ADVERSARIALLY ROBUST OUT-OF-DISTRIBUTION DETECTION USING LYAPUNOV-STABILIZED EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Despite significant advancements in out-of-distribution (OOD) detection, existing methods still struggle to maintain robustness against adversarial attacks, compromising their reliability in critical real-world applications. Previous studies have attempted to address this challenge by exposing detectors to auxiliary OOD datasets alongside adversarial training. However, the increased data complexity inherent in adversarial training, and the myriad of ways that OOD samples can arise during testing, often prevent these approaches from establishing robust decision boundaries. To address these limitations, we propose AROS, a novel approach leveraging neural ordinary differential equations (NODEs) with Lyapunov stability theorem in order to obtain robust embeddings for OOD detection. By incorporating a tailored loss function, we apply Lyapunov stability theory to ensure that both in-distribution (ID) and OOD data converge to stable equilibrium points within the dynamical system. This approach encourages any perturbed input to return to its stable equilibrium, thereby enhancing the model's robustness against adversarial perturbations. To not use additional data, we generate fake OOD embeddings by sampling from low-likelihood regions of the ID data feature space, approximating the boundaries where OOD data are likely to reside. To then further enhance robustness, we propose the use of an orthogonal binary layer following the stable feature space, which maximizes the separation between the equilibrium points of ID and OOD samples. We validate our method through extensive experiments across several benchmarks, demonstrating superior performance, particularly under adversarial attacks. Notably, our approach improves robust detection performance from 37.8% to **80.1%** on CIFAR-10 vs. CIFAR-100 and from 29.0% to **67.0%** on CIFAR-100 vs. CIFAR-10.

1 INTRODUCTION

Deep neural networks have demonstrated remarkable success in computer vision, achieving significant results across a wide range of tasks. However, these models are vulnerable to adversarial examples—subtly altered inputs that can lead to incorrect predictions (1; 2; 3). As a result, designing a defense mechanism has emerged as a critical task. Various strategies have been proposed, and adversarial training has become one of the most widely adopted approaches (4; 5; 6). Recently, Neural Ordinary Differential Equations (NODEs) have attracted attention as a defense strategy by leveraging principles from control theory. By leveraging the dynamical system properties of NODEs, and imposing stability constraints, these methods aim to enhance robustness with theoretical guarantees. However, they have been predominantly studied in the context of classification tasks (7; 8; 9; 10; 11; 12; 13; 14; 15), and not in out-of-distribution (OOD) detection.

OOD detection is a safety-critical task that is crucial for deploying models in the real world. In this task, training is limited to in-distribution (ID) data, while the inference task involves identifying OOD samples, i.e., samples that deviate from the ID data (16; 17). Recent advancements have demonstrated impressive performance gains across various detection benchmarks (18; 19; 20; 21). However, a significant challenge arises concerning the robustness of OOD detectors against adversarial attacks. An adversarial attack on a detector involves introducing minor perturbations to test samples, causing the detector to predict OOD as ID samples or vice versa. Yet, a robust OOD detector is imperative, especially in scenarios like medical diagnostics and autonomous driving (22; 23; 24; 25; 26).

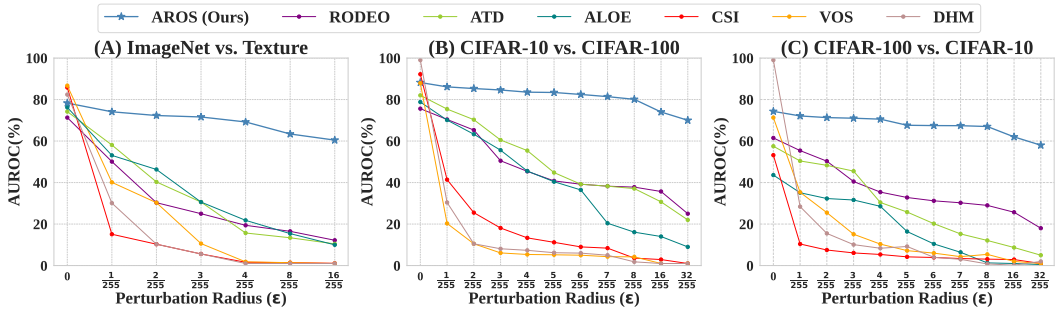

Figure 1: **OOD detection performance for various models under different perturbation magnitudes.** The perturbations are generated using PGD^{1000} (ℓ_∞) attack targeting both test ID and OOD samples. (A) ImageNet is used as the ID dataset, while the Texture dataset is used as the OOD during test time. (B) CIFAR-10 is utilized as the ID, with CIFAR-100 as the OOD. (C) CIFAR-100 is used as the ID, with CIFAR-10 as the OOD. A perfect detector achieves an AUROC of 100%, a random detector scores 50%, and a fully compromised detector under attack scores 0%. Notably, no other model achieves detection performance **above random** (i.e., greater than 50% AUROC) at $\epsilon = \frac{8}{255}$.

Recently, several approaches have sought to address this challenge by first demonstrating that relying solely on ID data is insufficient for building adversarially robust detectors (27; 28; 29; 23; 30; 26; 31; 32; 33; 34; 35; 36; 37; 38). Consequently, new methods propose incorporating copious amounts of auxiliary OOD data in conjunction with adversarial training to improve the detector's robustness. While effective, a significant gap remains between detector performance on clean data and their robustness against adversarial attacks (see Figure 1, Tables 1, 2a, and 2b).

This performance gap primarily arises from the wide variety of potential OOD samples encountered during testing. Relying exclusively on an auxiliary dataset to generate perturbed OOD data can bias the model toward specific OOD instances, thereby compromising the detector's ability to generalize to unseen OOD data during inference (16; 39; 40; 41; 42; 43). This limitation is particularly pronounced in adversarial settings, where adversarial training demands a higher level of data complexity compared to standard training (44; 45; 46; 5; 47). Additionally, the collection of auxiliary OOD data is a costly process, as it must be carefully curated to avoid overlap with ID semantics to ensure that the detector is not confused by data ambiguities (39; 41). Finally, as our empirical analysis reveals, existing OOD detection methods are vulnerable even to non-adversarial perturbations – a concerning issue for open-world applications, where natural factors such as lighting conditions or sensor noise can introduce significant variability (48) (see Table 3).

Our Contribution: To address these challenges, we propose AROS (**A**dversarially **R**obust **O**OD Detection through **S**tability), a novel approach that leverages NODEs with the Lyapunov stability theorem (Figure 2). This constraint asserts that small perturbations near stable equilibrium points decay over time, allowing the system state to converge back to equilibrium. By ensuring that both ID and OOD data are stable equilibrium points of the detector, the system's dynamics mitigate the effects of perturbations by guiding the state back to its equilibrium. Instead of using extra OOD image data, we craft fake OOD samples in the embedding space by estimating the ID boundary. Additionally, we show that adding an orthogonal binary layer increases the separation between ID and OOD equilibrium points, enhancing robustness. We evaluate AROS under both adversarial and clean setups across various datasets, including large-scale datasets such as ImageNet (49) and real-world medical imaging data (i.e., ADNI (22)), and compare it to previous state-of-the-art methods. Under adversarial scenarios, we apply strong attacks, including PGD^{1000} (44), AutoAttack (50), and Adaptive AutoAttack (51).

2 Preliminaries

Out-of-Distribution Detection. In an OOD detection setup, it is assumed that there are two sets: an ID dataset and an OOD dataset. We denote the ID dataset as $\mathcal{D}^{\text{in}}$, which consists of pairs $(\mathbf{x}^{\text{in}}, y^{\text{in}})$, where $\mathbf{x}^{\text{in}}$ represents the ID data, and $y^{\text{in}} \in \mathcal{Y}^{\text{in}} := \{1, \ldots, K\}$ denotes the class label. Let $\mathcal{D}^{\text{out}}$ represent the OOD dataset, containing pairs $(\mathbf{x}^{\text{out}}, y^{\text{out}})$, where $y^{\text{out}} \in \mathcal{Y}^{\text{out}} :=$

$\{K+1, \ldots, K+O\}$, and $\mathcal{Y}^{\text{out}} \cap \mathcal{Y}^{\text{in}} = \emptyset$ (52; 18). In practice, different datasets are often used for $\mathcal{D}^{\text{in}}$ and $\mathcal{D}^{\text{out}}$. Alternatively, another scenario is called open-set recognition, where a subset of classes within a dataset is considered as ID, while the remaining classes are considered as OOD (53; 16; 54; 55). A trained model $\mathcal{F}$ assigns an OOD score $S_{\mathcal{F}}$ to each test input, with higher scores indicating a greater likelihood of being OOD.

Adversarial Attack on OOD Detectors. Adversarial attacks involve perturbing an input sample x to generate an adversarial example x^* that maximizes the loss function $\ell(x^*; y)$. The perturbation magnitude is constrained by ϵ to ensure that the alteration remains imperceptible. Formally, the adversarial example is defined as $x^* = \arg\max_{x'} \ell(x'; y)$, subject to $\|x - x^*\|_p \leq \epsilon$, where p denotes the norm (e.g., $p = 2, \infty$) (56; 3; 44). A widely used attack method is Projected Gradient Descent (PGD) (44), which iteratively maximizes the loss by following the gradient sign of $\ell(x^*; y)$ with a step size α. For adversarial evaluation (28; 34; 37), we adapt this approach by targeting the OOD score $S_{\mathcal{F}}(x)$. Specifically, the adversarial attack aims to mislead the detector by increasing the OOD score for ID samples and decreasing it for OOD samples, causing misprediction: $x_0^* = x, \quad x_{t+1}^* = x_t^* + \alpha \cdot \text{sign}\left(\mathbb{I}(y) \cdot \nabla_x S_{\mathcal{F}}(x_t^*)\right), \quad x^* = x_n^*$, where n is the number of steps, and $\mathbb{I}(y) = +1$ if $y \in \mathcal{Y}^{\text{in}}$ and -1 if $y \in \mathcal{Y}^{\text{out}}$. This approach is consistently applied across all attacks considered in our study.

Neural ODE and Stability. In the NODE framework, the input and output are treated as two distinct states of a continuous dynamical system, whose evolution is described by trainable layers parameterized by weights ϕ and denoted as h_ϕ. The state of the neural ODE, represented by Z, evolves over time according to these dynamics, establishing a continuous mapping between the input and output (57; 58; 59). The relationship between the input and output states is governed by the following differential equations: $\frac{dz(t)}{dt} = h_\phi(z(t), t), \quad z(0) = z_{\text{input}}, \quad z(T) = z_{\text{output}}$.

3 Related Work

OOD Detection Methods. Existing OOD detection methods can be broadly categorized into post-hoc and training-based approaches. Post-hoc methods involve training a classifier on ID data and subsequently using statistics from the classifier's outputs or intermediate representations to identify OOD samples. For instance, Hendrycks et al. (52) propose using the maximum softmax probability distributions (MSP) as a metric. The MD method (60) leverages the Mahalanobis Distance in the feature space, and OpenMax (61) recalibrates classification probabilities to improve OOD detection. Training-based methods, modify the training process to enhance OOD detection capabilities. Such modifications can include defining additional loss functions, employing data augmentation techniques, or incorporating auxiliary networks. Examples of training-based methods designed for standard OOD detection include VOS (39), DHM (19), CATEX (62), and CSI (63). On the other hand, ATOM (30), ALOE (28), ATD (34), and RODEO (37) have been developed specifically for robust detection. For detailed descriptions of these methods, please refer to Appendix A1.

Stable NODE for Robustness. TiSODE (64) introduces a time-invariant steady NODE to constrain trajectory evolution by keeping the integrand close to zero. Recent works employ Lyapunov stability theory to develop provable safety certificates for neural network systems, particularly in classification tasks. PeerNets (9) was among the first to use control theory and dynamical systems to improve robustness. Kolter et al. (65) designed a Lyapunov function using neural network architectures to stabilize a base dynamics model's equilibrium. ASODE (66) uses non-autonomous NODEs with Lyapunov stability constraints to mitigate adversarial perturbations in slowly time-varying systems. LyaDEQ (67) introduces a new module based on ICNN (68) into its pipeline, leveraging deep equilibrium models and learning a Lyapunov function to enhance stability. SODEF (69) enhances robustness against adversarial attacks by applying regularizers to stabilize the behavior of NODE under the time-invariant assumption. In Table 4a, we analyze these stability-based classifiers as OOD detectors and highlight the potential of Lyapunov's theorem as a framework for robust OOD detection, and show our method's ability to improve performance over these excellent baselines.

4 Proposed Method

Motivation. A robust detector should be resistant to shifting ID test samples to OOD, and vice versa, under adversarial attack. A common approach for developing robust OOD detectors involves

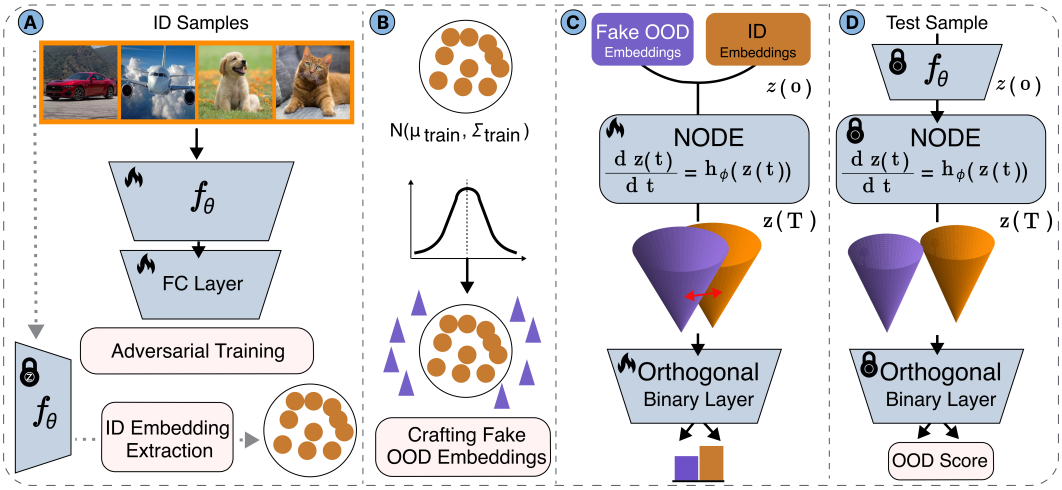

Figure 2: **An illustration of AROS**. **(A)** To obtain robust initial features for OOD detection, we perform adversarial training on a classifier using only ID samples. **(B)** We estimate the ID distribution within the embedding space and generate fake OOD embeddings as a proxy for real OOD data. This enables the creation of two balanced classes of samples: ID and fake OOD. **(C)** The model incorporates a NODE layer h_ϕ and an Orthogonal Binary Layer B_η. Using these two classes, we train the pipeline with the loss function $\mathcal{L}_{\text{SL}}$ to stabilize the system dynamics. **(D)** During inference, an input passes through the feature extractor f_θ, NODE h_ϕ, and Orthogonal Binary Layer B_η, and the resulting likelihood from B_η serves as the OOD score. The complete algorithmic workflow of AROS can be found in Appendix A2.

employing adversarial training on ID data, combined with an auxiliary real OOD dataset, to expose the detector to potential vulnerable perturbations. The core intuition is that adversarial training on ID data alone, without an accompanying OOD dataset, leaves the detector susceptible to perturbations that alter the boundary between ID and OOD data during testing (28; 29; 30; 23; 34; 26; 31; 33; 36; 35; 37). Beyond the unsatisfactory performance of the prior approach, there are further challenges with this strategy. A key issue is the cost of preparing an auxiliary dataset disjoint from the ID data, along with ensuring that the selected OOD images adequately cover the boundary between ID and OOD samples—a critical factor for such frameworks (70; 37; 30; 24). Moreover, adversarial training of neural networks is notably more data-intensive than standard setups, further increasing complexity (45; 46; 5; 47). There is also the concern that exclusively relying on perturbed OOD data may introduce biases toward specific OOD examples (39; 41). To address these challenges, we propose AROS, which utilizes provable stability theorems in the embedding space to develop a robust OOD detector without requiring exposure to perturbed OOD image data.

Overview of AROS. AROS ensures that perturbed input samples remain close to their non-perturbed counterparts in the feature space by leveraging the Lyapunov stability theorem (71; 72; 73; 74). By using a NODE, we consider the model as a dynamical system and design it so that ID and OOD samples converge to distinct stable equilibrium points of that system. This approach prevents significant deviations in the output when adversarial perturbations are applied. However, since OOD data is unavailable, we craft fake OOD samples in the embedding space by estimating the boundaries of the ID distribution and sampling from the corresponding low-likelihood regimes. To further avoid any misprediction between OOD and ID data caused by perturbations, we maximize the distance between their equilibrium points by leveraging an orthogonal binary layer for classification. In the following, we will thoroughly explain each proposed component, highlighting the benefits of AROS.

4.1 Fake Embedding Crafting Strategy

There have been efforts to utilize synthetic features, primarily under clean scenarios (39; 41; 75; 70). However, for adversarial settings, prior work has often relied on a large pre-trained model and additional data. In contrast, our approach limits information to ID samples, proposing to craft OOD

data from ID data in the embedding space. These generated OOD samples are subsequently utilized in the training step.

We employ a well-trained encoder to transform ID training data into robust embedding spaces. To achieve these embeddings, we first adversarially train a classifier on ID training samples using cross-entropy loss $\mathcal{L}_{\text{CE}}$ and the PGD$^{10}(l_\infty)$ attack. By removing the last fully connected layer from the classifier, we utilize the remaining encoder, denoted as f_θ, to extract ID embeddings r, where $r = f_\theta(x)$ from an ID training sample x (Figure 2A). Specifically, by considering $\mathcal{D}^{\text{in}}_{\text{train}}$ with K classes, we estimate their distribution as a K class-conditional Gaussian distribution, a well-known approach in the detection literature (39; 76; 77; 78; 79; 80).

We then select fake embeddings r from the feature space corresponding to class j such that $r \sim \mathcal{N}(\hat{\mu}_j, \hat{\Sigma}_j)$ satisfies:

$$\frac{1}{(2\pi)^{d/2}|\hat{\Sigma}_j|^{1/2}} \exp\left(-\frac{1}{2}(r-\hat{\mu}_j)^T \hat{\Sigma}_j^{-1}(r-\hat{\mu}_j)\right) \quad < \quad \beta, \tag{1}$$

where, β serves as a threshold, and we set that to a very small value (e.g., 0.001) (Figure 2B). Additionally, we conduct an ablation study to evaluate the impact of different values of β and discuss practical considerations (see Appendix A3). Here, d is the dimensionality of the feature vectors r, and $j = 1, \ldots, K$. The terms $\hat{\mu}_j$ and $\hat{\Sigma}_j$ represent the mean vector and covariance matrix of the j-th class of ID training samples in feature space, respectively:

$$\hat{\mu}_j = \frac{1}{n_j}\sum_{i:y_i=j} f_\theta(x_i), \quad \hat{\Sigma}_j = \frac{1}{n_j-1}\sum_{i:y_i=j}(f_\theta(x_i)-\hat{\mu}_j)(f_\theta(x_i)-\hat{\mu}_j)^T, \tag{2}$$

where n_j is the number of samples in class j. By sampling equally across each class of $\mathcal{D}^{\text{in}}_{\text{train}}$, we generate a set of synthetic, "fake" OOD embeddings (Figure 2C), denoted as r_{OOD}. We then construct a balanced training set by taking the union of the embeddings of ID samples and the OOD embeddings, defining it as: $X_{\text{train}} = \left\{f_\theta(\mathcal{D}^{\text{in}}_{\text{train}}) \cup r_{\text{OOD}}\right\}$. We define the labels y for this set as 0 for ID and 1 for fake OOD embeddings.

4.2 Lyapunov Stability for Robust OOD Detection

As mentioned, several approaches have been proposed to apply Lyapunov's theorem to deep networks in practice, including methods such as LyaDEQ (67), ASODE (66), and SODEF (69). Here, we utilize their framework to define the objective function and also benchmark our approach to these baselines. Amongst them, SODEF adopts a time-invariant (69; 64) assumption, which makes stability analysis more practical, as the behavior of the neural ODE depends solely on the state $z(t)$, independent of the specific time that the state is reached. This assumption implies that the equilibrium points of the NODE remain constant over time, facilitating a more tractable analysis of how perturbations evolve around these points (64; 81; 69). This is supported by our experiments in Table 4a, which highlight SODEF's superior robustness. Consequently, we adopt the time-invariant framework and use their approach to define the loss function. In order to gain intuition for our approach, we provide the basic mathematical overview of how we leverage the Lyapunov theorems. In this study, as a practical consideration, we assume that the networks utilized have continuous first derivatives with respect to the input $z(0)$, which has been shown to be a reasonable assumption (82).

For a given dynamic system $\frac{dz(t)}{dt} = h_\phi(z(t))$, a state $z^\star$ is an equilibrium point of system if $z^\star$ satisfies $h(z^\star) = 0$. An equilibrium point is stable if the trajectories starting near $z^\star$ remain around it all the time. More formally:

Definition 1: (Lyapunov stability (83)). An equilibrium $z^\star$ is said to be stable in the sense of Lyapunov if, for every $\varepsilon > 0$, there exists $\delta > 0$ such that, if $\|z(0) - z^\star\| < \delta$, then $\|z(t) - z^\star\| < \varepsilon$ for all $t \geq 0$. If $z^\star$ is stable, and $\lim_{t\to\infty} \|z(t) - z^\star\| = 0$, $z^\star$ is said to be asymptotically stable.

Theorem 1: (Hartman–Grobman Theorem (84)). *Consider a time-invariant system with continuous first derivatives, represented by $\frac{d\mathbf{z}(t)}{dt} = h(\mathbf{z}(t))$. For a fixed point $\mathbf{z}^*$, if the Jacobian matrix ∇h evaluated at $\mathbf{z}^*$ has no eigenvalues with a real part equal to zero, the behavior of the original nonlinear dynamical system can be analyzed by studying the linearization of the system around this fixed point. The linearized system is given by $\frac{d\mathbf{z}'(t)}{dt} = \mathbf{A}\mathbf{z}'(t)$, where $\mathbf{A}$ is the Jacobian matrix evaluated at $\mathbf{z}^*$. This allows for a simplified analysis of the local dynamics in the vicinity of $\mathbf{z}^*$.*

Theorem 2: (Lyapunov Stability Theorem (83)) *The equation* $\frac{d\mathbf{z}'(t)}{dt} = \mathbf{A}\mathbf{z}'(t)$, *is asymptotically stable if and only if all eigenvalues of* $\mathbf{A}$ *have negative real parts.*

Theorem 3: (Levy–Desplanques Theorem (85)) *Let* $A = [a_{ij}]$ *be an* n*-dimensional square matrix and suppose it is strictly diagonally dominant, i.e.,* $|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|$ *and* $a_{ii} \leq 0$ *for all* i*. Then every eigenvalue of* A *has a negative real part.*

Definition 1 introduces the concept of asymptotic stability. Building on this, *Theorem 1* demonstrates that the behavior of a nonlinear, time-invariant system near a fixed point can be effectively analyzed through its linearization. *Theorem 2* then establishes a key condition for the asymptotic stability of linear systems: all eigenvalues of the system matrix must have negative real parts. To facilitate the verification of this stability condition, *Theorem 3* provides a practical criterion based on the matrix's eigenvalues. In the subsequent section, we will introduce an objective function designed to adhere to these stability criteria.

4.3 Orthogonal Binary Layer and Training Step

We propose incorporating an orthogonal binary layer (86) denoted as B_η after the NODE h_ϕ in our pipeline to maximize the distance between the equilibrium points of ID and OOD data. Intuitively, this layer prevents the misalignment of convergence between perturbed OOD data and ID data by maximizing the distance between their equilibrium points. Given the output z from the h_ϕ, the orthogonal binary layer B_η applies a transformation using weights w such that $w^T w = I$, ensuring orthogonality. Although Lyapunov stability encourages perturbed inputs to converge to neighborhoods of their unperturbed counterparts, the infinite-depth nature of NODE (87) makes them susceptible to degraded activations due to exploding or vanishing gradients (88). The introduction of an orthogonal layer mitigates this risk. Moreover, encouraging orthogonality within neural networks has demonstrated multiple benefits, such as preserving gradient norms and enforcing low Lipschitz constants—both of which contribute to enhanced robustness (89; 90; 91).

To satisfy the aforementioned conditions, we optimize the following empirical Lagrangian $\mathcal{L}_{\text{SL}}$ with training data (X_{train}, y):

$$\mathcal{L}_{\text{SL}} = \min_{\phi,\eta} \frac{1}{|X_{\text{train}}|} \Bigg(\ell_{\text{CE}}(B_\eta(h_\phi(X_{\text{train}})), y) + \gamma_1 \|h_\phi(X_{\text{train}})\|_2 + \gamma_2 \text{exp}\left(-\sum_{i=1}^{n} [\nabla h_\phi(X_{\text{train}})]_{ii} \right) + \gamma_3 \text{exp}\left(\sum_{i=1}^{n} \left(-|[\nabla h_\phi(X_{\text{train}})]_{ii}| + \sum_{j \neq i} |[\nabla h_\phi(X_{\text{train}})]_{ij}| \right) \right) \Bigg) \tag{3}$$

Note that here, X_{train} serves as the initial hidden state, i.e., $z(0)$, for the NODE layer. The first term, ℓ_{CE}, is a cross-entropy loss function. The second term forces $z(0)$ to be near the equilibrium points, while the remaining terms ensure strictly diagonally dominant derivatives, as described in Theorem 3. The exp$(.)$ function is selected as a monotonically increasing function with a minimum bound to limit the unbounded influence of the two regularizers, preventing them from dominating the loss. We choose the hyperparameters as $\gamma_1 = 1$ and $\gamma_2 = \gamma_3 = 0.05$, and we discuss the ablation study on these hyperparameters, as well as other training step details, in Appendix A4. By optimizing this objective function, the model learns Lyapunov-stable representations where ID and OOD equilibrium points are well-separated in the feature space after the NODE. The B_η captures the probability distribution over the binary classes (ID vs. fake OOD), and for the OOD score of an input x, we use its probability assigned to the OOD class (Figure 2D).

5 Experiments

Here we present empirical evidence to validate the effectiveness of our method under various setups, including adversarial attacks, corrupted inputs (non-adversarial perturbations), and clean inputs (non-perturbed scenarios).

First, we adversarially train a classifier on ID data and then use it to map the data into a robust embedding space. A Gaussian distribution is fitted around these embeddings, and low-likelihood regions of the distribution are sampled to create fake OOD data as a proxy for OOD test samples. We

Table 1. Performance of OOD detection methods under clean evaluation, random corruption (Gaussian noise), and PGD (l_∞) adversarial attack with 1000 steps and $\frac{8}{255}$, as well as AutoAttack and Adaptive AutoAttack (AA), measured by AUROC (%). A clean evaluation is one where no attack is made on the data. For corruption evaluation, Gaussian noise from the ImageNet-C (48) benchmark was used. The best results are highlighted in **bold**, and the second-best results are underlined in each row.

† These methods leveraged auxiliary datasets and these * used large pretrained models as part of their pipeline.

Dataset		Attack	Method								
$\mathcal{D}_{in}$	$\mathcal{D}_{out}$		**VOS** (ResNet)	**DHM** (WideResNet)	**CATEX*** (CLIP)	**CSI** (ResNet)	**ATOM†** (DenseNet)	**ALOE†** (WideResNet)	**ATD†*** (WideResNet)	**RODEO†*** (CLIP)	**AROS** (WideResNet)
CIFAR10	**CIFAR100**	**Clean**	87.9	**100.0**	88.3	92.2	94.2	78.8	82.0	75.6	88.2
		Corruption	56.2	57.7	60.4	54.7	57.3	54.5	59.2	58.6	**84.3**
		PGD^{1000}	4.2	1.8	0.8	3.6	1.6	16.1	37.1	37.8	**80.1**
		AutoAttack	0.0	1.2	0.0	0.4	0.5	14.8	36.2	35.9	**78.9**
		AdaptiveAA	0.0	0.0	1.7	0.0	0.0	11.5	34.8	32.3	**76.4**
CIFAR100	**CIFAR10**	**Clean**	71.3	**100.0**	85.1	53.2	87.5	43.6	57.5	61.5	74.3
		Corruption	53.8	58.2	57.4	50.1	55.3	56.1	56.0	54.9	**71.8**
		PGD^{1000}	5.4	0.0	4.0	2.8	2.0	1.3	12.1	29.0	**67.0**
		AutoAttack	2.6	0.0	0.3	0.9	0.0	0.0	10.5	28.3	**66.5**
		AdaptiveAA	0.0	1.4	0.0	0.0	0.0	0.2	9.4	26.7	**65.2**

then demonstrate that time invariance, which establishes that the NODE's behavior does not explicitly depend on time, leads to more stable behavior of the detector under adversarial attacks (see Section 5). Consequently, we leverage Lyapunov stability regularization under a time-invariant assumption for training. However, a potential challenge arises when ID and OOD equilibrium points are located near each other. As a remedy, we introduce an orthogonal binary layer (86; 92; 93) that increases the separation between ID and OOD equilibrium points, thereby expanding the stability region around each point. This enhances the model's robustness against shifting adversarial samples from OOD to ID and vice versa. Finally, we use the orthogonal binary layer's confidence output as the OOD score during inference.

Experimental Setup & Datasets. We evaluated OOD detection methods under both adversarial and clean scenarios (see Tables 1 and 2a). Each experiment utilized two disjoint datasets: one as the ID dataset and the other as the OOD test set. For Table 1, CIFAR-10 or CIFAR-100 (94) served as the ID datasets. Table 2a extends the evaluation to ImageNet-1k as the ID dataset, with OOD datasets being comprised of Texture (95), SVHN (96), iNaturalist (97), Places365 (98), LSUN (99), and iSUN (100) – all sourced from disjoint categories (48).

An OSR (101) setup was also tested, in which each experiment involved a single dataset that was randomly split into ID (60%) and OOD (40%) subclasses, with results averaged over 10 trials. Datasets used for OSR included CIFAR-10, CIFAR-100, ImageNet-1k, MNIST (102), FMNIST (103), and Imagenette (104) (Table 2b). Additionally, models were evaluated on corrupted data using the CIFAR-10-C and CIFAR-100-C benchmarks (48) (see Table 3). Specifically, both the ID and OOD data were perturbed with corruptions that did not alter semantics but introduced slight distributional shifts during testing. Further details on the datasets are provided in Appendix A5.

Evaluation Details. For adversarial evaluation, all ID and OOD test data were perturbed by using a fully end-to-end PGD (l_∞) attack targeting their OOD scores (as described in Section 2). We used $\epsilon = \frac{8}{255}$ for low-resolution images and $\epsilon = \frac{4}{255}$ for high-resolution images. The PGD attack steps denoted as M were set to 1000, with 10 random initializations sampled from the interval $(-\epsilon, \epsilon)$. The step size for the attack was set to $\alpha = 2.5 \times \frac{\epsilon}{M}$ (4). Additionally, we considered AutoAttack and Adaptive AutoAttack (Table 1). Details on how these attacks are tailored for the detection task can be found in Appendix A4. As the primary evaluation metric, we used AUROC, representing the area under the receiver operating characteristic curve. Additionally, we used AUPR and FPR95 as supplementary metrics, with results presented in Table 4b. AUPR represents the area under the precision-recall curve, while FPR95 measures the false positive rate when the model correctly identifies 95% of the true positives.

Table 2a. Performance of OOD detection methods under clean evaluation and $\text{PGD}^{1000}(l_\infty)$ measured by AUROC (%). The perturbation budget ϵ is set to $\frac{8}{255}$ for low-resolution datasets and $\frac{4}{255}$ for high-resolution datasets. The table cells denote results in the 'Clean/PGD^{1000}' format.

Dataset		Method								
$\mathcal{D}_{in}$	$\mathcal{D}_{out}$	VOS	DHM	CATEX	CSI	ATOM	ALOE	ATD	RODEO	AROS (Ours)
CIFAR-10	CIFAR-100	87.9/4.2	**100.0**/1.8	88.3/0.8	92.2/3.6	94.2/1.6	78.8/16.1	82.0/37.1	75.6/37.8	88.2/**80.1**
	SVHN	93.3/2.8	**100.0**/ 4.5	91.6/2.3	97.4/1.7	89.2/4.7	83.5/26.6	87.9/39.0	83.0/38.2	93.0/**86.4**
	Places	89.7/5.2	**99.6**/0.0	90.4/ 4.7	93.6/0.1	98.7/5.6	85.1/21.9	92.5/59.8	96.2/70.2	90.8/**83.5**
	LSUN	98.0/7.3	**100.0**/2.6	95.1/0.8	97.7/0.0	99.1/1.0	98.7/50.7	96.0/68.1	99.0/ **85.1**	90.6/ 82.4
	iSUN	94.6/0.5	99.1/2.8	93.2/ 4.4	95.4/3.6	**99.5**/2.5	98.3/49.5	94.8/65.9	97.7/78.7	88.9/**81.2**
CIFAR-100	CIFAR-10	71.3/5.4	**100.0**/2.6	85.1/ 4.0	53.2/0.7	87.5/2.0	43.6/1.3	57.5/12.1	61.5/29.0	74.3/**67.0**
	SVHN	92.6/3.2	**100.0**/0.8	94.6/5.7	90.5/4.2	92.8/5.3	74.0/18.1	72.5/27.6	76.9/31.4	81.5/**70.6**
	Places	75.5/0.0	**100.0**/3.9	87.3/1.4	73.6/0.0	94.8/3.0	75.0/12.4	83.3/40.0	93.0/66.6	77.0/**69.2**
	LSUN	92.9/5.7	**100.0**/1.6	94.0/8.9	63.4/1.8	96.6/1.5	98.7/50.7	96.0/68.1	98.1/63.1	74.3/**68.1**
	iSUN	70.2/4.5	99.6/3.6	81.2/0.0	81.4/3.0	96.4/1.4	**98.3**/49.5	94.8/65.9	95.1/65.6	72.8/**67.9**
ImageNet-1k	Texture	86.7/0.8	82.4/0.0	92.7/0.0	85.8/0.6	**88.9**/7.3	76.2/21.8	74.2/15.7	71.3/19.4	78.3/**69.2**
	iNaturalist	94.5/0.0	80.7/0.0	97.9/2.0	85.2/1.7	83.6/10.5	78.9/19.4	72.5/12.6	72.7/15.0	84.6/**75.3**
	Places	90.2/0.0	76.2/0.4	**90.5**/0.0	83.9/0.2	84.5/12.8	78.6/15.3	75.4/17.5	69.2/18.5	76.2/**68.1**
	LSUN	91.9/0.0	82.5/0.0	**92.9**/0.4	78.4/1.9	85.3/11.2	77.4/ 16.9	68.3/15.1	70.4/16.2	79.4/**69.0**
	iSUN	92.8/2.7	81.6/0.0	**93.7**/0.0	77.5/0.0	80.3/14.1	75.3/11.8	76.6/15.8	72.8/17.3	80.3/**71.6**
	Mean	88.1/2.8	**93.4**/1.6	91.2/2.3	83.3/1.5	91.4/5.6	81.4/25.5	81.6/37.4	82.1/44.4	82.0/**74.0**

Table 2b. Performance (Clean/PGD^{1000}) of OOD detection methods under clean and $\text{PGD}^{1000}(l_\infty)$, measured by AUROC (%), on the OSR setup, which splits one dataset's classes randomly to create $\mathcal{D}_{in}$ and $\mathcal{D}_{out}$.

Dataset	Method								
	VOS	DHM	CATEX	CSI	ATOM	ALOE	ATD	RODEO	AROS (Ours)
MNIST	86.3/4.8	92.6/0.4	92.3/1.9	93.6/6.1	74.8/4.1	79.5/37.3	68.7/56.5	**97.2**/85.0	94.4/**86.3**
FMNIST	78.1/2.0	85.9/0.0	87.0/0.4	84.6/1.2	64.3/4.2	72.6/28.5	59.6/42.1	**87.7**/65.3	84.1/**72.6**
CIFAR-10	74.7/0.0	90.8/0.0	**95.1**/0.0	91.4/0.6	68.3/5.0	52.4/25.6	49.0/32.4	79.6/62.7	78.8/**69.5**
CIFAR-100	63.5/0.0	78.6/0.0	**91.9**/0.0	86.7/1.9	51.4/2.6	49.8/18.2	50.5/36.1	64.1/35.3	67.0/**58.2**
Imagenette	76.7/0.0	84.2/0.0	**96.4**/1.6	92.8/0.0	63.5/8.2	61.7/14.2	63.8/28.4	70.6/39.4	78.2/**67.5**
ADNI	73.5/4.1	69.4/5.2	**86.9**/0.1	82.1/0.0	66.9/2.3	64.0/11.0	68.3/33.9	75.5/24.6	80.9/**61.7**
Mean	75.5/1.8	83.6/0.9	**91.6**/0.7	88.5/1.6	64.9/4.4	63.3/22.5	60.0/38.2	79.1/52.1	80.6/**69.3**

Reported and Re-Evaluated Results. Some methods may show different results here compared to those reported in their original papers (30; 28) due to our use of stronger attacks we incorporated for evaluation, or the more challenging benchmarks used. For example, ALOE (28) considered a lower perturbation budget for evaluation (i.e., $\frac{1}{255}$), and the ATD (34) and RODEO (37) benchmarks used CIFAR-10 vs. a union of several datasets, rather than CIFAR-10 vs. CIFAR-100. The union set included datasets such as MNIST, which is significantly different from CIFAR-10, leading to a higher reported robust performance.

Results Analysis. Without relying on additional datasets or pretrained models, AROS significantly outperforms existing methods in adversarial settings, achieving up to a 40% improvement in AUROC and demonstrating competitive results under clean setups (see Table 2b). Specifically, AROS also exhibits greater robustness under various corruptions, further underscoring its effectiveness in OOD detection. We further verify our approach through an extensive ablation study of various components in AROS (see Section 6).

Table 3. Performance of OOD detection methods under various types of non-adversarial perturbations, referred to as image corruptions, as introduced in the CIFAR-10-C and CIFAR-100-C datasets (48), measured by AUROC (%). Specifically, test inputs, including both ID and OOD, are perturbed with a particular corruption in each experiment.

Dataset $\mathcal{D}_{in}$	Dataset $\mathcal{D}_{out}$	Methods	Corruption: Gauss.	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	*Mean*
CIFAR-10-C	CIFAR-100-C	**VOS**	56.2	67.5	76.5	77.7	73.9	78.7	76.3	72.0	54.1	77.0	58.5	79.1	**81.2**	**83.6**	74.4	72.5
		DHM	57.7	78.7	72.4	75.4	75.6	73.9	77.5	75.8	70.8	56.8	74.5	58.0	77.4	78.4	80.6	72.2
		CATEX	62.4	**80.9**	73.0	78.4	76.3	78.6	**81.3**	79.9	78.9	58.3	80.0	54.0	79.0	80.5	82.4	74.9
		CSI	54.7	58.0	58.7	62.9	61.7	69.0	65.9	77.2	69.2	74.8	**91.9**	65.8	74.2	62.6	74.9	68.1
		ATOM	57.3	75.5	63.6	70.7	72.2	69.9	74.6	77.2	76.5	55.3	80.5	54.1	74.7	77.4	80.8	70.7
		ALOE	54.5	76.4	64.0	71.5	73.0	70.9	75.5	78.2	77.9	56.3	81.5	54.0	76.9	79.3	82.1	71.5
		ATD	59.2	79.2	71.0	76.7	76.9	75.6	79.5	78.2	74.9	59.5	77.8	59.5	79.0	80.8	**82.9**	73.7
		RODEO	58.6	76.0	68.5	73.5	73.8	72.1	75.5	74.5	70.9	57.8	74.5	57.7	75.3	76.8	79.5	71.0
		AROS	**84.3**	76.5	**79.2**	**83.8**	**77.3**	**82.0**	**81.3**	**83.4**	**84.0**	**84.0**	84.7	**83.3**	80.7	79.6	82.5	**81.8**
CIFAR-100-C	CIFAR-10-C	**VOS**	53.8	55.7	65.6	58.2	47.1	51.4	57.6	53.9	59.0	57.2	56.5	54.8	48.2	59.4	51.1	55.3
		DHM	58.2	59.9	64.0	57.7	48.9	58.0	57.4	57.6	58.5	57.9	58.1	58.3	49.8	55.6	56.7	57.1
		CATEX	57.4	60.2	65.7	**59.6**	64.9	62.9	59.3	67.5	61.4	**59.8**	60.0	**64.2**	56.8	57.5	**58.6**	61.0
		CSI	50.1	48.8	50.6	47.8	47.5	46.9	46.8	50.6	50.3	51.8	49.9	52.2	42.9	48.0	47.7	48.8
		ATOM	55.3	51.2	53.1	50.2	49.9	49.2	49.6	53.1	52.8	54.4	52.4	54.8	45.0	50.4	50.8	51.5
		ALOE	56.1	53.4	62.8	54.5	51.8	54.9	54.1	54.4	55.6	54.8	52.7	56.4	47.8	51.7	53.2	54.3
		ATD	56.0	57.4	61.5	57.5	44.8	57.1	54.2	56.9	58.3	55.2	53.7	57.5	49.3	50.8	56.0	55.1
		RODEO	54.9	58.1	60.6	56.4	51.0	60.5	58.9	58.4	57.9	54.6	57.4	52.3	52.7	53.5	51.2	55.9
		AROS	**71.8**	**74.8**	**67.7**	**59.6**	**72.6**	**73.9**	**65.7**	**68.5**	**64.4**	**59.8**	**75.0**	**64.2**	**72.8**	**69.5**	**58.6**	**67.9**

Table 4a. Comparison of post-hoc OOD detection methods using different classifiers trained with various strategies and evaluated with multiple scoring functions. The comparison (Clean/PGD^{1000}) is conducted under clean and PGD^{1000} conditions, measured by AUROC (%).

Classifier	Posthoc Method	CIFAR-10: CIFAR-100	CIFAR-10: SVHN	CIFAR-100: CIFAR-10	CIFAR-100: SVHN
Standard	MSP	87.9/0.0	91.8/1.4	75.4/0.2	71.4/3.6
	MD	**88.5**/4.3	**99.1**/0.6	75.0/1.9	**98.4**/0.6
	OpenMax	86.4/0.0	94.7/2.8	77.6/0.0	93.9/4.2
AT	MSP	79.3/16.0	85.1/19.7	67.2/10.7	74.6/11.3
	MD	81.4/25.6	88.2/27.5	71.8/15.0	81.5/19.7
	OpenMax	82.4/27.8	86.5/26.9	**80.0**/16.4	75.4/22.9
ODENet	MSP	84.2/10.6	89.3/15.4	69.7/12.5	76.1/23.8
	MD	80.7/9.1	84.6/13.0	66.4/14.8	72.9/16.4
	OpenMax	83.8/14.2	87.4/20.9	70.3/15.6	75.6/18.2
LyaDEQ	MSP	77.5/56.5	83.7/58.5	69.1/48.0	69.4/53.3
	MD	79.1/56.9	82.0/56.5	60.3/ 53.4	69.3/54.2
	OpenMax	76.0/47.4	77.5/56.5	67.8/57.1	73.3/ 58.0
ASODE	MSP	76.3/56.3	80.5/62.5	64.6/44.9	64.6/58.9
	MD	74.9/49.5	76.1/54.4	59.3/ 52.0	72.1/55.1
	OpenMax	72.6/44.2	75.9/57.9	66.1/52.1	80.5/ 50.4
SODEF	MSP	83.5/61.9	86.4/65.3	67.2/53.1	73.7/60.4
	MD	75.4/57.7	81.9/64.2	65.8/58.4	71.8/62.5
	OpenMax	82.8/65.3	86.4/69.1	66.3/56.6	75.2/64.9
AROS	N/A	88.2/**80.1**	93.0/**86.4**	74.3/**67.0**	82.5/**70.6**

Table 4b. Performance of OOD detection methods under clean and PGD^{1000}, measured by AUPR↑ (%) and FPR95 ↓(%) metrics. The perturbation budget ϵ is set to $\frac{8}{255}$. The table cells present results in the 'Clean/PGD^{1000}' format.

Method	Metric	CIFAR-10: CIFAR-100	CIFAR-10: SVHN	CIFAR-100: CIFAR-10	CIFAR-100: SVHN
VOS	AUPR↑	85.8/0.0	90.4/6.2	75.8/0.0	93.9/7.6
	FPR95↓	35.2/100.0	38.2/99.8	48.7/100.0	41.5/98.2
DHM	AUPR↑	**100.0**/0.3	**100.0**/4.8	**100.0**/0.0	**100.0**/3.2
	FPR95↓	**0.2**/99.2	**0.0**/98.5	**1.1**/100.0	**0.4**/99.7
CATEX	AUPR↑	89.5/0.4	93.1/7.6	84.2/0.0	96.6/1.3
	FPR95↓	36.6/99.1	27.3/95.6	42.8/100.0	37.1/98.4
CSI	AUPR↑	93.4/0.0	98.2/4.6	65.8/0.0	82.9/0.4
	FPR95↓	40.6/100.0	37.4/99.1	65.2/100.0	42.6/97.5
ATOM	AUPR↑	97.9/5.8	98.3/11.6	89.3/5.1	94.6/7.2
	FPR95↓	24.0/96.4	12.7/93.1	38.6/98.0	29.2/97.9
ALOE	AUPR↑	80.4/21.7	86.5/27.3	54.8/9.2	85.1/18.6
	FPR95↓	38.6/89.2	45.1/93.7	72.8/96.1	57.4/84.8
ATD	AUPR↑	81.9/44.6	85.3/53.7	61.4/27.2	68.3/26.1
	FPR95↓	47.3/86.2	42.4/83.0	68.2/94.8	59.0/91.9
RODEO	AUPR↑	83.5/47.0	88.2/51.6	72.8/26.5	81.7/42.9
	FPR95↓	42.9/81.3	49.6/75.4	65.3/89.0	61.8/83.5
AROS	AUPR↑	87.2/**80.5**	97.2/**91.4**	71.0/**65.3**	72.4/**66.8**
	FPR95↓	39.3/**45.2**	15.5/**27.0**	54.2/**67.8**	46.3/**62.7**

We note the superiority of AROS compared to representative methods in terms of robust OOD detection. Notably, AROS, *without* relying on pre-trained models or extra datasets, improves adversarial robust OOD detection performance from 45.9% to **74.0%**. In the OSR setup, the results increased from 52.1% to **69.3%**. Similar gains are observed in robustness against corruptions, as shown in Table 3.

For instance, performance improved from 72.5% to **81.8%** on the CIFAR-10-C vs. CIFAR-100-C setup, and from 61.0% to **67.9%** on the CIFAR-100-C vs. CIFAR-10-C benchmark. Meanwhile, AROS achieves competitive results in clean scenarios (82.0%) compared to state-of-the-art methods like DHM (93.4%), though it should be noted that DHM performs near zero under adversarial attacks. The trade-off between robustness and clean performance is well-known in the field (5; 105; 44), and AROS offers the best overall balance among existing methods. Furthermore, we demonstrate that by using pre-trained models or auxiliary data, AROS's clean performance can be further improved (see Appendix A3). Moreover, we provide additional experiments in Appendix A3 to support our claims.

Table 5. An ablation study (Clean/PGD^{1000}), measured by AUROC (%), on our method with the exclusion of different components while keeping all others intact. The left side is the configurations.

Config	Components						CIFAR10		CIFAR100		ImageNet-1k	
	Adv. Trained Backbone	Fake Sampling	Orthogonal Binary Layer	Extra Data	$\mathcal{L}_{\text{CE}}$	$\mathcal{L}_{\text{SL}}$	CIFAR100	SVHN	CIFAR10	SVHN	Texture	iNaturalist
A	✓	✓	✓	-	✓	-	81.4/17.6	86.9/23.5	68.4/12.7	79.0/16.2	76.4/18.8	82.7/20.3
B	-	✓	✓	-	-	✓	90.1/56.7	93.8/51.5	75.2/41.8	82.0/47.5	**81.9**/36.0	84.9/48.6
C	✓	✓	-	-	-	✓	85.6/67.3	88.2/74.6	66.9/57.1	78.4/63.3	75.4/60.7	79.8/70.2
D	✓	-	✓	-	-	✓	85.3/76.5	89.4/78.1	70.5/61.3	74.4/62.5	76.1/67.4	81.3/72.7
E$_{(Ours)}$	✓	✓	✓	-	-	✓	88.2/80.1	93.0/86.4	74.3/67.0	81.5/70.6	78.3/69.2	84.6/75.3
F $_{(Ours+Data)}$	✓	✓	✓	✓	-	✓	**90.4**/**81.6**	**94.2**/**87.9**	**75.7**/**68.1**	**82.2**/**71.8**	79.2/**70.4**	**85.1**/**76.8**

Classifier Training Strategies for Robust OOD Detection. We assessed the impact of different training strategies on the robust OOD detection performance of various classifiers, including those trained with standard training, adversarial training (AT), and NODE-based methods such as ODENet, LyaDEQ, SODEF, and ASODE. To utilize these classifiers as OOD detectors, various post-hoc score functions were applied, as described in Section 3. The results are presented in Table 4a. In brief, adversarially trained classifiers exhibit enhanced robustness compared to standard training but still fall short of optimal performance. Furthermore, the time-invariance assumption in SODEF leads to improved robust performance relative to ODENet, LyaDEQ, and ASODE by effectively constraining the divergence between output states, which motivated us to explore similar frameworks. Notably, AROS demonstrates superior performance compared to all these approaches.

Implementation details. We use a WideResNet-70-16 model as f_θ (106) and train it for 200 epochs on classification using PGD^{10}. For the integration of h_ϕ, an integration time of $T = 5$ is applied. To implement the orthogonal layer B_η, we utilize the `geotorch.orthogonal` library. Training with the loss $\mathcal{L}_{\text{SL}}$ is performed over 100 epochs. We used SGD as the optimizer, employing a cosine learning rate decay schedule with an initial learning rate of 0.05 and a batch size of 128. See Appendix A3 for more details and additional ablation studies on different components of AROS.

6 Ablation Study

AROS Components. To verify the effectiveness of AROS, we conducted ablation studies across various datasets. The corresponding results are presented in Table 5. In each experiment, individual components were replaced with alternative ones, while the remaining elements were held constant. In *Config A*, we ignored the designed loss function $\mathcal{L}_{\text{SL}}$ and instead utilized the cross-entropy loss function $\mathcal{L}_{\text{CE}}$ for binary classification. *Config B* represents the scenario in which we train the classifier in the first step without adversarial training on the ID data, instead using standard training. This reduces robustness as f_θ becomes more susceptible to perturbations within ID classes, ultimately making the final detector more vulnerable to attacks. In *Config C*, the orthogonal binary layer was replaced with a regular binary layer. In *Config D*, rather than estimating the ID distribution and sampling OOD data in the embedding space, we substituted this process by creating random Gaussian noise in the embedding space as fake OOD data. This removes the conditioning of the fake OOD distribution on the ID data and, as a result, makes them unrelated. This is in line with previous works that have shown that related and nearby auxiliary OOD samples are more useful (43; 24). *Config E* represents our default pipeline. Finally, in *Config F*, we extended AROS by augmenting the fake OOD embedding data with additional OOD images (i.e., Food-101 (107)) alongside the proposed fake OOD strategy. Specifically, we transformed these additional OOD images into the embedding space using f_θ and combined them with the crafted fake embeddings, which led to enhanced performance.

7 Conclusions

In this paper we introduce AROS, a framework for improving OOD detection under adversarial attacks. By leveraging Lyapunov stability theory, AROS drives ID and OOD samples toward stable equilibrium points to mitigate adversarial perturbations. Fake OOD samples are generated in the embedding space, and a tailored loss function is used to enforce stability. Additionally, an orthogonal binary layer is employed to enhance the separation between ID and OOD equilibrium points. Limitations and future directions can be found in the Appendix A6.

REFERENCES

[1] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] Florian Tramer, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations (ICLR)*, 2018.

[7] Yuri S Ledyaev and Eduardo D Sontag. A lyapunov characterization of robust stabilization. *Nonlinear Analysis-Series A Theory and Methods and Series B Real World Applications*, 37(7):813–840, 1999.

[8] Fabio Carrara, Roberto Caldelli, Fabrizio Falchi, and Giuseppe Amato. On the robustness to adversarial examples of neural ode image classifiers. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2019.

[9] Jan Svoboda, Jonathan Masci, Federico Monti, Michael Bronstein, and Leonidas Guibas. Peernets: Exploiting peer wisdom against adversarial attacks. In *International Conference on Learning Representations*, 2019.

[10] Arash Rahnama, Andre T Nguyen, and Edward Raff. Robust design of deep neural networks against adversarial attacks based on lyapunov theory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8178–8187, 2020.

[11] Mingjie Li, Lingshen He, and Zhouchen Lin. Implicit euler skip connections: Enhancing adversarial robustness via numerical stability. In *International Conference on Machine Learning*, pages 5874–5883. PMLR, 2020.

[12] Ivan Dario Jimenez Rodriguez, Aaron Ames, and Yisong Yue. Lyanet: A lyapunov framework for training neural odes. In *International conference on machine learning*, pages 18687–18703. PMLR, 2022.

[13] Zonghan Yang, Tianyu Pang, and Yang Liu. A closer look at the adversarial robustness of deep equilibrium models. *Advances in Neural Information Processing Systems*, 35:10448–10461, 2022.

[14] Sergey Dashkovskiy, Oleksiy Kapustyan, and Vitalii Slynko. Robust stability of a nonlinear ode-pde system. *SIAM Journal on Control and Optimization*, 61(3):1760–1777, 2023.

[15] Mustafa Zeqiri, Mark Niklas Müller, Marc Fischer, and Martin Vechev. Efficient certified training and robustness verification of neural odes. *arXiv preprint arXiv:2303.05246*, 2023.

[16] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021.

[17] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1893–1902, 2015.

[18] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34, 2021.

[19] Senqi Cao and Zhongfei Zhang. Deep hybrid models for out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4733–4743, 2022.

[20] Feng Xue, Zi He, Chuanlong Xie, Falong Tan, and Zhenguo Li. Boosting out-of-distribution detection with multiple pre-trained models. *arXiv preprint arXiv:2212.12720*, 2022.

[21] Xuefeng Du, Zhen Fang, Ilias Diakonikolas, and Yixuan Li. How does unlabeled data provably help out-of-distribution detection? *arXiv preprint arXiv:2402.03502*, 2024.

[22] Ronald Carl Petersen, Paul S Aisen, Laurel A Beckett, Michael C Donohue, Anthony Collins Gamst, Danielle J Harvey, CR Jack Jr, William J Jagust, Leslie M Shaw, Arthur W Toga, et al. Alzheimer's disease neuroimaging initiative (adni) clinical characterization. *Neurology*, 74(3):201–209, 2010.

[23] Adam Goodge, Bryan Hooi, See Kiong Ng, and Wee Siong Ng. Robustness of autoencoders for anomaly detection under adversarial impact. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 1244–1250, 2021.

[24] 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.

[25] Emily Roberts and Rajesh Gupta. Applying novelty detection techniques for quality assurance in manufacturing. *Journal of Industrial and Production Engineering*, 38(4):251–262, 2021.

[26] Shao-Yuan Lo, Poojan Oza, and Vishal M Patel. Adversarially robust one-class novelty detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[27] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

[28] Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Robust out-of-distribution detection for neural networks. *arXiv preprint arXiv:2003.09711*, 2020.

[29] Rui Shao, Pramuditha Perera, Pong C Yuen, and Vishal M Patel. Open-set adversarial defense. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pages 682–698. Springer, 2020.

[30] Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Atom: Robustifying out-of-distribution detection using outlier mining. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part III 21*, pages 430–445. Springer, 2021.

[31] Alexander Meinke, Julian Bitterwolf, and Matthias Hein. Provably adversarially robust detection of out-of-distribution data (almost) for free. *Advances in Neural Information Processing Systems*, 35:30167–30180, 2022.

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

[33] 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.

[34] Mohammad Azizmalayeri, Arshia Soltani Moakhar, Arman Zarei, Reihaneh Zohrabi, Mohammad Manzuri, and Mohammad Hossein Rohban. Your out-of-distribution detection method is not robust! *Advances in Neural Information Processing Systems*, 35:4887–4901, 2022.

[35] Nicola Franco, Daniel Korth, Jeanette Miriam Lorenz, Karsten Roscher, and Stephan Guennemann. Diffusion denoised smoothing for certified and adversarial robust out-of-distribution detection. *arXiv preprint arXiv:2303.14961*, 2023.

[36] Louis Béthune, Paul Novello, Thibaut Boissin, Guillaume Coiffier, Mathieu Serrurier, Quentin Vincenot, and Andres Troya-Galvis. Robust one-class classification with signed distance function using 1-lipschitz neural networks. *arXiv preprint arXiv:2303.01978*, 2023.

[37] Hossein Mirzaei, Mohammad Jafari, Hamid Reza Dehbashi, Ali Ansari, Sepehr Ghobadi, Masoud Hadi, Arshia Soltani Moakhar, Mohammad Azizmalayeri, Mahdieh Soleymani Baghshah, and Mohammad Hossein Rohban. Rodeo: Robust outlier detection via exposing adaptive out-of-distribution samples. In *Forty-first International Conference on Machine Learning*, 2024.

[38] Peter Lorenz, Mario Fernandez, Jens Müller, and Ullrich Köthe. Deciphering the definition of adversarial robustness for post-hoc ood detectors. *arXiv preprint arXiv:2406.15104*, 2024.

[39] Xuefeng Du, Zhaoning Wang, Mu Cai, and Yixuan Li. Vos: Learning what you don't know by virtual outlier synthesis. *arXiv preprint arXiv:2202.01197*, 2022.

[40] Peyman Morteza and Yixuan Li. Provable guarantees for understanding out-of-distribution detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(7):7831–7840, 2022.

[41] 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.

[42] Xin Zou and Weiwei Liu. On the adversarial robustness of out-of-distribution generalization models. *Advances in Neural Information Processing Systems*, 36, 2024.

[43] 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.

[44] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[45] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31, 2018.

[46] Preetum Nakkiran. Adversarial robustness may be at odds with simplicity. *arXiv preprint arXiv:1901.00532*, 2019.

[47] Sravanti Addepalli, Samyak Jain, et al. Efficient and effective augmentation strategy for adversarial training. *Advances in Neural Information Processing Systems*, 35:1488–1501, 2022.

[48] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[49] 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.

[50] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. *CoRR*, abs/2003.01690, 2020.

[51] Ye Liu, Yaya Cheng, Lianli Gao, Xianglong Liu, Qilong Zhang, and Jingkuan Song. Practical evaluation of adversarial robustness via adaptive auto attack, 2022.

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

[53] Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, Wenxuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, et al. Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35:32598–32611, 2022.

[54] 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.

[55] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3614–3631, 2020.

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

[57] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, pages 6571–6583, 2018.

[58] Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[59] Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models. In *International Conference on Learning Representations (ICLR)*, 2019.

[60] 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.

[61] Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1563–1572, 2016.

[62] Kai Liu, Zhihang Fu, Chao Chen, Sheng Jin, Ze Chen, Mingyuan Tao, Rongxin Jiang, and Jieping Ye. Category-extensible out-of-distribution detection via hierarchical context descriptions. *Advances in Neural Information Processing Systems*, 36, 2024.

[63] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in neural information processing systems*, 33:11839–11852, 2020.

[64] Hanshu Yan, Jiawei Du, Vincent YF Tan, and Jiashi Feng. On robustness of neural ordinary differential equations. *arXiv preprint arXiv:1910.05513*, 2019.

[65] J Zico Kolter and Gaurav Manek. Learning stable deep dynamics models. *Advances in neural information processing systems*, 32, 2019.

[66] Xiyuan Li, Zou Xin, and Weiwei Liu. Defending against adversarial attacks via neural dynamic system. *Advances in Neural Information Processing Systems*, 35:6372–6383, 2022.

[67] Haoyu Chu, Shikui Wei, Ting Liu, Yao Zhao, and Yuto Miyatake. Lyapunov-stable deep equilibrium models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(10):11615–11623, 2024.

[68] Brandon Amos, Lei Xu, and J Zico Kolter. Input convex neural networks. In *International conference on machine learning*, pages 146–155. PMLR, 2017.

[69] Qiyu Kang, Yang Song, Qinxu Ding, and Wee Peng Tay. Stable neural ode with lyapunov-stable equilibrium points for defending against adversarial attacks. *Advances in Neural Information Processing Systems*, 34:14925–14937, 2021.

[70] Xuefeng Du, Yiyou Sun, Xiaojin Zhu, and Yixuan Li. Dream the impossible: Outlier imagination with diffusion models. *arXiv preprint arXiv:2309.13415*, 2023.

[71] Charles Dawson, Zengyi Qin, Sicun Gao, and Chuchu Fan. Safe nonlinear control using robust neural lyapunov-barrier functions. In *Conference on Robot Learning*, pages 1724–1735. PMLR, 2022.

[72] Nur Uddin, Hendra G Harno, and Rianto Adhy Sasongko. Altitude control system design of bicopter using lyapunov stability approach. In *2021 International Symposium on Electronics and Smart Devices (ISESD)*, pages 1–6. IEEE, 2021.

[73] Aman Sharma and Narendra Kumar. Lyapunov stability theory based non linear controller design for a standalone pv system. In *2020 IEEE International Conference for Innovation in Technology (INOCON)*, pages 1–7. IEEE, 2020.

[74] Ya-Chien Chang, Nima Roohi, and Sicun Gao. Neural lyapunov control. In *Advances in Neural Information Processing Systems*, pages 3240–3249, 2019.

[75] Luc PJ Sträter, Mohammadreza Salehi, Efstratios Gavves, Cees GM Snoek, and Yuki M Asano. Generalad: Anomaly detection across domains by attending to distorted features. *arXiv preprint arXiv:2407.12427*, 2024.

[76] Matan Jacob Cohen and Shai Avidan. Transformaly–two (feature spaces) are better than one. *arXiv preprint arXiv:2112.04185*, 2021.

[77] Swaminathan Venkataramanan et al. Gaussian latent representations for uncertainty estimation using mahalanobis distance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2023.

[78] Xun Jin et al. Shape your space: A gaussian mixture regularization approach to latent space geometry optimization. In *Advances in Neural Information Processing Systems*, pages 12688–12701, 2021.

[79] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. 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.

[80] 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.

[81] Stefano Massaroli, Atsushi Yamashita, and Hajime Asama. Dissecting neural odes. In *Advances in Neural Information Processing Systems*, 2020.

[82] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[83] Chi-Tsong Chen. *Linear system theory and design*. Saunders college publishing, 1984.

[84] DK Arrowsmith and CM Place. Differential equations, maps and chaotic behavior. *Champman and Hall, London*, 1995.

[85] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[86] Shuai Li, Kui Jia, Yuxin Wen, Tongliang Liu, and Dacheng Tao. Orthogonal deep neural networks. *IEEE transactions on pattern analysis and machine intelligence*, 43(4):1352–1368, 2019.

[87] Mufan Li, Mihai Nica, and Dan Roy. The future is log-gaussian: Resnets and their infinite-depth-and-width limit at initialization. *Advances in Neural Information Processing Systems*, 34:7852–7864, 2021.

[88] R Pascanu. Understanding the exploding gradient problem. *arXiv preprint arXiv:1211.5063*, 2012.

[89] Patricia Pauli, Anne Koch, Julian Berberich, Paul Kohler, and Frank Allgöwer. Training robust neural networks using lipschitz bounds. *IEEE Control Systems Letters*, 6:121–126, 2021.

[90] Yujia Huang, Huan Zhang, Yuanyuan Shi, J Zico Kolter, and Anima Anandkumar. Training certifiably robust neural networks with efficient local lipschitz bounds. *Advances in Neural Information Processing Systems*, 34:22745–22757, 2021.

[91] Monty-Maximilian Zühlke and Daniel Kudenko. Adversarial robustness of neural networks from the perspective of lipschitz calculus: A survey. *ACM Computing Surveys*, 2024.

[92] Cong Xu, Xiang Li, and Min Yang. An orthogonal classifier for improving the adversarial robustness of neural networks. *Information Sciences*, 591:251–262, 2022.

[93] Sima Behpour, Thang Long Doan, Xin Li, Wenbin He, Liang Gou, and Liu Ren. Gradorth: a simple yet efficient out-of-distribution detection with orthogonal projection of gradients. *Advances in Neural Information Processing Systems*, 36, 2024.

[94] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[95] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[96] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, volume 2011, page 5, 2011.

[97] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[98] 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.

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

[100] Jia-Bin Xu and Jianxiong Xiao. Isun: Large-scale scene understanding. In *arXiv preprint arXiv:1502.03509*, 2015.

[101] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need? *ICLR*, 2022.

[102] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[103] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. In *arXiv preprint arXiv:1708.07747*, 2017.

[104] Jeremy Howard. Imagenette. `https://github.com/fastai/imagenette`. Accessed: 2024-09-27.

[105] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018.

[106] Sergey Zagoruyko. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[107] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014.

[108] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[109] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022.

[110] Yong Chen, Xuedong Li, Xu Wang, Peng Hu, and Dezhong Peng. Diffilter: Defending against adversarial perturbations with diffusion filter. *IEEE Transactions on Information Forensics and Security*, 2024.

[111] Kaiyu Song, Hanjiang Lai, Yan Pan, and Jian Yin. Mimicdiffusion: Purifying adversarial perturbation via mimicking clean diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24665–24674, 2024.

[112] Minjong Lee and Dongwoo Kim. Robust evaluation of diffusion-based adversarial purification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 134–144, 2023.

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

[114] Runpeng Yu, Songhua Liu, Xingyi Yang, and Xinchao Wang. Distribution shift inversion for out-of-distribution prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3592–3602, 2023.

[115] Yuanpu Cao, Lu Lin, and Jinghui Chen. Adversarially robust industrial anomaly detection through diffusion model. *arXiv preprint arXiv:2408.04839*, 2024.

[116] Xiangyuan Yang, Jie Lin, Hanlin Zhang, Xinyu Yang, and Peng Zhao. Improving the transferability of adversarial examples via direction tuning. *arXiv preprint arXiv:2303.15109*, 2023.

[117] Qinliang Lin, Cheng Luo, Zenghao Niu, Xilin He, Weicheng Xie, Yuanbo Hou, Linlin Shen, and Siyang Song. Boosting adversarial transferability across model genus by deformation-constrained warping. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 3459–3467, 2024.

[118] Han Wu, Guanyan Ou, Weibin Wu, and Zibin Zheng. Improving transferable targeted adversarial attacks with model self-enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24615–24624, 2024.

[119] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th annual computer security applications conference*, pages 113–125, 2019.

[120] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33:3533–3545, 2020.

[121] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, pages 2196–2205. PMLR, 2020.

[122] Matteo Croce and Matthias Hein. Reliable evaluation of adversarial robustness with autoattack. In *International Conference on Machine Learning*, pages 2701–2711. PMLR, 2020.

[123] Maria Andriushchenko, Yang Song, and Zachary C Lipton. Square attack: a query-efficient black-box adversarial attack via random search. In *International Conference on Learning Representations*, 2020.

[124] Mu Zhou, Lucas Stoffl, Mackenzie Weygandt Mathis, and Alexander Mathis. Rethinking pose estimation in crowds: Overcoming the detection information bottleneck and ambiguity. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14689–14699, October 2023.

[125] Shaokai Ye, Anastasiia Filippova, Jessy Lauer, Maxime Vidal, Steffen Schneider, Tian Qiu, Alexander Mathis, and Mackenzie W. Mathis. Superanimal pretrained pose estimation models for behavioral analysis. *Nature Communications*, 15, 2024.

[126] Zijiao Chen, Jiaxin Qing, Tiange Xiang, Wan Lin Yue, and Juan Helen Zhou. Seeing beyond the brain: Conditional diffusion model with sparse masked modeling for vision decoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22710–22720, 2023.

Appendix

A1 Review of Related Work

In our experimental evaluation we compare our results with works that have proposed a defense mechanism for their OOD detection method, as well as those that have not. We will describe each separately below. Subsequently, we will detail the post-hoc score functions.

Adversarially Robust OOD Detection Methods. Previous adversarially robust methods for OOD detection have built on the outlier exposure (27) technique and conducted adversarial training on a union of exposed outliers and ID samples. These methods primarily aim to enhance the outlier exposure technique by improving the diversity of the extra OOD image dataset. To tackle this, ALOE leverages an auxiliary OOD dataset and perturbs the samples to adversarially maximize the KL-divergence between the model's output and a uniform distribution. It is important to note that the adversarial evaluation in their pipeline was less rigorous than that in our approach, due primarily to the limited budget allocated for adversarial attacks. ATOM, in contrast, selectively samples informative outliers rather than using random outliers. The adversarial evaluation of ATOM is also not entirely standard, as it exclusively targets OOD test samples. However, a more comprehensive approach would involve attacking both ID and OOD test samples to ensure enhanced robustness. Meanwhile, ATD employs a GAN to generate auxiliary OOD images for adversarial training, instead of relying on an external dataset. RODEO aims to demonstrate that enhancing the diversity of auxiliary OOD images, while maintaining their stylistic and semantic alignment with ID samples, will improve robustness. Consequently, they utilize a pretrained CLIP model (108) and a diffusion model for OOD image synthesis.

Standard OOD Detection Methods. CSI enhances the outlier detection task by building on standard contrastive learning through the introduction of 'distributionally-shifted augmentations'—transformations that encourage the model to treat augmented versions of a sample as OOD. This approach enables the model to learn representations that more effectively distinguish between ID and OOD samples. The paper also proposes a detection score based on the contrastive features learned through this training scheme, which demonstrates effectiveness across various OOD detection scenarios, including one-class, multi-class, and labeled multi-class settings. DHM involves modeling the joint density of data and labels in a single forward pass. By factorizing this joint density into three sources of uncertainty (aleatoric, distributional, and parametric), DHMs aim to distinguish in-distribution samples from OOD samples. To achieve computational efficiency and scalability, the method employs weight normalization during training and utilizes normalizing flows (NFs) to model the probability distribution of the features. The key idea is to use bi-Lipschitz continuous mappings, enabled by spectral normalization, which allows the use of state-of-the-art deep neural networks for learning expressive and geometry-preserving representations of data. VOS uses virtual outliers to regularize the model's decision boundary and improve its ability to distinguish between ID and OOD data. The framework includes an unknown-aware training objective that uses contrastive learning to shape the uncertainty surface between the known data and the synthesized outliers. This method is effective for both object detection and image classification. CATEX use of two hierarchical contexts—perceptual and spurious—to describe category boundaries more precisely through automatic prompt tuning in vision-language models like CLIP. The perceptual context distinguishes between different categories (e.g., cats vs. apples), while the spurious context helps identify samples that are similar but not ID (e.g., distinguishing cats from panthers). This hierarchical structure helps create more precise category boundaries.

Post-hoc Methods for OOD Detection. Post-hoc methods for OOD detection are approaches applied after training a classifier on ID samples. These methods utilize information from the classifier to indicate OOD detection. A simple but effective approach is the Maximum Softmax Probability (MSP) method. Applied to a K-class classifier f_c, MSP returns $\max_{c\in\{1,2,\ldots,K\}} f_c(x)$ as the likelihood that the sample x belongs to the ID set. In contrast, OpenMax replaces the softmax layer with a calibrated layer that adjusts the logits by fitting a class-wise probability model, such as the Weibull distribution. Another perspective on OOD detection is to measure the distance of a sample to class-conditional distributions. The Mahalanobis distance (MD) is a prominent method for this. For an ID set with K classes, MD-based approaches 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 follows:

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

The MD for a sample z' relative to class k is defined as: $MD_k(z') = (z' - \mu_k)^T \Sigma^{-1}(z' - \mu_k)$.

The final score MD used for OOD detection is given by: $\text{score}_{\text{MD}}(x') = -\min_k\{MD_k(z')\}$

A2 ALGORITHM

Here we provide pseudocode for our proposed AROS framework, designed for adversarially robust OOD detection. We begin by leveraging adversarial training on ID data to obtain robust feature representations, utilizing the well-known practice of adversarial training with 10-step PGD. These features are then used to fit class-conditional multivariate Gaussians, from which we sample low-likelihood regions to generate fake OOD embeddings, effectively creating a proxy for real OOD data in the embedding space. By constructing a balanced training set of ID and fake OOD embeddings, we then employ a stability-based objective using a NODE pipeline, coupled with an orthogonal binary layer. This layer maximizes the separation between the equilibrium points of ID and OOD samples, promoting a robust decision boundary under perturbations. During inference, we compute the OOD score based on the orthogonal binary layer's output, enabling the model to reliably distinguish ID samples from OOD samples, even in the presence of adversarial attacks.

Algorithm 1 Adversarially Robust OOD Detection through Stability (AROS)

Require: ID training samples $\mathcal{D}_{\text{train}}^{\text{in}}$ consisting of N samples spanning k classes, a k-class classifier f_θ, a time-invariant NODE h_ϕ, an orthogonal binary layer B_η, an ℓ_∞ norm constraint, perturbation budget ϵ set to $\frac{4}{255}$ for low-resolution and $\frac{8}{255}$ for high-resolution, Lyapunov-based loss $\mathcal{L}_{\text{ST}}$.

Ensure: Adversarially Robust OOD Detector

1:
2: **Step A-1. Adversarial Training of Classifier f_θ on ID:**
3: **for** each sample $(x, y) \in \mathcal{D}_{\text{train}}^{\text{in}}$ **do**
4: Generate adversarial example x^* using PGD^{10} to maximize the cross-entropy loss:
5: $x^* \leftarrow x + \alpha \cdot \text{sign}(\nabla_x \mathcal{L}_{\text{CE}}(f_\theta(x), y))$, constrained by ϵ
6: Train the classifier f_θ on (x^*, y) to improve robustness
7: **end for**
8: **Step A-2. Feature Extraction:**
9: Map ID samples to the robust embedding space: $\text{ID}_{\text{features}} = f_\theta(\mathcal{D}_{\text{train}}^{\text{in}})$
10:
11: **Step B. Fake OOD Embedding Generation:**
12: Fit a class-conditional multivariate Gaussian $\mathcal{N}(\mu_j, \Sigma_j)$ on $\text{ID}_{\text{features}}$ for each class j
13: **for** $0 \leq j < K$ **do**
14: **for** $0 \leq i < \left[\frac{N}{K}\right]$ **do**
15: Sample synthetic OOD embeddings r from the low-likelihood regions of $\mathcal{N}(\mu_j, \Sigma_j)$
16: $r_{\text{OOD}} \leftarrow r_{\text{OOD}} \cup \{r\}$
17: **end for**
18: **end for**
19: Construct a balanced training set $X_{\text{train}} = \text{ID}_{\text{features}} \cup r_{\text{OOD}}$
20:
21: **Step C. Lyapunov Stability Objective Function:**
22: Create the pipeline $B_\eta(h_\phi(.))$ and train it on X_{train} using $\mathcal{L}_{\text{ST}}$
23:
24: **Step D. Inference:**
25: **for** each test sample x_{test} **do**
26: Compute the OOD score based on the probability output of the orthogonal binary layer for each input image x: OOD score$(x) = f_\theta(B_\eta(h_\phi(x)))[1]$
27: **end for**

A3 SUPPLEMENTARY EXPERIMENTAL RESULTS AND DETAILS FOR AROS

A3.1 ADDITIONAL EXPERIMENTAL RESULTS

Each experiment discussed in the main text was repeated 10 times, with the reported results representing the mean of these trials. Here, we provide the standard deviation across these runs, as summarized in Table 6a.

Table 6a: Standard deviation of AROS performance under both clean and PGD, across 10 repeated experiments 'Clean/PGD^{1000}'.

AROS	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
Mean Performance	88.2 / 80.1	93.0 / 86.4	90.6 / 82.4	74.3 / 67.0	81.5 /7 0.6	74.3 / 68.1	78.3 / 69.2	84.6 / 75.3	79.4 / 69.0
Standard Deviation	± 0.9 / ± 1.6	± 0.7 / ± 1.3	± 0.6/± 0.9	± 1.3 / ± 1.8	± 1.4 / ± 2.0	± 1.8 / ± 2.3	± 2.5 / ± 3.1	± 1.9 / ± 3.0	± 2.8 / ± 3.7

A3.1.1 COMPARATIVE ANALYSIS OF AROS AND DIFFUSION-BASED PURIFICATION METHODS

We present additional experiments to further demonstrate the effectiveness of AROS. We compare AROS's performance against diffusion-based (purification) methods, with results detailed in Table 7a.

Purification techniques in adversarial training aim to enhance model robustness by 'purifying' or 'denoising' adversarial examples prior to feeding them into the model. The primary goal is to mitigate adversarial perturbations through preprocessing, often by leveraging neural networks or transformation-based methods to restore perturbed inputs to a state resembling clean data. A common approach involves training a generative diffusion model on the original training samples, which is then utilized as a purification module (109; 110; 111). In contrast, certain approaches categorize stable NODE-based methods as non-diffusion-based strategies for improving robustness (112).

However, in the context of OOD detection, purification using diffusion methods (113) may not be effective. This is primarily because diffusion models trained on ID samples tend to shift the features of unseen OOD data towards ID features during the reverse process. Such a shift can mistakenly transform OOD samples into ID, compromising both OOD detection performance and clean robustness (114). To emphasize this distinction, we compare our approach against purification using diffusion-based models. Specifically, we adopt the AdvRAD (115) setup for OOD detection and present a comparison with our method in Table 7a.

Table 7a: Comparison of AROS and AdvRAD under clean and PGD$^{1000}(l_\infty)$ evaluation, measured by AUROC (%). The table cells denote results in the 'Clean/PGD^{1000}' format. The perturbation budget ϵ is set to $\frac{8}{255}$ for low-resolution datasets and $\frac{4}{255}$ for high-resolution datasets.

Method	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
AdvRAD	61.0/49.0	68.5/52.7	66.8/50.7	54.8/49.4	60.1/52.0	54.8/50.2	57.7/51.0	62.4/56.5	58.5/50.9
AROS	88.2/80.1	93.0/86.4	90.6/ 82.4	74.3/67.0	81.5/70.6	74.3/68.1	78.3/ 69.2	84.6/75.3	79.4/69.0

A3.1.2 AROS UNDER ADDITIONAL ADVERSARIAL ATTACKS

In this section, we further emphasize the robustness of AROS by evaluating its performance under different attacks. First, we conducted additional experiments considering different perturbation norms, specifically substituting the l_∞ norm with the l_2 norm in PGD^{1000}, setting $\epsilon = \frac{128}{255}$. The outcomes of this evaluation are presented in Table 8a. See Figure **??** for an illustration of some clean and perturbed samples.

Next, we evaluated the robustness of AROS under PGD$^{1000}(l_\infty)$ with a higher perturbation budget (i.e., $\epsilon = \frac{16}{255}$), as shown in Table 8b. Furthermore, the performance of AROS under AutoAttack (AA) and adaptive AA is reported in Table 9a.

Additionally, evaluating a model against a variety of transfer-based adversarial attacks is crucial for understanding its robustness in real-world scenarios, where adversarial examples crafted for one

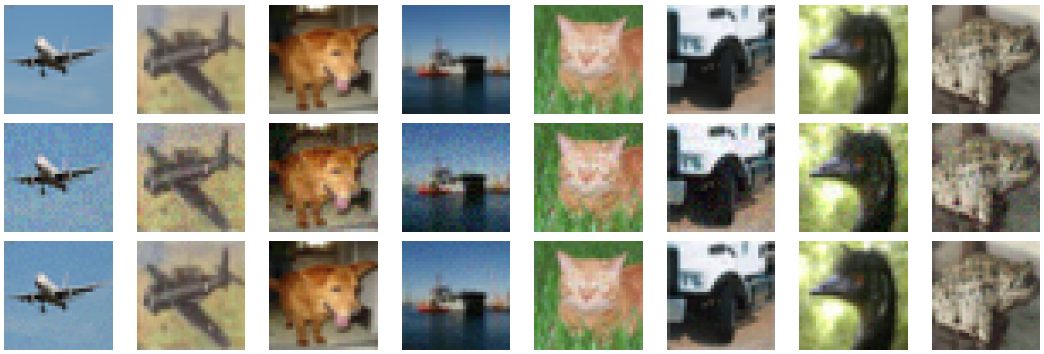

Figure 3: Visualization of clean and perturbed images from the CIFAR-10 dataset to illustrate the impact of perturbations on semantic content. The first row depicts clean images, while the second and third rows show images perturbed with L_∞ norm of $\frac{8}{255}$ and L_2 norm of $\frac{128}{255}$, respectively. Despite the added perturbations, the semantic content of the images remains unchanged, demonstrating that robustness expectations from models under these perturbations are fair.

model can successfully deceive others. These attacks simulate diverse and challenging conditions by targeting features such as generalization, input transformations, and model invariance, providing a comprehensive assessment of the model's resilience. Such evaluations reveal potential vulnerabilities, measure performance under adversarial conditions, and offer insights into the model's ability to generalize across different attack strategies.

Motivated by this, we evaluate our method, as well as DHM and RODEO, against several adversarial attacks, including DTA (116), DeCoWA (117), and SASD-WS (118). We selected DHM and RODEO for comparison due to their strong performance in clean detection and robust detection, respectively. For these experiments, we utilized the implementation provided in the `https://github.com/Trustworthy-AI-Group/TransferAttack` repository. The results of this evaluation are presented in Table 9b.

The consistent and superior performance of AROS against the aforementioned attacks underscores its effectiveness in adversarial scenarios.

Table 8a: Evaluation of AROS under PGD$^{1000}(l_2)$, where $\epsilon = \frac{128}{255}$ is used for low-resolution datasets and $\epsilon = \frac{64}{255}$ for high-resolution datasets. The results are measured by AUROC (%).

Attack	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
PGD$^{1000}(l_2)$	81.6	87.1	83.7	67.4	71.0	69.3	70.5	75.9	69.8

Table 8b: : Evaluation of AROS under PGD$^{1000}(l_\infty)$, where $\epsilon = \frac{16}{255}$ is used for all datasets, including both high- and low-resolution ones. The results are reported in terms of AUROC (%).

Attack	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
PGD$^{1000}(l_\infty)$	70.3	78.4	72.6	58.2	63.9	60.4	60.5	66.8	62.7

A3.2 Detecting Backdoored Samples

We conducted an experiment to evaluate our method's ability to detect clean samples (without triggers) as ID and poisoned samples as OOD. The results are presented in this section, with STRIP (119) included as a baseline method for identifying Trojaned samples.

Table 9a: Evaluation of AROS under AutoAttack and Adaptive AA. The perturbation budget ϵ is set to $\frac{8}{255}$ for low-resolution datasets and $\frac{4}{255}$ for high-resolution datasets.

Attack	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
AutoAttack	78.9	83.4	80.2	66.5	70.2	68.9	67.4	73.6	67.1
Adaptive AA	76.4	82.9	78.6	65.2	67.4	68.3	66.1	70.5	66.9

Table 9b: Comparison of AROS, DHM, and RODEO under Transfer-Based Attacks. The perturbation budget ϵ is set to $\frac{16}{255}$. Results compare AROS, DHM, and RODEO across DTA, DeCoWA, and SASD-WS attacks.

Attack	Methods	CIFAR-10			CIFAR-100			ImageNet-1k		
		CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
DTA	AROS	**84.7**	**86.4**	88.2	**71.5**	**76.6**	**71.7**	**76.2**	**83.5**	**75.9**
	RODEO	61.6	68.5	**89.3**	51.1	63.1	70.9	61.1	60.6	59.2
	DHM	72.9	78.3	78.9	75.4	79.1	74.1	55.9	59.2	57.6
DeCoWA	AROS	**83.8**	**85.2**	**86.8**	**69.6**	**77.2**	**69.4**	**72.6**	**79.5**	**74.8**
	RODEO	57.8	64.6	80.4	43.0	57.7	67.5	54.7	54.8	52.5
	DHM	60.1	64.3	64.9	68.9	62.1	60.8	54.9	58.4	56.0
SASD-WS	AROS	**82.3**	**84.9**	**81.7**	**67.6**	**75.0**	**67.7**	**72.4**	**77.5**	**71.2**
	RODEO	49.5	53.4	79.4	44.7	56.5	66.1	53.2	46.9	47.0
	DHM	64.6	69.3	61.3	56.3	67.8	59.7	53.1	51.3	49.2

It is important to highlight a key distinction in the context of OOD detection, which underpins our study. In traditional OOD detection scenarios, ID and OOD datasets typically differ at the semantic level (e.g., CIFAR-10 versus SVHN). However, detecting backdoored samples presents a unique challenge: the presence of triggers modifies clean images at the pixel level rather than the semantic level. Consequently, this task often hinges on texture-level differences rather than semantic distinctions, necessitating models designed specifically to leverage this inductive bias effectively.

Despite these challenges, our results demonstrate that AROS achieves strong performance in detecting poisoned samples. This underscores the versatility and effectiveness of AROS, even when applied to the nuanced problem of backdoor detection. The results are presented in Table 10a.

Table 10a: Effectiveness of the Proposed Method for Detecting Backdoor Attack Samples across CIFAR-10, CIFAR-100, and GTSRB datasets. The results are presented for different backdoor attack methods, demonstrating the performance of AROS and STRIP.

Method	Backdoor Attack	CIFAR-10	CIFAR-100	GTSRB
AROS	Badnets	**80.3**	67.5	72.8
	Wanet	**62.7**	**58.9**	**54.4**
	SSBA	**57.2**	**72.6**	**66.0**
STRIP	Badnets	79.2	**86.0**	**87.1**
	Wanet	39.5	48.5	35.6
	SSBA	36.4	68.5	64.1

A3.2.1 Architecture comparison & Enhancing Clean Performance with Transfer Learning

Additionally, we investigate the influence of replacing our default backbone architecture, WideResNet, with alternative architectures (Table 11a). We also explore transfer learning techniques to enhance clean performance by leveraging robust pre-trained classifiers.

Lastly, while AROS demonstrates a significant improvement of up to 40% in adversarial robustness, it shows a performance gap of approximately 10% when compared to state-of-the-art clean detection methods. Although this trade-off between robustness and clean performance is well-documented in the literature (5; 105; 44), our aim is to enhance clean performance. A promising approach would involve enhancing clean performance while preserving robustness by leveraging transfer learning through distillation from a large, robust pretrained model, rather than training the classifier

from scratch, as is currently done in the pipeline. Specifically, by utilizing adversarially pretrained classifiers on ImageNet (120), we aim to improve our clean performance. Results in lower part of Table 11a indicate that leveraging such pretrained models can improve clean performance by 6%. Moreover, we also consider using different architectures trained from scratch to further demonstrate the robustness of AROS across various backbones.

Table 11a: Ablation study on different backbone architectures & Transfer learning. Results are reported under clean and $\text{PGD}^{1000}(l_\infty)$ evaluations, measured by AUROC (%). Each table cell presents results in the 'Clean/PGD^{1000}' format.

† Denotes that the backbone is adversarially pretrained on ImageNet.

Backbone (f_θ)	**CIFAR-10**			**CIFAR-100**			**ImageNet-1k**		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
WideResNet (default)	88.2/80.1	93.0/86.4	90.6/ 82.4	74.3/67.0	81.5/70.6	74.3/68.1	78.3/ 69.2	84.6/75.3	79.4/69.0
PreActResNet	83.5/74.6	92.1/82.4	87.9/79.6	69.6/62.2	76.7/65.2	69.9/62.7	73.4/69.3	82.2/73.6	76.8/65.6
ResNet18	82.2/74.4	90.5/80.7	84.5/81.0	70.7/63.7	79.0/66.4	74.6/66.4	77.4/64.4	81.8/72.3	78.7/67.8
ResNet50	86.5/79.2	92.5/86.1	90.8/82.7	74.5/66.4	81.7/69.6	72.8/68.3	77.9/67.4	82.7/75.1	77.9/68.5
WideResNet †	91.2/75.2	97.4/80.9	92.1/77.9	80.5/60.1	87.8/70.0	76.7/62.9	-	-	-
PreActResNet†	93.1/79.5	96.3/84.4	96.5/76.6	75.4/62.2	83.1/66.7	79.4/64.9	-	-	-
ResNet18†	90.2/78.0	97.9/83.9	97.5/78.6	76.2/63.7	83.0/65.2	78.0/62.2	-	-	-
ResNet50†	90.6/76.8	95.5/82.3	94.6/74.7	79.5/64.9	85.7/69.7	78.7/45.5	-	-	-
ViT-b-16†	94.3/71.6	97.8/79.7	94.6/76.8	85.2/61.0	87.6/66.9	82.7/59.2	-	-	-

A3.3 HYPERPARAMETER ABLATION STUDY

A3.3.1 ABLATION STUDY ON HYPERPARAMETER OF FAKE GENERATION (β)

To craft fake samples, we fit a GMM to the embedding space of ID samples. The objective is to sample from the GMM such that the likelihood of the samples is low, ensuring they do not belong to the ID distribution and are located near its boundaries. This approach generates near-OOD samples, which are valuable for understanding the distribution manifold and improving the detector's performance.

In practice, an encoder is used as a feature extractor to obtain embeddings of ID samples. A GMM is then fitted to these embeddings, and the likelihood of each training sample under the GMM is computed. The embeddings are subsequently sorted based on their likelihoods, and the β-th minimum likelihood of the training samples is used as a threshold. Random samples are drawn from the GMM, and their likelihoods are compared against this threshold. If a sample's likelihood is lower than the threshold, it is retained as fake OOD; otherwise, it is discarded.

For instance, setting $\beta = 0.1$ ensures that the crafted fake OOD samples have a likelihood of belonging to the ID distribution that is lower than 90% of the ID samples. Conversely, setting $\beta = 0.5$ corresponds to randomly sampling from the ID distribution, rather than targeting low-likelihood regions, which leads to poor performance. An ablation study in our manuscript explores the effects of varying β values on model performance.

The sensitivity of AROS to this hyperparameter is analyzed in detail. As shown in Table 12a, AROS demonstrates consistent performance for small values of β. Our experimental results indicate that selecting β values in the range $[0.0, 0.1]$ achieves optimal performance, highlighting the robustness of AROS to changes in β.

Table 12a: Ablation study on the β hyperparameter. Results are reported under clean and PGD$^{1000}(l_\infty)$ evaluations, measured by AUROC (%). Each table cell presents results in the 'Clean/PGD^{1000}' format.

Hyperparameter	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
$\beta = 0.001$(default)	88.2/80.1	93.0/86.4	90.6/ 82.4	74.3/67.0	81.5/70.6	74.3/68.1	78.3/ 69.2	84.6/75.3	79.4/69.0
$\beta = 0.01$	87.3/80.2	92.2/85.3	89.4/80.7	74.7/65.8	80.2/68.9	73.1/66.6	78.1/69.5	83.0/74.2	79.0/68.5
$\beta = 0.025$	86.3/79.6	93.8/85.0	89.3/83.1	72.7/67.5	81.7/70.8	74.3/68.0	76.7/67.7	84.7/75.3	79.8/69.7
$\beta = 0.05$	86.5/80.1	92.5/86.1	90.8/82.7	74.5/66.4	81.7/69.6	72.8/68.3	77.9/67.4	82.7/75.1	77.9/68.5
$\beta = 0.075$	88.4/78.4	92.7/85.2	89.1/82.3	72.4/65.7	81.5/69.4	72.5/67.9	78.6/69.7	84.8/73.5	78.8/69.3
$\beta = 0.1$	86.6/79.5	92.0/85.4	90.9/80.9	72.3/65.8	80.3/69.4	73.3/66.7	77.8/67.5	84.1/74.9	78.2/67.8
$\beta = 0.25$	79.4/67.5	89.0/79.5	78.7/69.4	67.7/54.3	69.5/60.3	70.1/62.7	62.6/65.0	65.2/57.9	70.9/62.0
$\beta = 0.5$	65.4/53.0	77.4/62.3	64.7/54.6	57.5/40.9	52.4/49.2	49.1/41.9	55.8/48.9	67.4/59.7	57.9/53.8

A3.3.2 ABLATION STUDY ON HYPERPARAMETERS OF THE OBJECTIVE FUNCTION (γ)

In our proposed method, we introduce the empirical loss function $\mathcal{L}_{\text{SL}}$ as follows:

$$\begin{aligned}\mathcal{L}_{\text{SL}} = \min_{\phi,\eta} \frac{1}{|X_{\text{train}}|} \Bigg(& \ell_{\text{CE}}(B_\eta(h_\phi(X_{\text{train}})), y) + \gamma_1 \|h_\phi(X_{\text{train}})\|_2 \\ & + \gamma_2 \exp\left(-\sum_{i=1}^{n} [\nabla h_\phi(X_{\text{train}})]_{ii}\right) \\ & + \gamma_3 \exp\left(\sum_{i=1}^{n}\left(-|[\nabla h_\phi(X_{\text{train}})]_{ii}| + \sum_{j\neq i} |[\nabla h_\phi(X_{\text{train}})]_{ij}|\right)\right)\Bigg) \end{aligned} \tag{1}$$

Here, γ_1 controls the regularization term $\|h_\phi(X_{\text{train}})\|_2$, which encourages the system's state to remain near the equilibrium point. This term helps mitigate the effect of perturbations by ensuring that trajectories stay close to the equilibrium. The hyperparameters γ_2 and γ_3 weight the exponential terms designed to enforce Lyapunov stability conditions by ensuring that the Jacobian matrix satisfies strict diagonal dominance, as per Theorem 3 in the paper. Specifically, γ_2 weights the term

$\exp\left(-\sum_{i=1}^{n}[\nabla h_\phi(X_{\text{train}})]_{ii}\right)$, encouraging the diagonal entries of the Jacobian to be negative with large magnitudes. γ_3 weights the term

$$\exp\left(\sum_{i=1}^{n}\left(-|[\nabla h_\phi(X_{\text{train}})]_{ii}| + \sum_{j\neq i}|[\nabla h_\phi(X_{\text{train}})]_{ij}|\right)\right),$$

promoting strict diagonal dominance by penalizing large off-diagonal entries relative to the diagonal entries.

We set $\gamma_1 = 1$ to balance the influence of the regularization term with the primary cross-entropy loss. This ensures sufficient regularization without overwhelming the classification objective. For γ_2 and γ_3, we choose $\gamma_2 = \gamma_3 = 0.05$. Preliminary experiments indicate that these values effectively enforce stability without causing numerical issues or hindering optimization.

We will now discuss why we select γ_2 and γ_3 equally:

- **Balanced Contribution**: Both regularization terms serve complementary purposes in enforcing the stability conditions. Equal weighting ensures that neither term dominates, maintaining a balanced emphasis on both negative diagonal dominance and strict diagonal dominance.
- **Simplified Hyperparameter Tuning**: Setting γ_2 and γ_3 equal reduces the hyperparameter search space, simplifying the tuning process without sacrificing performance.
- **Empirical Validation**: Experiments showed that equal values yield robust performance, and deviating from this balance did not provide significant benefits.

Regarding the effect of γ_1:

- **Low Values** ($\gamma_1 < 1$): Reduces the emphasis on keeping the state near the equilibrium, making the system more susceptible to perturbations.
- **High Values** ($\gamma_1 > 1$): Overly constrains the state to the equilibrium point, potentially limiting the model's capacity to learn discriminative features.
- **Chosen Value** ($\gamma_1 = 1$): Offers a good balance, ensuring sufficient regularization without compromising learning.

Similarly, the effect of γ_2 and γ_3 is as follows:

- **Low Values** (< 0.05): Diminish the impact of the stability constraints, reducing robustness.
- **High Values** (> 0.05): Overemphasize the stability terms, potentially hindering the optimization of the primary classification loss.
- **Equal Values** ($\gamma_2 = \gamma_3$): Ensure balanced enforcement of both stability conditions, leading to optimal performance.

Our stability framework draws inspiration from prior works on deep equilibrium-based models (8; 9; 10; 11; 12; 13; 14; 15), which proposed similar regularization and hyperparameter tuning techniques. To evaluate the robustness of AROS with respect to these hyperparameters, we conducted extensive ablation studies, holding all components constant while varying the values of γ_1, γ_2, and γ_3. These experiments demonstrate that AROS consistently performs well across a broad range of hyperparameter values, including extreme cases (e.g., $\gamma_1 = 2$). The results of this analysis, presented in Table 13a, confirm the robust performance of AROS under varying hyperparameter configurations.

A3.4 Time Complexity

The computational complexity of our model is reported in Table 14a.

A3.5 Discussion on the Fake Generation Strategy

It has been shown that utilizing auxiliary realistic OOD samples is generally effective for improving OOD detection performance. However, this strategy comes with several challenges, as discussed before.

Table 13a: An ablation study on the hyperparameters γ_3 and γ_2 and γ_1. Results are reported under clean and PGD$^{1000}(l_\infty)$ evaluations, measured by AUROC (%). Each table cell presents results in the 'Clean/PGD^{1000}' format.

Hyperparameter	CIFAR-10			CIFAR-100			ImageNet-1k		
	CIFAR-100	**SVHN**	**LSUN**	**CIFAR-10**	**SVHN**	**LSUN**	**Texture**	**iNaturalist**	**LSUN**
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.05$ (default)	88.2/80.1	93.0/86.4	90.6/ 82.4	74.3/67.0	81.5/70.6	74.3/68.1	78.3/ 69.2	84.6/75.3	79.4/69.0
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.025$	86.4/78.2	90.4/86.3	88.6/80.4	74.9/64.1	78.8/68.7	72.8/68.3	79.2/69.7	83.3/76.0	78.8/67.5
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.075$	87.5/80.3	92.1/87.2	88.9/81.6	72.7/67.7	80.8/68.7	73.2/68.8	76.4/69.9	83.4/73.4	80.2/69.9
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.1$	86.9/78.6	93.6/82.1	85.8/78.2	70.0/65.8	80.6/66.2	72.6/67.3	79.3/69.6	82.8/74.0	78.9/67.2
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.25$	86.2/78.7	94.3/82.3	87.1/77.9	70.2/65.2	81.3/65.8	73.9/66.7	78.6/70.1	82.8/74.1	78.9/68.2
$\gamma_1 = 1, \gamma_2 = \gamma_3 = 0.5$	85.2/76.9	93.6/81.5	85.0/76.0	69.7/64.2	78.1/64.4	72.6/65.2	77.0/66.7	83.6/74.4	78.2/67.3
$\gamma_1 = \gamma_2 = \gamma_3 = 1$	84.4/75.3	90.5/80.5	81.5/73.4	69.8/62.2	77.5/65.3	70.1/64.4	73.5/67.3	82.7/70.9	76.7/65.4
$\gamma_1 = \gamma_2 = \gamma_3 = 0.25$	85.4/77.4	94.6/82.9	86.3/76.9	69.9/64.4	79.9/65.1	74.4/65.3	77.2/68.6	85.4/76.0	79.2/66.8
$\gamma_1 = 0.5, \gamma_2 = \gamma_3 = 0.05$	86.0/78.6	92.7/82.5	86.3/77.7	69.7/66.1	81.2/65.5	70.9/67.1	78.8/68.6	83.0/73.7	78.8/67.1
$\gamma_1 = 1, \gamma_2 = 0.1, \gamma_3 = 0.05$	84.0/76.3	91.5/81.5	81.7/76.4	70.1/63.4	76.2/63.8	69.8/67.7	77.4/69.5	83.2/71.0	78.5/64.8
$\gamma_1 = 1, \gamma_2 = 0.05, \gamma_3 = 0.1$	83.5/77.5	90.2/78.0	81.4/75.1	67.2/63.5	76.7/66.2	67.5/60.4	74.3/70.0	83.6/74.1	74.2/61.7
$\gamma_1 = 2, \gamma_2 = 0.5, \gamma_3 = 0.5$	80.4/74.0	93.5/80.7	81.8/70.9	65.2/66.0	77.3/60.0	65.0/60.1	78.4/63.3	80.3/70.6	71.6/59.5

Table 14a: Time Complexity of Model Steps for CIFAR-10, CIFAR-100, and ImageNet-1k, measured on a NVIDIA RTX A5000 GPU (on a workstations running Ubuntu 20.04, Intel Core i9-10900X: 10 cores, 3.70 GHz, 19.25 MB cache; within Docker).

Step	**CIFAR-10**	**CIFAR-100**	**ImageNet-1k**
Step 1: Adversarial Training of Classifier	15 hours	15 hours	180 hours
Step 2: Crafting Fake OOD Data	7 hours	7 hours	150 hours
Step 3: Training with Stability Loss ($\mathcal{L}_{\text{SL}}$)	8 hours	8 hours	100 hours

First, in certain scenarios, access to an external realistic OOD dataset may not be feasible, and acquiring such data can be challenging. Even when a suitable dataset is available, it must be processed to remove ID concepts to prevent the detector from being misled. This preprocessing step is both time-consuming and computationally expensive. Additionally, studies highlight a potential risk of bias being introduced into the detector when trained on specific auxiliary datasets. Such reliance on a particular realistic dataset may undermine the detector's ability to generalize effectively to diverse OOD samples. These issues become even more pronounced in adversarial training setups, where the complexity of the required data is significantly higher. Motivated by these challenges, this study proposes an alternative strategy that does not depend on the availability of an auxiliary OOD dataset. Notably, our approach is flexible and can incorporate auxiliary OOD datasets as additional information if they are available. To validate this, we conducted an experiment assuming access to a realistic OOD dataset (i.e., Food-101). In this scenario, we computed embeddings of the real OOD samples and used them alongside crafted fake OOD embeddings during training. The results, presented in Table 4 (Setup A), demonstrate improved performance compared to using fake OOD embeddings alone. Furthermore, related studies have shown that in adversarial training, using samples near the decision boundary of the distribution improves robustness by encouraging compact representations. This boundary modeling is critical for enhancing the model's robustness, especially against adversarial attacks that exploit vulnerabilities near the decision boundary. In light of this, our approach shifts focus from generating "realistic" OOD data to estimating low-likelihood regions of the in-distribution. We generate fake "near" OOD data that is close to the ID boundary, which is particularly beneficial for adversarial robustness.

For better intuition regarding usefulness of auxiliary near OOD samples here We will provide a simple example that highlights the effectiveness of near-distribution crafted OOD samples in the adversarial setup. We assume that the feature space is one-dimensional, i.e. $\mathbb{R}$, and the ID class is sampled according to a uniform distribution $U(0, a-\epsilon)$, with $a > 0$, and $\epsilon < a$. We assume that the OOD class is separable with a safety margin of 2ϵ from the ID class to allow a perfectly accurate OOD detector under input perturbations of at most ϵ at inference. For instance, we let $U(a+\epsilon, b)$ be the feature distribution under the OOD class. The goal is to leverage crafted fake OOD samples to find a robust OOD detector under the ℓ_2 bounded perturbations of norm ϵ. We assume that the crafted OOD samples data distribution is not perfectly aligned with the anomaly data, e.g. crafted OOD

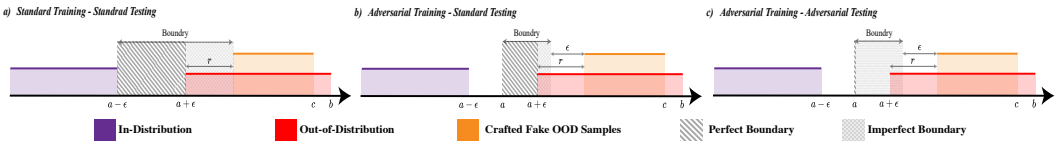

Figure 4: A depiction in a one-dimensional feature space where the crafted fake OOD samples form a subset of actual OOD data. The gray area represents feasible thresholds separating ID (purple) and fake OOD data (orange). r indicates the shift in exposed fake OOD samples from the real OOD samples. Bold gray lines represent perfect test AUROC thresholds. **Left**: In standard scenarios, perfect thresholds are abundant, even if the fake OOD samples are distant from the ID data. **Middle**: With adversarial training, the feasible thresholds decrease due to the maximum margin constraint, impacting the perfect thresholds. Large deviations in the crafted fake OOD data reduce the set of perfect thresholds. **Right**: For adversarial testing, the overlap between feasible and perfect thresholds narrows to point a, emphasizing the importance of near-OOD properties in adversarial contexts.

samples comes from $U(a+r, c)$, with $r \geq \epsilon$. It is evident that the optimal robust decision boundary under infinite training samples that separates the ID and crafted OOD samples would be a threshold k satisfying $a \leq k \leq a + r - \epsilon$. The test adversarial error rate to separate ID and OOD classes is $\frac{1}{2}.\mathbb{I}(k \geq a+\epsilon).\left(\frac{k-a-\epsilon}{b-a-\epsilon} + \frac{\min(k+\epsilon,b)-k}{b-a-\epsilon}\right) + \frac{1}{2}.\mathbb{I}(a < k < a+\epsilon)\frac{\min(k+\epsilon,b)-a-\epsilon}{b-a-\epsilon}$, assuming that the classes are equally probable a prior. It is obvious the adversarial error rate would be zero for $k = a$. But otherwise, if $k \geq a + \epsilon$ the classifier incurs classification error in two cases; in intervals $(a+\epsilon, k)$ (even without any attack), and $(k, \min(k+\epsilon, b))$ in which a perturbation of $-\epsilon$ would cause classification error. Also if $a < k < a + \epsilon$, classification error only happens at $(a+\epsilon, \min(k+\epsilon, b))$. Now, for the crafted OOD samples to be near-distribution, $r \to \epsilon$, which forces k to be a in the training, and makes the test adversarial error zero. Otherwise, the adversarial error is proportional to k, for k being distant from b. Therefore, in the worst case, if $k = a+r-\epsilon$, we get an adversarial error proportional to r. As a result, minimizing r, which makes the crafted OOD samples near-distribution, would be an effective strategy in making the adversarial error small. Refer to the figure 4 for further intuition and clarity.

To provide a more practical intuition about our crafted fake OOD samples, we present t-SNE visualizations of the embedding space. These visualizations demonstrate that the crafted fake data are positioned near the ID samples and effectively cover their boundary. Please refer to Figures 5 and 6.

To further demonstrate the superiority of our strategy for crafting fake OOD samples—a simple yet effective technique—we conducted an ablation study by replacing our proposed method with alternative approaches. Please refer to Table 15a for the results.

Table 15a: Ablation study on different fake OOD crafting strategies. Results are measured by AUROC (%). The perturbation budget ϵ is set to $\frac{8}{255}$ for low-resolution datasets and $\frac{4}{255}$ for high-resolution datasets. Table cells present results in the 'Clean/PGD^{1000}' format. We evaluate alternative OOD data synthesis strategies, such as random Gaussian and uniform noise, while keeping other components fixed.

Fake Crafting	CIFAR-10			CIFAR-100			ImageNet-1k		
Strategy	CIFAR-100	SVHN	LSUN	CIFAR-10	SVHN	LSUN	Texture	iNaturalist	LSUN
AROS	88.2/80.1	93.0/86.4	90.6/82.4	74.3/67.0	81.5/70.6	74.3/68.1	78.3/69.2	84.6/75.3	79.4/69.0
Random Gaussian Noise	85.3/76.5	89.4/78.1	82.5/77.1	70.5/61.3	74.4/62.5	71.5/64.6	76.1/67.4	81.3/72.7	75.8/67.3
Random Uniform Noise	81.2/74.3	87.1/79.5	84.8/75.4	65.6/59.6	73.8/63.5	65.8/60.1	70.0/63.8	78.8/67.3	71.6/62.8

A3.6 ORTHOGONAL BINARY LAYER

To demonstrate the superiority of the orthogonal binary layer used in our pipeline, we conducted an ablation study. In this study, we fixed all other components and replaced the orthogonal binary layer with a regular fully connected (FC) layer, then compared the results across different budget levels. The results of this comparison are presented in Table 16a. Additionally, we visualized the embedding space corresponding to the regular FC layer and the orthogonal binary layer in Figure 7.

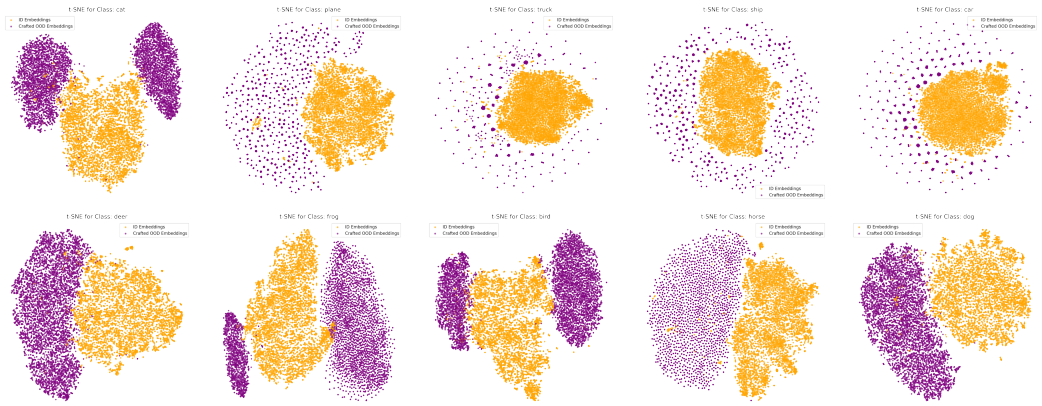

Figure 5: t-SNE visualization of CIFAR-10 embeddings and the corresponding crafted OOD embeddings for each class. Orange points represent the ID embeddings for each class, while purple points represent the synthetic OOD embeddings crafted using a GMM. The visualization highlights the separability between ID and OOD embeddings. The crafted embeddings are positioned near the boundaries of the ID concepts, emphasizing they are near OOD samples and they coverage the OOD space. The β hyperparameter used in this experiment is set to 0.001.

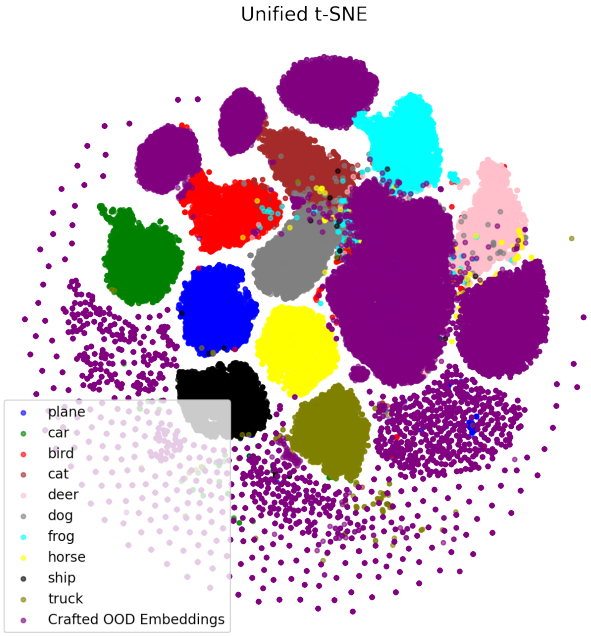

Figure 6: Unified t-SNE visualization of embeddings for all CIFAR-10 classes and their corresponding crafted OOD embeddings. Each color represents a specific CIFAR-10 class, while the purple points represent the synthetic OOD embeddings crafted using a GMM. The figure demonstrates the clustering of ID embeddings for each class and the distinct distribution of crafted OOD embeddings. The β hyperparameter used in this experiment is set to 0.01.

These visualizations highlight the enhanced separability provided by the orthogonal binary layer in both clean and adversarial scenarios.

The orthogonal binary layer B_η is designed to apply a transformation to the NODE output $h_\phi(z)$, where the weights W of the layer are constrained to be orthogonal ($W^T W = I$). This constraint encourages maximal separation between the equilibrium points of ID and OOD data by ensuring that

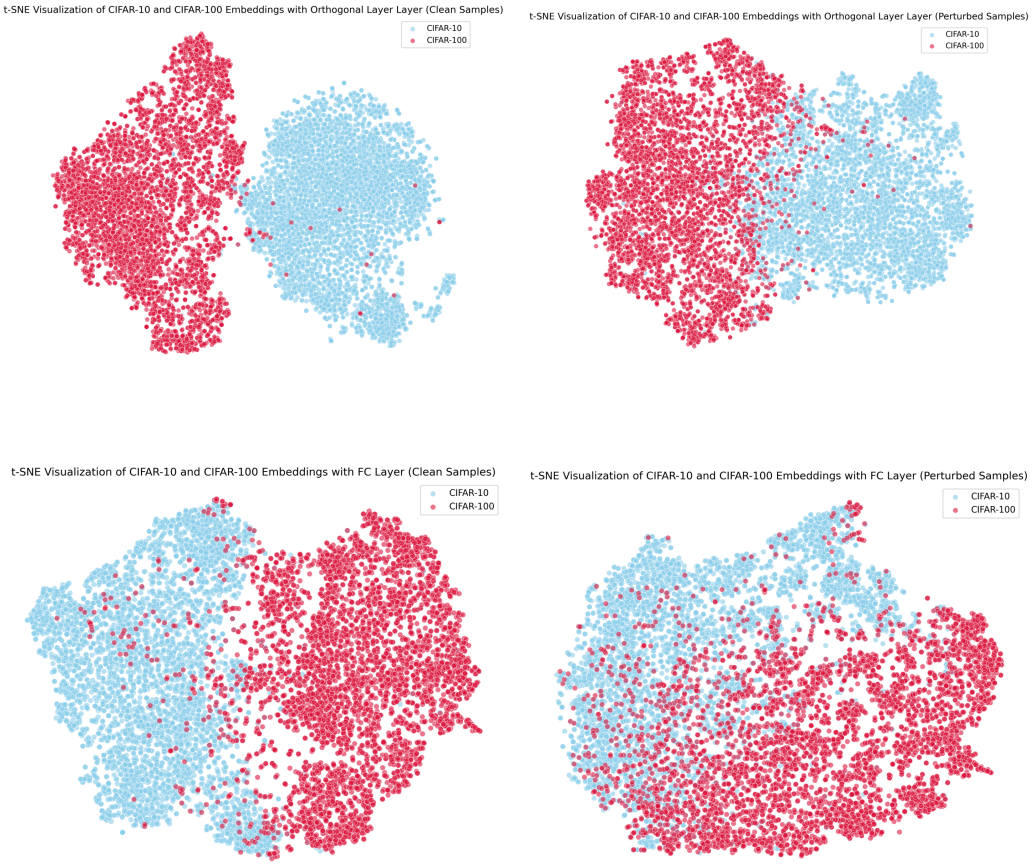

Figure 7: t-SNE visualizations of CIFAR-10 (ID) and CIFAR-100 (OOD) embeddings, illustrating the effect of an orthogonal binary layer compared to a regualr fully connected (FC) layer under clean and perturbed sample scenarios. Top row: Embeddings with the orthogonal binary layer for clean (left) and perturbed (right) samples, showing enhanced ID-OOD separation. Bottom row: Embeddings with the FC layer for clean (left) and perturbed (right) samples, demonstrating reduced separation in adversarial settings. These results highlight the orthogonal binary layer's effectiveness in preserving ID-OOD distinction, in both clean and adversarial conditions.

Table 16a: Comparison of performance between the orthogonal binary layer (Ortho.) and the regular fully connected (FC) layer across different datasets and perturbation levels (ϵ). The results demonstrate that the orthogonal binary layer consistently outperforms the regular FC layer in both clean ($\epsilon = \frac{0}{255}$) and adversarial scenarios, with varying levels of perturbation ($\epsilon = \frac{2}{255}, \frac{4}{255}, \frac{8}{255}$).

ϵ	Methods	CIFAR-10			CIFAR-100			ImageNet-1k		
		CIFAR-100	SVHN	LSUN	CIFAR-10	SVHN	LSUN	Texture	iNaturalist	LSUN
$\frac{0}{255}$ (Clean)	FC	85.6	88.2	87.5	66.9	78.4	71.5	75.4	79.8	76.1
	Ortho.	88.2	93.0	90.6	74.3	81.5	74.3	78.3	84.0	79.4
$\frac{2}{255}$	FC	82.0	84.9	84.1	63.6	75.8	68.6	73.2	76.9	72.2
	Ortho.	85.5	90.7	88.8	72.2	79.9	72.1	75.7	82.7	76.6
$\frac{4}{255}$	FC	75.2	78.1	80.3	57.9	71.6	63.5	68.2	72.9	66.7
	Ortho.	83.1	87.5	85.2	69.5	74.5	70.4	71.6	78.3	68.0
$\frac{8}{255}$	FC	67.3	74.6	79.5	57.1	63.3	62.4	60.7	70.2	65.1
	Ortho.	80.1	86.4	82.4	67.0	70.6	68.1	69.2	75.3	69.0

the learned representations preserve distinct directions in the feature space. Specifically, the layer operates as:

$$B_\eta(z) = Wz + b, \text{ subject to } W^T W = I,$$

where W represents the weight matrix, b is the bias term, and the orthogonality constraint is enforced during training using a regularization term.

To clarify its role in L_{SL}, we now explicitly annotate B_η as the mapping responsible for projecting the NODE output $h_\phi(z)$ into a binary classification space (ID vs. OOD).

The L_{SL} with explicit reference to W and b would be as follows:

$$\begin{aligned} L_{SL} = \min_{\phi,w} \frac{1}{|X_{\text{train}}|} \Bigg(& \ell_{\text{CE}}((Wh_\phi(X_{\text{train}}) + b), y) + \gamma_1 \|\|h_\phi(X_{\text{train}})\|\|_2 \\ & + \gamma_2 \exp\Bigg(- \sum_{i=1}^{n} [\nabla h_\phi(X_{\text{train}})]_{ii} \Bigg) \\ & + \gamma_3 \exp\Bigg(\sum_{i=1}^{n} (-|[\nabla h_\phi(X_{\text{train}})]_{ii}| \\ & + \sum_{j \neq i} |[\nabla h_\phi(X_{\text{train}})]_{ij}|) \Bigg) \Bigg) \end{aligned}$$

where the orthogonality constraint $W^T W = I$ is enforced via regularization during optimization.

A4 Details of Evaluation and Experimental Setup

OOD detection can be framed as a binary classification task, where training samples are confined to a single set (the ID data), and during testing, input samples from OOD set must be identified. A key challenge is that OOD data is not as clearly defined as ID data; any semantic content absent from the ID distribution is considered OOD. The primary objective in OOD detection is to develop a pipeline capable of assigning meaningful OOD scores to input samples, where a higher score suggests that the model perceives the input as having a greater likelihood of being OOD. An optimal OOD detector would produce score distributions for OOD and ID samples that are fully separated, with no overlap, ensuring clear differentiation between the two.

Formally, the OOD detection decision can be performed as follows:

$$G_\lambda(\text{x}) = \begin{cases} \text{ID} & \text{if } S_{\mathcal{F}}(x) \leq \lambda \\ \text{OOD} & \text{if } S_{\mathcal{F}}(x) > \lambda \end{cases},$$

where λ is a threshold parameter.

Thus, the scoring function $S_{\mathcal{F}}$ is central to the performance of the OOD detector. In the setup of adversarial evaluation, PGD and other attack methods, we adversarially target $S_{\mathcal{F}}$ and perturb the inputs as described in Section 2. Intuitively, these attacks aim to shift ID samples closer to OOD and vice versa. This approach is fair and consistent with existing OOD detection frameworks, as all such methods define an OOD score function. Furthermore, this adversarial strategy has been previously explored in related work. In the following, we will provide details on AutoAttack and adaptive AutoAttack in our evaluation.

AutoAttack. AutoAttack is an ensemble of six attack methods: APGD with Cross-Entropy loss, APGD with Difference of Logits Ratio (DLR) loss, APGDT, FAB (121), multi-targeted FAB (122), and Square Attack (123). However, the DLR loss-based attacks assume that the target model is a classifier trained on more than two classes. In OOD detection, the problem is more akin to binary classification, as we are only distinguishing between OOD and ID classes. Therefore, we excluded

these specific DLR-based attacks in our adaptation of AutoAttack and instead used an ensemble of the remaining attacks.

Adaptive AA. We also employed Adaptive AutoAttack (AA) for evaluation, an attack framework designed to efficiently and reliably approximate the lower bound of a model's robustness. Adaptive AA integrates two main strategies: Adaptive Direction Initialization to generate better starting points for adversarial attacks, and Online Statistics-Based Discarding to prioritize attacking easier-to-perturb images. Unlike standard AutoAttack, Adaptive AA adapts its attack directions based on the model's specific defense properties and dynamically allocates iterations to improve the efficiency of successful attacks. For the perturbation budget, we used a similar budget to that considered for PGD, and for other hyperparameters, we used their default values. For the implementation of adversarial attacks, we utilized the `Torchattacks` library in PyTorch.

A4.1 Details of the Reported Results

To report and evaluate previous works, we reproduced their results using the official GitHub repositories. For the clean evaluation setup, if our reproduced results differed from the originally reported values, we reported the higher value. If no results were available, we presented our reproduced results. A similar approach was taken for adversarial evaluation: if their evaluation details and considered datasets matched ours, we used their reported results. However, if the attacks details in their evaluation differed or if they did not consider our datasets or benchmarks, we reported our reproduced results.

A4.2 Metrics

Our main results are reported using AUROC for comparison across methods, as it is more commonly utilized in the detection literature. However, in Table 4b, we also compare our method's performance using other metrics, including AUPR and FPR95. Below, we provide an explanation of each of these metrics.

AUROC. The Area Under the Receiver Operating Characteristic Curve (AUROC) is a metric that evaluates classification performance by measuring the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) across various threshold settings. TPR, also known as recall or sensitivity, measures the proportion of actual positives correctly identified, while FPR measures the proportion of actual negatives incorrectly classified as positives. The AUROC, which ranges from 0.5 (random classifier) to 1.0 (perfect classifier), provides an aggregate performance measure.

AUPR. The Area Under the Precision-Recall Curve (AUPR) is particularly useful for imbalanced datasets where the positive class is rare. The AUPR captures the trade-off between precision (the proportion of predicted positives that are actual positives) and recall. A high AUPR indicates a model with both high precision and recall, providing valuable insights in scenarios where the dataset is skewed.

FPR95. False Positive Rate at 95% True Positive Rate (FPR95), which assesses the model's ability to correctly identify ID samples while rejecting OOD samples. Specifically, FPR95 is the false positive rate when the true positive rate is set to 95%, indicating how often OOD samples are misclassified as ID. Lower values of FPR95 indicate better OOD detection capabilities, as the model more accurately differentiates between ID and OOD samples. Together, these metrics offer a comprehensive understanding of model performance and robustness in classification and OOD detection tasks.

A5 Additional Details about the Datasets

In our experiments, we considered several datasets. Our perturbation budget is determined based on the image size of the ID training set. It is set to $\frac{4}{255}$ for scenarios where the ID training set consists of high-resolution images (e.g., ImageNet-1K), and $\frac{8}{255}$ for high-resolution datasets (e.g., CIFAR-10 and CIFAR-100). In all experiments, the ID and OOD datasets contain disjoint semantic classes. If there is an overlap, we exclude those semantics from the OOD test set. In all experiments, the ID and OOD datasets contain disjoint semantic classes. If there is an overlap, we exclude those semantics from the OOD test set. In the following, we provide a brief explanation of the datasets used.

CIFAR-10 and CIFAR-100. are benchmark datasets commonly used for image classification tasks. CIFAR-10 consists of 60,000 color images of size 32×32 pixels across 10 classes, with 6,000 images per class. CIFAR-100 is similar but contains 100 classes with 600 images each, providing a more fine-grained classification challenge.

ImageNet-1k. ImageNet-1K contains 1,281,167 training images, 50,000 validation images and 100,000 test images. This dataset was compiled to facilitate research in computer vision by providing a vast range of images to develop and test algorithms, particularly in the areas of object recognition, detection, and classification.

Texture Dataset. Texture Dataset is designed for studying texture recognition in natural images. It comprises diverse textures captured in the wild, enabling research on classifying and describing textures under varying conditions.

Street View House Numbers (SVHN). SVHN is a real-world image dataset for developing machine learning and object recognition algorithms with minimal data preprocessing. It contains over 600,000 digit images obtained from house numbers in Google Street View images.

iNaturalist. The iNaturalist Species Classification and Detection Dataset aims to address real-world challenges in computer vision by focusing on large-scale, fine-grained classification and detection. This dataset, consisting of 859,000 images from over 5,000 different species of plants and animals, is noted for its high class imbalance and the visual similarity of species within its collection. The images are sourced globally, contributed by a diverse community through the iNaturalist platform, and verified by multiple citizen scientists.

Places365. Places365 is a scene-centric dataset containing over 10 million images spanning 365 scene categories. It is designed for scene recognition tasks and aids in understanding contextual information in images.

Large-scale Scene Understanding (LSUN). The Large-scale Scene Understanding (LSUN) challenge is designed to set a new standard for large-scale scene classification and comprehension. It features a classification dataset that includes 10 scene categories, such as dining rooms, bedrooms, conference rooms, and outdoor churches, among others. Each category in the training dataset comprises an extensive range of images, from approximately 120,000 to 3,000,000 images. The dataset also provides 300 images per category for validation and 1,000 images per category for testing.

iSUN. The iSUN dataset is a large-scale eye tracking dataset that leverages natural scene images from the SUN database. It was specifically developed to address the limitations of small, in-lab eye tracking datasets by enabling large-scale data collection through a crowdsourced, webcam-based eye tracking system deployed on Amazon Mechanical Turk.

MNIST. The MNIST dataset (Modified National Institute of Standards and Technology dataset) is a large collection of handwritten digits that is widely used for training and testing in the field of machine learning. It contains 70,000 images of handwritten digits from 0 to 9, split into a training set of 60,000 images and a test set of 10,000 images. Each image is a 28x28 pixel grayscale image.

Fashion-MNIST (FMNIST). FMNIST dataset is a collection of article images designed to serve as a more challenging replacement for the traditional MNIST dataset. It consists of 70,000 grayscale images divided into 60,000 training samples and 10,000 test samples, each image having a resolution of 28x28 pixels. The dataset contains 10 different categories of fashion items such as T-shirts/tops, trousers, pullovers, dresses, coats, sandals, shirts, sneakers, bags, and ankle boots.

Imagenette. Imagenette is a subset of ImageNet consisting of ten easily classified classes. It was released to encourage research on smaller datasets that require less computational resources, facilitating experimentation and algorithm development.

CIFAR-10-C and CIFAR-100-C. CIFAR-10-C and CIFAR-100-C are corrupted versions of CIFAR-10 and CIFAR-100, respectively. They include various common corruptions and perturbations such as noise, blur, and weather effects, used to evaluate the robustness of image classification models against real-world imperfections.

ADNI Neuroimaging Dataset. The Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset is a large-scale collection of neuroimaging data aimed at tracking the progression of Alzheimer's disease. In our study, we categorize the dataset into six classes based on cognitive status and

disease progression. The classes CN (Cognitively Normal) and SMC (Subjective Memory Concerns) are designated as ID, representing individuals without significant neurodegenerative conditions or with only minor memory concerns. The other four classes—AD (Alzheimer's Disease), MCI (Mild Cognitive Impairment), EMCI (Early Mild Cognitive Impairment), and LMCI (Late Mild Cognitive Impairment)—are treated as OOD, encompassing various stages of cognitive decline related to Alzheimer's. The dataset contains 6,000 records for each class, providing a comprehensive representation of the spectrum of cognitive health and Alzheimer's disease. A summary of each class is as follows:

- **AD**: Patients diagnosed with Alzheimer's disease, showing significant cognitive decline.
- **CN**: Individuals with normal cognitive functioning, serving as a control group.
- **MCI**: A stage of cognitive decline that lies between normal aging and more advanced impairment, often preceding Alzheimer's disease.
- **EMCI**: Patients with early signs of mild cognitive impairment, indicating the initial onset of cognitive issues.
- **LMCI**: Patients with symptoms indicative of more advanced mild cognitive impairment, closer to Alzheimer's in severity.
- **SMC**: Individuals reporting memory concerns but performing normally on cognitive assessments.

We used the middle slice of MRI scans from ADNI phases 1, 2, and 3. We should note that the ADNI dataset was the only one we split into ID and OOD using the strategy mentioned earlier. For the other datasets in our OSR experiments, we randomly divided each dataset into ID and OOD at a ratio of 0.6 and 0.4, respectively.

A5.1 Details on Dataset Corruptions

Corruptions: The first type of corruption is Gaussian noise, which commonly appears in low-light conditions. Shot noise, also known as Poisson noise, results from the discrete nature of light and contributes to electronic noise. Impulse noise, similar to salt-and-pepper noise but in color, often arises due to bit errors. Defocus blur happens when an image is out of focus. Frosted glass blur is caused by the appearance of "frosted glass" on windows or panels. Motion blur occurs due to rapid camera movement, while zoom blur happens when the camera moves swiftly towards an object. Snow is an obstructive form of precipitation, and frost occurs when lenses or windows accumulate ice crystals. Fog obscures objects and is often simulated using the diamond-square algorithm. Brightness is affected by the intensity of daylight, and contrast levels change based on lighting conditions and the object's color. Elastic transformations distort small image regions by stretching or contracting them. Pixelation results from upscaling low-resolution images. Lastly, JPEG compression, a lossy format, introduces noticeable compression artifacts.

A6 Limitations and Future Research

We considered classification image-based benchmarks, that while difficult, do not cover the full breadth of real-world attacks in situations such as more complex open-set images or video, i.e., time-series data. Future work should test our proposed method in video streaming data (time-series), as inherently we leverage stability properties in dynamical systems that could be very attractive for such settings. Namely, because video frames are often highly correlated, one can even leverage the prior stability points in time in order to make better future predictions. This could also lower the compute time.

Moreover, moving to tasks beyond classification could be very attractive. Concretely, one example is pose estimation. This is a keypoint detection task that can often be cast as a panoptic segmentation task, where each keypoint needs to be identified in the image and grouped appropriately (124; 125). By using AROS, one could find OOD poses could not only improve data quality, but alert users to wrongly annotated data. Another example is in brain decoding where video frame or scene classification is critical, and diffusion models are becoming an attractive way to leverage generative models (126). Given AROS ability in OOD detection, this could be smartly used to correct wrong

predictions. In summary, adapting AROS to these other data domains could further extend its applicability.