, pages 274–283. PMLR, 2018.

[63] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.

[64] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, 2020.

[65] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019.

[66] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*, 2018.

[67] Julian Bitterwolf, Alexander Meinke, and Matthias Hein. Certifiably adversarially robust detection of out-of-distribution data. *Advances in Neural Information Processing Systems*, 33:16085–16095, 2020.

[68] Alexander Meinke, Julian Bitterwolf, and Matthias Hein. Provably robust detection of out-of-distribution data (almost) for free. *arXiv preprint arXiv:2106.04260*, 2021.

[69] Hossein Mirzaei, Mohammadreza Salehi, Sajjad Shahabi, Efstratios Gavves, Cees GM Snoek, Mohammad Sabokrou, and Mohammad Hossein Rohban. Fake it till you make it: Near-distribution novelty detection by score-based generative models. *arXiv preprint arXiv:2205.14297*, 2022.

[70] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.1.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Error bars were too small to have any visual impact.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We have used a single RTX 2060 Super GPU.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] Data are publicly available, which have been cited.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix

A Smaller Perturbation Budget

In Table 1, we reported the AUROC for the OOD detection for the perturbed in/out distributions by the PGD attack with $\epsilon = \frac{8}{255}$. Here, a similar evaluation is conducted with half the perturbation budget. The results are displayed in Table 6. Similar to Table 1, HAT is the best method among the previous defenses, and ATD outperforms it with a significant margin. ATD achieves satisfactory robustness in all of the attack settings in both datasets, along with a decent clean detection AUROC score.

It should be noted that attacking both in and out sets is included in the experiments to give a sense of how the distribution of OOD scores changes under adversarial attacks, but evaluating the AUROC when perturbing either in or out set is more reasonable in practice.

Table 6: OOD detection AUROC under PGD attack with $\epsilon = \frac{4}{255}$ for various methods trained with CIFAR-10 or CIFAR-100 as the closed set. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked. The best and second-best results are distinguished with bold and underlined text for each column.

Method	In-Distribution Dataset							
	CIFAR-10				CIFAR-100			
	Clean	In	Out	In and Out	Clean	In	Out	In and Out
OpenGAN-fea	0.971	0.521	0.444	0.318	**0.958**	0.283	0.382	0.123
OpenGAN-pixel	0.818	0.001	0.052	0.000	0.767	0.001	0.026	0.000
ViT (MSP)	0.975	0.578	0.280	0.013	0.879	0.362	0.157	0.005
ViT (MD)	**0.995**	0.260	0.571	0.004	<u>0.951</u>	0.094	0.327	0.001
ViT (RMD)	0.951	0.526	0.493	0.043	0.915	0.366	0.371	0.041
ViT (OpenMax)	<u>0.984</u>	0.339	0.355	0.007	0.907	0.164	0.205	0.003
AT (MSP)	0.735	0.573	0.564	0.394	0.603	0.416	0.375	0.226
AT (MD)	0.771	0.603	0.650	0.473	0.649	0.454	0.495	0.310
AT (RMD)	0.836	0.640	0.690	0.453	0.700	0.512	0.511	0.329
AT (OpenMax)	0.805	0.647	0.669	0.489	0.650	0.469	0.491	0.326
HAT(MSP)	0.770	0.665	0.658	0.541	0.612	0.484	0.457	0.339
HAT(MD)	0.789	0.691	0.688	0.579	0.810	<u>0.712</u>	<u>0.711</u>	<u>0.596</u>
HAT (RMD)	0.878	<u>0.740</u>	<u>0.764</u>	0.573	0.730	0.579	0.562	0.402
HAT (OpenMax)	0.821	0.729	0.741	<u>0.633</u>	0.703	0.587	0.586	0.464
OSAD (MSP)	0.698	0.521	0.516	0.347	0.557	0.346	0.295	0.157
OSAD (MD)	0.626	0.500	0.521	0.402	0.615	0.491	0.510	0.389
OSAD (RMD)	0.776	0.576	0.619	0.384	0.680	0.500	0.495	0.321
OSAD (OpenMax)	0.827	0.696	0.699	0.535	0.647	0.476	0.478	0.318
AOE (MSP)	0.780	0.658	0.654	0.514	0.566	0.435	0.430	0.313
AOE (MD)	0.709	0.515	0.587	0.407	0.743	0.535	0.636	0.437
AOE (RMD)	0.780	0.562	0.595	0.337	0.682	0.464	0.468	0.270
AOE (OpenMax)	0.797	0.675	0.702	0.559	0.591	0.427	0.468	0.315
ALOE (MSP)	0.843	0.667	0.612	0.373	0.701	0.489	0.440	0.258
ALOE (MD)	0.827	0.602	0.646	0.406	0.793	0.623	0.656	0.466
ALOE (RMD)	0.815	0.512	0.570	0.213	0.632	0.398	0.402	0.207
ALOE (OpenMax)	0.869	0.658	0.679	0.422	0.731	0.483	0.509	0.282
ATD (Ours)	0.943	**0.915**	**0.919**	**0.883**	0.877	**0.818**	**0.817**	**0.742**

Table 7: Comparing OOD detection methods robustness. For each of MSP, MD, RMD, and OpenMax, the AUROC is averaged over six different baselines in Table 1 that use these detection methods. A clean evaluation is the one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Method	In-Distribution Dataset							
	CIFAR-10				CIFAR-100			
	Clean	In	Out	In and Out	Clean	In	Out	In and Out
MSP	0.800	0.515	0.439	0.204	0.653	0.340	0.258	0.100
MD	0.786	0.374	0.500	0.196	**0.760**	**0.368**	**0.456**	**0.201**
RMD	0.839	0.427	0.469	0.109	0.723	0.362	0.348	0.117
OpenMax	**0.850**	**0.514**	**0.532**	**0.239**	0.705	0.312	0.342	0.131

Figure 5: Images from the open and closed sets before and after the attack with $\epsilon = \frac{8}{255}$. HAT+MSP is used as the base model in attacking the images. The images are perceptually similar before and after the attack. Also, the OOD detection score for each image is displayed above it, which demonstrates the effectiveness of the attack.

B Detection Method

To investigate the effect of the detection method, we averaged the results in Table 1 over all the baselines (ViT, AT, HAT, OSAD, AOE, and ALOE) for MSP, MD, RMD, and OpenMax. According to the results in Table 7, OpenMax and MD are the best choices when considering CIFAR-10 and CIFAR-100 as the closed set, respectively.

C Visualization of Adversarial Images

A number of the original and adversarial images for the open and closed sets are shown in Fig. 5. Clearly, images have not changed perceptually under the attack with $\epsilon = \frac{8}{255}$, while the detection score has changed significantly.

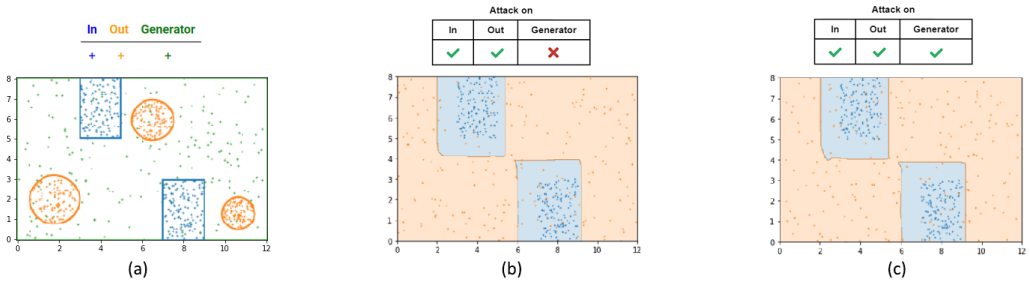

Figure 6: Effect of attacking generated data along with the in and out sets.

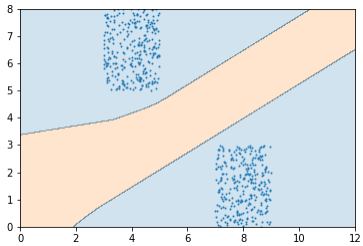

Figure 7: OOD detection with an adversarially trained model using MSP as the detection method. Each of the rectangular distributions are regarded as a separate class, and adversarial training is conducted on them. Next, OOD scores are extracted with the MSP method all over the space, and thresholded to achieve the in/out decision boundaries. The blue and orange backgrounds are regarded as the in- and out-distributions, respectively.

D Toy Example Exploration

In section 3.2, a toy example was investigated with different attack settings. Here, we use this toy example to investigate two other configurations. First, we use a similar setting to check the effect of attacking the generated data along with the closed and open sets. The results in Fig. 6 demonstrate that attacking the generated data during training does not change the decision boundaries with the setting in this example. As mentioned earlier, we believe that attacking the generator should not be done in practice due to the inevitable instability in training of the GAN.

In the second configuration, we use this toy example to simulate the pure adversarial training with MSP detection method in OOD detection. To this end, we only consider the in-distribution data in training. Each of the blue rectangles in Fig. 7 are regarded as a separate class, and adversarial training is performed on them. Next, MSP is used as the score function all over the considered space. Finally, the scores are separated with a threshold to detect OOD samples. The in/out classification decision boundaries are displayed in Fig. 7, which demonstrates that pure AT is not as effective as ATD for in/out robust classification due to the lack of OOD samples in the training.

E Number of Attack Steps

To check the effect of increasing the attack steps, we plotted the AUROC score with various attack steps for HAT, OSAD, and ATD. According to Fig. 8, 10 steps are enough for evaluation of HAT and OSAD, and 50 steps are enough for ATD. Note that we have used attack with 100 steps in the evaluations of ATD to ensure its robustness.

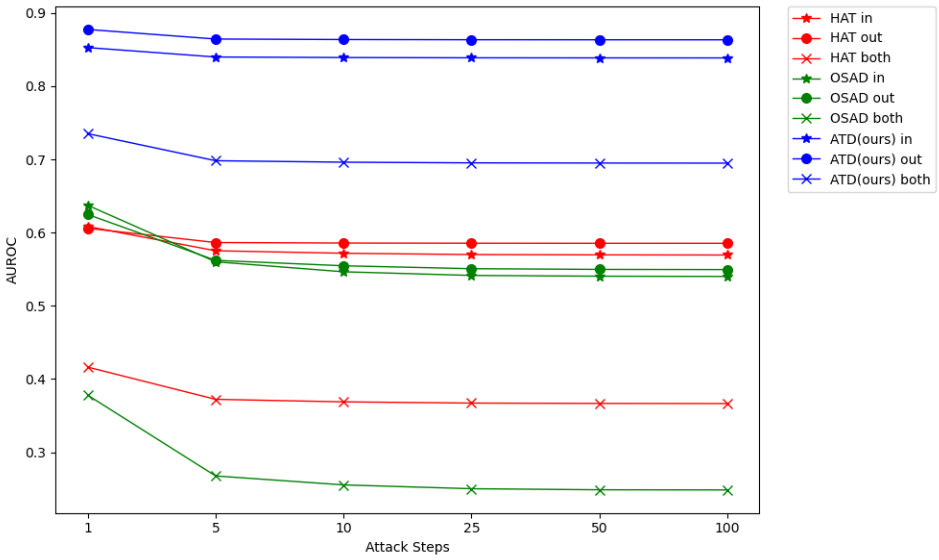

Figure 8: Effect of the number of PGD attack steps on the detection AUROC in HAT, OSAD, and ATD methods trained with CIFAR-10 as the closed set. Three attacking scenarios are considered: attacking only the in data, attacking only the out data, and attacking both in and out data. Attacks are conducted with 1, 5, 10, 25, 50, and 100 steps. The AUROC reaches a stable value after 50 steps for all the methods and scenarios.

Table 8: Comparison of AUROC score for OOD detection with ATD method under PGD-100 attack and AutoAttack with $\epsilon = \frac{8}{255}$. CIFAR-10 and CIFAR-100 datasets are used as the in-distribution dataset. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Attack	In-Distribution Dataset							
	CIFAR-10				CIFAR-100			
	Clean	In	Out	In and Out	Clean	In	Out	In and Out
PGD-100	0.943	0.837	0.862	0.693	0.877	0.734	0.739	0.553
Auto Attack	0.943	0.834	0.861	0.688	0.877	0.729	0.737	0.545

F Evaluation With AutoAttack

Despite the high power of the PGD attack in assessing the robustness, new attacks have been proposed recently that aim to better evaluate the robustness with stronger attacks. Among them, AutoAttack [64] has become a more reliable substitute to PGD and have proven to be much more difficult to be fooled. AutoAttack propose to use an ensemble of four diverse attacks to reliably evaluate the robustness. These attacks are run sequentially on an input batch, and the one that causes the most loss is used for perturbing the input data. To ensure the reliability of our results, we have evaluated our method against such a strong attack with $\epsilon = \frac{8}{255}$ in addition to PGD-100 in the Table 8. The results indicate that the model is sufficiently robust against such a strong attack, and the results are still promising.

G TinyImageNet Dataset

Adversarial robustness on larger and sophisticated datasets is still a challenging issue, even in closed set classification. For instance, the adversarial accuracy on the closed set in the TinyImageNet dataset ($\epsilon = \frac{8}{255}$) is less than 20% [46]. On the other hand, closed set accuracy significantly affects OOD

Table 9: OOD detection AUROC under PGD attack with $\epsilon = \frac{4}{255}$ for ViT, AT, HAT, ALOE, and ATD trained with TinyImageNet as the closed set. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Method	Clean	In	Out	In and Out
ViT (MSP)	0.924	0.229	0.040	0.000
ViT (MD)	**0.954**	0.198	0.208	0.000
ViT (RMD)	0.934	0.204	0.250	0.001
ViT (OpenMax)	0.915	0.216	0.031	0.000
AT (MSP)	0.557	0.313	0.309	0.127
AT (MD)	0.397	0.160	0.208	0.055
AT (RMD)	0.522	0.254	0.263	0.086
AT (OpenMax)	0.536	0.272	0.305	0.112
HAT (MSP)	0.563	0.377	0.376	0.200
HAT (MD)	0.676	0.498	0.495	0.309
HAT (RMD)	0.710	0.479	0.419	0.205
HAT (OpenMax)	0.574	0.381	0.385	0.205
ALOE (MSP)	0.575	0.233	0.080	0.005
ALOE (MD)	0.689	0.083	0.350	0.002
ALOE (RMD)	0.500	0.136	0.087	0.008
ALOE (OpenMax)	0.569	0.114	0.120	0.002
ATD (Ours)	0.883	**0.786**	**0.781**	**0.635**

detection [11]. As a result, adversarially robust OOD detection on large-scale datasets would be much harder since the robust closed set classification itself is still an open challenging issue.

Still, we run an experiment on TinyImageNet as the in-distribution dataset that compares our method with ViT, ALOE, HAT, and AT methods. AT, HAT, and ALOE are evaluated against PGD-10 with $\epsilon = \frac{4}{255}$, while PGD-100 with $\epsilon = \frac{4}{255}$ is used for ViT and ATD. The training hyper-parameters are similar to the setup in section 4.1. Also, TinyImageNet is excluded from the OOD datasets. The results are listed in Table 9 that demonstrate our method's robustness.

H Certified Detection

Recently, several works have tried to tackle the issue of unreliable evaluations of robustness in neural networks by providing provable guarantees. This is not limited to OOD detection, and it was first noticed in image classification methods. For instance, a provable bound on the confidence score of neural network was provided for image classification methods using randomized smoothing method [65], or a lower and upper bound was provided by IBP [66] for the output of each layer in the neural network given that input x is varied in the ℓ_∞ ball of radius ϵ. Similar efforts have been made in the OOD detection field. For instance, GOOD [67] and ProoD [68] use a similar method to IBP to provide certified bounds.

Despite the advances in the certified robustness research field, the certified defenses still cannot compete with the adversarially trained models against commonly perturbation budget $\epsilon = \frac{8}{255}$ since their goal is to provide provable bounds in specific conditions based on the input itself and the adversary perturbation budget, and the model is not optimized only on the adversarial examples.

I OOD Detection Details

In the experiments, CIFAR-10 and CIFAR-100 are considered the in-distribution datasets, while OOD datasets include MNIST, TinyImageNet, Places365, LSUN, iSUN, Birds, Flowers, and COIL-100. Next, the results are averaged over all the OOD datasets and reported in Table 1.

In this section, the details of the experiments are provided for each OOD dataset in the Tables 10 and 11 for CIFAR-10 and CIFAR-100, respectively. For the baseline methods which are evaluated

with different detection methods such as MSP, MD, RMD, and Openmax, only the one with the best average across all the "Clean", "In", "Out", "In and Out" cases is chosen. Based on these tables, our method is more robust than other methods, even in a single OOD dataset in addition to the average case. Hence, we can conclude that our method is more robust than other baselines across a wide range of OOD datasets.

Note that some of the OOD datasets are somewhat similar to the in-distribution data (e.g. Birds and the bird class from CIFAR-10). These datasets can be regarded as the near-OOD data that the model is expected to detect them as well as the other OODs. Therefore, averaging across all these datasets helps to evaluate the models even on the near-OOD data [69]. In addition, in Tables 10 and 11, CIFAR-10 and CIFAR-100 are also considered as the OOD dataset for the other one, which can be regarded as another case of the near-OOD study [70].

J Computational Cost

Using a single RTX 2060 Super GPU and the setup mentioned in the Section 4.1, our model training and evaluation last for 5 and 2.75 hours, respectively. The time for pretraining the feature extractor model is not included in the mentioned times, which will last about 9 hours in total. Therefore, training ATD would be feasible in a reasonable time, even for large datasets. The evaluation time of the other baseline methods is also similar when using the same setting and MSP detector. This is because they all need to obtain features from the image through a convolutional backbone, which takes most of the evaluation time. Furthermore, when using distance-based detectors for the baseline methods, the evaluation time will increase by five times since they require fitting a class conditional distribution to the pre-logit features.

Table 10: OOD detection AUROC for each of OOD datasets under PGD attack with $\epsilon = \frac{8}{255}$ for various methods trained with CIFAR-10 as the closed set. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Method	Attacked Distribution	Out-Distribution Dataset								
		MNIST	TiImgNet	Places	LSUN	iSUN	Birds	Flower	COIL	CIFAR100
OpenGAN-fea	Clean	0.994	0.953	0.950	0.965	0.963	0.983	0.983	0.981	0.950
	In	0.631	0.363	0.392	0.434	0.425	0.520	0.489	0.527	0.316
	Out	0.486	0.433	0.363	0.420	0.419	0.408	0.387	0.481	0.296
	In and Out	0.303	0.152	0.173	0.240	0.230	0.345	0.301	0.385	0.101
ViT (RMD)	Clean	0.987	0.952	0.983	0.984	0.986	0.760	0.996	0.959	0.973
	In	0.393	0.457	0.491	0.551	0.544	0.099	0.452	0.429	0.443
	Out	0.550	0.408	0.481	0.401	0.396	0.235	0.480	0.621	0.364
	In and Out	0.035	0.023	0.031	0.020	0.021	0.006	0.026	0.039	0.017
AT (OpenMax)	Clean	0.804	0.810	0.825	0.850	0.839	0.751	0.855	0.703	0.796
	In	0.488	0.460	0.482	0.507	0.495	0.403	0.512	0.397	0.444
	Out	0.729	0.455	0.493	0.509	0.507	0.419	0.530	0.426	0.439
	In and Out	0.396	0.165	0.189	0.196	0.195	0.147	0.209	0.166	0.160
HAT (OpenMax)	Clean	0.750	0.831	0.847	0.880	0.857	0.741	0.888	0.776	0.805
	In	0.538	0.616	0.641	0.677	0.650	0.516	0.690	0.573	0.579
	Out	0.669	0.647	0.678	0.707	0.679	0.537	0.687	0.581	0.606
	In and Out	0.452	0.406	0.438	0.458	0.435	0.320	0.441	0.367	0.368
OSAD (OpenMax)	Clean	0.862	0.819	0.833	0.864	0.840	0.765	0.886	0.750	0.799
	In	0.665	0.498	0.538	0.566	0.532	0.419	0.653	0.475	0.474
	Out	0.794	0.506	0.528	0.533	0.525	0.504	0.574	0.469	0.484
	In and Out	0.553	0.193	0.212	0.207	0.203	0.191	0.266	0.187	0.181
AOE (OpenMax)	Clean	0.584	0.820	0.877	0.922	0.902	0.798	0.723	0.749	0.782
	In	0.287	0.539	0.620	0.680	0.648	0.528	0.419	0.500	0.492
	Out	0.496	0.571	0.650	0.699	0.670	0.575	0.456	0.571	0.530
	In and Out	0.225	0.283	0.345	0.380	0.357	0.287	0.198	0.310	0.252
ALOE (MSP)	Clean	0.746	0.821	0.851	0.987	0.983	0.799	0.790	0.768	0.788
	In	0.463	0.609	0.661	0.936	0.922	0.591	0.585	0.541	0.458
	Out	0.521	0.470	0.478	0.757	0.743	0.463	0.437	0.434	0.476
	In and Out	0.227	0.216	0.228	0.516	0.504	0.218	0.196	0.193	0.170
ATD (Ours)	Clean	0.988	0.880	0.925	0.960	0.948	0.936	0.997	0.908	0.820
	In	0.938	0.685	0.795	0.861	0.840	0.839	0.983	0.757	0.580
	Out	0.977	0.726	0.813	0.879	0.857	0.853	0.985	0.808	0.641
	In and Out	0.902	0.470	0.607	0.690	0.668	0.690	0.937	0.581	0.380

Table 11: OOD detection AUROC for each of OOD datasets under PGD attack with $\epsilon = \frac{8}{255}$ for various methods trained with CIFAR-100 as the closed set. A clean evaluation is one where no attack is made on the data, whereas an in/out evaluation means that the corresponding data is attacked.

Method	Attacked Distribution	Out-Distribution Dataset								
		MNIST	TiImgNet	Places	LSUN	iSUN	Birds	Flower	COIL	CIFAR10
OpenGAN-fea	Clean	0.990	0.883	0.945	0.971	0.964	0.966	0.968	0.977	0.929
	In	0.267	0.134	0.169	0.217	0.205	0.195	0.178	0.219	0.299
	Out	0.493	0.149	0.223	0.331	0.347	0.268	0.347	0.438	0.276
	In and Out	0.146	0.039	0.049	0.073	0.075	0.074	0.093	0.157	0.091
ViT (RMD)	Clean	0.838	0.901	0.923	0.916	0.914	0.978	0.966	0.881	0.948
	In	0.271	0.359	0.325	0.340	0.336	0.516	0.495	0.280	0.386
	Out	0.192	0.335	0.419	0.199	0.252	0.687	0.454	0.351	0.485
	In and Out	0.017	0.031	0.037	0.006	0.009	0.105	0.055	0.035	0.058
AT (RMD)	Clean	0.411	0.723	0.731	0.760	0.725	0.731	0.776	0.744	0.675
	In	0.178	0.369	0.386	0.404	0.375	0.373	0.449	0.396	0.320
	Out	0.353	0.329	0.347	0.354	0.326	0.360	0.411	0.425	0.295
	In and Out	0.142	0.120	0.127	0.129	0.119	0.134	0.157	0.163	0.107
HAT (MD)	Clean	0.992	0.662	0.780	0.826	0.787	0.829	0.879	0.723	0.540
	In	0.962	0.379	0.521	0.583	0.531	0.598	0.676	0.444	0.268
	Out	0.988	0.389	0.524	0.580	0.523	0.585	0.708	0.525	0.291
	In and Out	0.943	0.161	0.253	0.292	0.254	0.318	0.429	0.254	0.108
OSAD (MD)	Clean	0.959	0.483	0.557	0.556	0.548	0.545	0.696	0.575	0.503
	In	0.865	0.239	0.293	0.279	0.280	0.270	0.429	0.286	0.261
	Out	0.942	0.266	0.320	0.306	0.299	0.313	0.494	0.384	0.277
	In and Out	0.820	0.099	0.121	0.104	0.106	0.110	0.229	0.140	0.103
AOE (MD)	Clean	0.982	0.540	0.687	0.740	0.724	0.747	0.837	0.690	0.443
	In	0.715	0.211	0.332	0.386	0.385	0.418	0.493	0.312	0.149
	Out	0.969	0.299	0.428	0.482	0.460	0.534	0.669	0.472	0.236
	In and Out	0.684	0.096	0.151	0.192	0.187	0.255	0.322	0.154	0.061
ALOE (MD)	Clean	0.966	0.581	0.750	0.831	0.801	0.784	0.851	0.779	0.436
	In	0.806	0.235	0.424	0.596	0.557	0.503	0.579	0.430	0.131
	Out	0.947	0.265	0.439	0.506	0.488	0.513	0.647	0.535	0.168
	In and Out	0.731	0.063	0.141	0.207	0.221	0.237	0.318	0.192	0.030
ATD (Ours)	Clean	0.973	0.737	0.833	0.892	0.865	0.934	0.972	0.806	0.575
	In	0.903	0.497	0.659	0.735	0.704	0.835	0.921	0.619	0.320
	Out	0.959	0.492	0.636	0.724	0.688	0.834	0.909	0.667	0.322
	In and Out	0.863	0.260	0.417	0.494	0.473	0.662	0.801	0.453	0.138