Stop Diverse OOD Attacks: Knowledge Ensemble for Reliable Defense

Zhenbo Shi[1,2,3,4], **Xiaoman Liu**[1,2], **Yuxuan Zhang**[1], **Shuchang Wang**[1], **Rui Shu**[1], **Zhidong Yu**[1,4,*], **Wei Yang**[1,2,4,*], **Liusheng Huang**[1,2]

[1] School of Computer Science and Technology, University of Science and Technology of China, Hefei, China
[2] Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou, China
[3] Laboratory for Advanced Computing and Intelligence Engineering, Wuxi, China
[4] Hefei National Laboratory, University of Science and Technology of China, Hefei, China
[*] yuzd@mail.ustc.edu.cn, qubit@ustc.edu.cn

Abstract

Enhancing defense through model ensemble is an emerging trend, where the challenge lies in how to use ensemble knowledge to counter Out-of-Distribution (OOD) attacks. In this paper, we propose the Reliable Defense Ensemble (REE) to address this issue. REE optimizes the ensemble knowledge of models through aggregation and enhances multidimensional robust performance through collaboration. It employs the Dynamic Synergy Amplification for weight allocation and strategy adjustment. Furthermore, we design a new Kernel Anomaly Smoothing Detection Module, which detects anomalous attacks using a smoothing feature function based on Gaussian kernel mean embedding and a multi-layer feedback structure. Particularly, we build a framework that uses reinforcement learning to iteratively fine-tune the parameters of inter-model communication and consensus. Extensive experimental results show that REE outperforms current state-of-the-art methods by a large margin in defending against OOD attacks.

Introduction

Background and Related Work. Deep learning is developing rapidly (Ge, Fu, and Zha 2022; Ge et al. 2024; Shi et al. 2022), but the emergence of adversarial attacks poses huge challenges. As Szegedy et al. (2013) proved that even imperceptible perturbations in the input can mislead a model. Papernot et al. (2016) further exploits the adversarial saliency between input features and output values, which cause misclassification with slight modifications. In the study of defense strategies, Goodfellow, Shlens, and Szegedy (2014) first propose adversarial training and put forward new ideas for model robustness. However, as Tramèr et al. (2017) highlight, the adaptive and continuously evolving nature of attacks necessitates more sophisticated and resilient defense strategies. Understanding the vulnerable parts of models and designing new defense strategies is critical to maintaining the integrity and trustworthiness of AI systems.

Distribution anomalies pose a strong robustness challenge to the model. While poisoning attacks typically affect the training phase, and our work centers on inference-stage defenses, research on poisoning defense has indeed provided valuable insights for our approach. BagFlip (Zhang, Albarghouthi, and D'Antoni 2022) combines bagging and noise technology to combat trigger-free and backdoor attacks, and uses the Neyman–Pearson lemma to calculate the authentication radius. In response to the problem that NLP models are vulnerable to backdoor attacks, Shen et al. (2022) proposed an optimization method to reverse the backdoor trigger, as well as a dynamic temperature scaling and rollback mechanism. Random Transformation defense (RT) uses a differentiable method to resist adversarial attacks, and utilizes transfer attacks and the random optimization algorithm (Sitawarin, Golan-Strieb, and Wagner 2022). Ho and Vasconcelos (2022) proposed an image classification adversarial defense DISCO based on local implicit functions, which performs projection and conditional modeling on local manifolds. In addition, current integrated defense approaches remain vulnerable to new, cleverly crafted attack vectors that can dynamically deceive defense strategies. For example, these works (Seligmann et al. 2024; Croce et al. 2022) have conducted extensive and in-depth research on ensemble models.

Adaptive defense is considered an important step to combat the limitations of static defense, Croce et al. (2022) evaluated nine different adaptive defense methods, showing that their performance relative to static defense often does not significantly improve, and sometimes even decreases the robustness of the model. Model stealing attack gradient redirection (GRAD2) (Mazeika, Li, and Forsyth 2022) can maintain high security while maintaining service quality for normal users. In reinforcement learning, Bharti et al. (2022) proposed provable defense methods against backdoor strategies, using the concept of safe subspaces to eliminate the impact of backdoor triggers. Considering the impact of different normalization layers on attack migration performance, Dong et al. (2022) designed Random Normalization Aggregation (RNA) based on this. This method forms a huge random space by aggregating multiple normalization methods to increase the difficulty of being attacked in a white-box setting.

LeadFL (Zhu, Roos, and Chen 2023) focuses on solving the problem of resisting model poisoning attacks in federated learning, and introduces regularization on the client side to face sudden and highly changing malicious attack patterns. This mechanism can be used in conjunction with any existing server-side defense strategy, and was compared with algorithms such as SparseFed, Multi-Krum, and Bulyan un-

[*]Corresponding Authors.

der different data distributions (IID and non-IID) and attack modes. NNSplitter (Zhou et al. 2023b) provides defense for DNN models through automated weight obfuscation. Zhou et al. (2023a) proposed the adversarial defense strategy PAD from the phase perspective of the image. This strategy improves the robustness of the model in the face of attacks through phase-level adversarial training and amplitude-based preprocessing operations. Reconstructive neuron pruning emphasizes asymmetric unlearning and recovery methods that can effectively expose and prune backdoor neurons with minimal clean data (Li et al. 2023).

Our Aim and Contributions. This paper aims to address the challenge of how to use ensemble knowledge to counter OOD attacks. To achieve this, we propose the Reliable Defense Ensemble (REE) to tackle this issue. Our approach focuses on two innovations within REE: Dynamic Synergy Amplification (DSA) and the Kernel Anomaly Smoothing Detection Module (KASD). DSA optimizes the ensemble knowledge of models and enhances multidimensional robust performance through collaborative knowledge. It uses iterative reinforcement for weight allocation and strategy adjustment, strengthening the overall resilience of the model group. KASD detects anomalous attacks by using a smoothing feature function based on Gaussian kernel mean embedding and a multi-layer feedback structure. Particularly, we build a framework that uses reinforcement learning to iteratively fine-tune the parameters for communication and consensus among models.

To summarize, we make the following contributions:

- We propose the Reliable Defense Ensemble (REE), which uses aggregation and collaboration to enhance the robustness of models and uses these methods to address the challenge of using ensemble knowledge to counter OOD attacks.
- We design the Dynamic Synergy Amplification (DSA), which implements weight allocation and strategy adjustment through a quantified threat index to avoid the problem of individual models being unable to effectively share critical knowledge.
- We put forward the Kernel Anomaly Smoothing Detection Module (KASD), which uses a smoothing feature function based on Gaussian mean embedding for anomaly detection and employs a multi-layer feedback structure to identify new attacks.

Defense Principles

Theoretical Framework. Let $\mathbf{M}$ represent the feature space of a single model and $\mathbf{X}$ denote the input space. For a given model $\mathbf{M} \in \mathcal{M}$ ($\mathcal{M} = \{\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_n\}$) and an input $\mathbf{X} \in \mathcal{X}$, an adversarial vulnerability $\nu(\mathbf{M}, \mathbf{X})$ is defined as one that can generate adversarial examples to mislead $\mathbf{M}$. $\rho(\mathcal{M})$ represents the degree of robustness of the model group to attacks. The synergy score $\boldsymbol{S}$ of a model group $\mathcal{M}$ is defined as a function of the robustness measure and interaction dynamics of each model, i.e. $\boldsymbol{S}(\mathcal{M}) = \Phi(\{\rho(\mathbf{M}_i)\}_{i=1}^n, \boldsymbol{\mathcal{I}})$, where Φ is an aggregation function encapsulating the collective defense strategy, and $\boldsymbol{\mathcal{I}}$ represents the information interaction between models. Furthermore, Dynamic Synergy Amplification (DSA) is applied to the model group $\mathcal{M}$ as a series of operations $\mathbf{D}$ to dynamically enhance its collective resilience. For a given input x, the output of the model under DSA is given by $\mathbf{D}(\mathbf{M}, \mathbf{X}) = \sum_{i=1}^n p_i \mathbf{M}_i(\mathbf{X})$, where p_i is the dynamic weight. It reflects the contribution of model $\mathbf{M}_i$ based on the current threat situation and network collaboration. In the Kernel Anomaly Smoothing Detection Module (KASD), $\mathcal{F}(\mathcal{H})$ functions as a strategic prediction map for defense, using a smooth feature function based on Gaussian kernel mean embedding, all informed by historical data $\mathcal{H}$.

Architectural Overview. The core of REE is a collection of models (as shown in Fig. 1), each of which has a specific Elastic Morphism Mapping $\mathcal{R}(\cdot)$ that maps the model response to the adversarial feature space. The process is formalized as follows, where $\mathcal{R}(\mathbf{M}_i)$ represents the elastic embedding of model $\mathbf{M}_i$ in the network, and the form of the comprehensive model elasticity $\mathbf{R}_{\text{net}}$ is as $\mathbf{R}_{\text{net}} = \mathbf{D}(\mathbf{r}(\mathbf{M}_i), \mathbf{X})$. This design confronts the non-differentiable nature of adversarial landscapes, acknowledging the resilience of individual models, as well as the elasticity of entire ensembles, all existing within a complex, often discrete space. The convex hull on the feature space is defined as $\Omega = \sum_{i=1}^n \boldsymbol{\alpha}_i \cdot \mathcal{R}(\mathbf{M}_i)$, where $\sum_{i=1}^n \boldsymbol{\alpha}_i = 1, \boldsymbol{\alpha}_i \geq 0$. Therefore, any collaborative strategy $\boldsymbol{S}_t$ at time t can be represented by a convex hull: $\Omega(\boldsymbol{S}_t) = \sum_{i=1}^n \boldsymbol{\alpha}_{i,t} \cdot \mathcal{R}(\mathbf{M}_i)$ where $\sum_{i=1}^n \boldsymbol{\alpha}_{i,t} = 1, \boldsymbol{\alpha}_{i,t} \geq 0$. Following the method in (Chen, Ding, and Carin 2015; Dong et al. 2021), we use SoftMax transformation on the weight $\boldsymbol{\alpha}_{i,t}$ to ensure that the dynamic policy obeys the convex combination constraint.

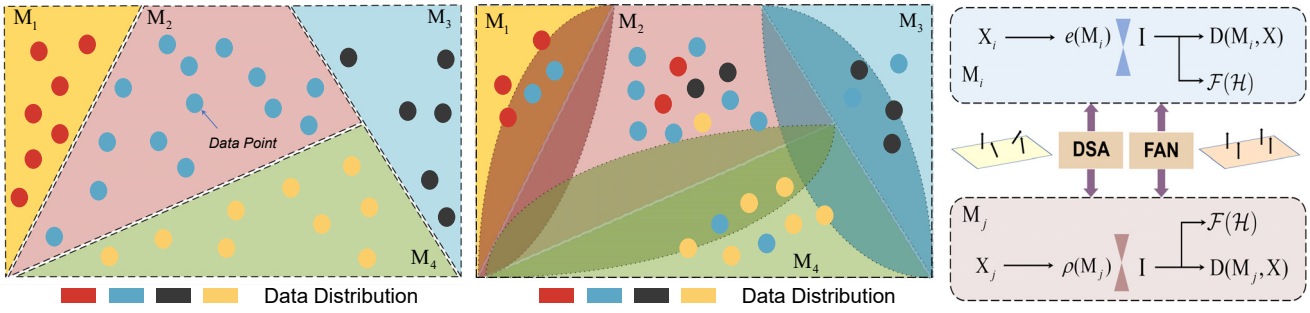
Approach

Dynamic Synergy Amplification

Synergy Scoring. We treat the model group as a whole, and the Synergy Scoring is crucial to deal with the multifaceted entities of the attack in Dynamic Synergy Amplification (DSA). The probability that any model $\mathbf{M}_i$ is the best fit for a given challenge is uniformly assumed to be $\mathbf{V}(\mathbf{M}_i) = 1/N$, where N is the total number of models.

Given the complexity of adversaries, estimates of individual model responses are insufficient. Therefore, we propose a method similar to the high-dimensional estimators (Ziegel 2003; Zou and Hastie 2005; Cheng, Diakonikolas, and Ge 2019; Diakonikolas et al. 2019) in robust statistical theory. We define $n_i(\mathrm{T})$ as the participation count of model i up to time T. DSA operates by continuously analyzing incoming threats in real-time and dynamically adjusting the distribution of weights across the model ensemble. This adjustment is based on a quantified threat index, which measures the severity and type of incoming adversarial attacks. Each model's weight is recalibrated using a softmax function over the threat index, ensuring that models better suited to countering the current attack have increased influence in the ensemble's decision-making process. The synergy vector $\hat{\boldsymbol{\varphi}}_i$ of all models is calculated as follows:

$$\hat{\boldsymbol{\varphi}}_i = \frac{1}{n_i(\mathrm{T})} \sum_{t:\mathbf{M}_t=i} \boldsymbol{\gamma}_t \cdot \left(\boldsymbol{v}_{(d(\mathbf{M}_t),a_t)} + \|\boldsymbol{\theta}_{\triangle}\|^2 \right) \cdot \|\boldsymbol{\theta}_{\triangledown}\|^2 \tag{1}$$



(a) Pre-Defense Performance (b) Post-Training by REE (c) Pipeline

Figure 1: SubFig. (a) illustrates the initial performance of the model group on training dataset-similar and OOD data, showing the challenges posed by adversarial examples and increased distribution differences. SubFig. (b) shows the improvement in model recognition ability after robust training of the model cohort in REE. The simplified pipeline of DSA, KASD, and REE is shown in SubFig. (c).

where $\boldsymbol{\gamma}_t$ represents the elasticity measure observed at time t, $d(\mathbf{M}_t)$ represents the defense strategy adopted by model $\mathbf{M}_t$, $\boldsymbol{v}_{(d,a)}$ are basis vectors specified in the feature space. $\boldsymbol{\theta}_{\triangle} = \boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{avg}}$ and $\boldsymbol{\theta}_{\triangledown} = \boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{prev}}$, $\boldsymbol{\theta}_t$ represents the parameter vector of the model at time t, $\boldsymbol{\theta}_{\text{avg}}$ is the historical average vector of model parameters, and $\boldsymbol{\theta}_{\text{prev}}$ is the model parameter at the previous time point. Furthermore, the first and second moments of the random quantity $\hat{\varphi}_i$ are expected to be closely consistent with the true cometric, ensuring that $\mathbb{E}[\hat{\boldsymbol{\varphi}}_i]$ is close to the actual synergy, while the covariance $\text{Cov}(\hat{\boldsymbol{\varphi}}_i)$ remains tightly coupled.

To robustly assess collective network synergies, we employ a multidimensional robust estimator. This estimator treats the set of $\{\hat{\boldsymbol{\varphi}}_i\}_{i=1}^{\mathbf{M}}$ as a sequence of observations derived from the network's interaction with adversarial instances. Assuming that the adversarial environment changes slowly enough for the network to adapt, and that the contribution of each model is not negligible, the estimated synergy score $\hat{\boldsymbol{\sigma}}$ converges to the true synergy value $\mathrm{T} \to \infty$, ensuring the collective integrity of the model.

Adjusting Individual Model Weights. We first define a recalibration function that dynamically adjusts the weight of each model's contribution based on each model's performance and adversarial environment. Let $\boldsymbol{\beta}_{i,t}$ represent the weight of the i_{th} model at time t, and let $\mathcal{A}$ represent the current data distribution. The recalibration function is defined as:

$$\boldsymbol{\beta}_{i,t+1} = \boldsymbol{\beta}_{i,t} \cdot \varrho(\ell_{i,t}, \mathcal{A}_t) \cdot \min(\eta, \tanh(-(\mathcal{A}_t \diamondsuit \mathcal{A}_{t-1}))) \tag{2}$$

where $\ell_{i,t}$ is the loss generated by the i_{th} model at time t, and η is an adjustable parameter. $\varrho(\ell_{i,t}, \mathcal{A}_t)$ is a weight adjustment function, and $\mathcal{A}_t \diamondsuit \mathcal{A}_{t-1}$ represents the difference between the current data distribution $\mathcal{A}_t$ and the previous time point data distribution $\mathcal{A}_{t-1}$. This feature ensures that models that perform well under the current adversarial strategy are more prominent in the ensemble.

Policy Alignment and Iterative Reinforcement. To align the high-dimensional policies of the model population with the adversarial context, we define a policy vector $\mathbf{s}_i$ for each model. The alignment process involves projecting these vectors onto a recalibrated feature space. This is achieved by using a projection matrix $\mathbf{P}_t$ that evolves over time $\mathbf{s}_i^{(t+1)} = \mathbf{P}_t \cdot \mathbf{s}_i^{(t)}$. $\mathbf{P}_t$ is calculated based on the current adversarial environment and the historical performance of the strategy, ensuring that the recalibrated strategy is optimal against current and foreseeable threats. The recalibration process involves iterative cycles of unlearning and reinforcement. During the unlearning phase, strategies considered less effective or compromised are weakened:

$$\boldsymbol{\mathcal{U}}(\mathbf{s}_i^{(t)}) = \boldsymbol{\varsigma} \cdot \mathbf{s}_i^{(t)} \cdot \exp(\boldsymbol{\varrho}(\boldsymbol{\ell}_{i,t}, \mathcal{A}_t)) \cdot \mathbf{1}_{\{\ell_{i,t} > \tau\}} \tag{3}$$

where $\varsigma \in (0,1)$ is the attenuation factor, τ is the loss threshold. Strategies with losses greater than τ will be scaled down. In contrast, during the reinforcement phase, strategies that demonstrate effectiveness are strengthened:

$$\boldsymbol{\mathcal{B}}\left(\mathbf{s}_i^{(t)}\right) = \boldsymbol{\kappa} \cdot \log(1 + ||\mathbf{s}_i^{(t)}||^2) \cdot \mathbf{s}_i^{(t)} \cdot \mathbf{1}_{\{\ell_{i,t} \leq \tau\}} \tag{4}$$

where κ is the strengthening factor, and this design can encourage individual models to share critical effective knowledge.

Kernel Anomaly Smoothing Detection

The proposed Kernel Anomaly Smoothing Detection Module (KASD), which uses a smoothing feature function based on Gaussian mean embedding for anomaly detection and employs a multi-layer feedback structure to identify new attacks

Kernel-Based Anomaly Detection. KASD employs a kernel-based anomaly detection approach. Specifically, we improve the Gaussian Kernel-based Maximum Mean Difference (GK-MMD) measure on this model group robustness task, $\mathbf{A}^{(\mathrm{G})}(\cdot, \cdot k^{(\mathrm{G})})$, quantifying the difference between historical and current adversarial distribution (Gretton et al. 2012; Sutherland et al. 2016). The Gaussian kernel is defined as $\boldsymbol{k}^{(\mathrm{G})}(x,y) = \exp\left(-\frac{|x-y|^2}{2\sigma_\phi}\right)$, where σ_ϕ is a learnable length scale parameter. This approach effectively captures subtle changes in adversarial environments.

To capture more effective patterns in adversarial strategies, KASD incorporates Deep Kernel Learning (DKL) (Wilson

et al. 2016; Liu et al. 2020). This approach combines traditional kernel methods with deep learning to create more expressive feature representations. The depth kernel is defined as $\boldsymbol{k}^{(\mathrm{D})}(x,y) = (1-\boldsymbol{\mu}) \cdot \exp\left(-|\boldsymbol{\phi}(x)-\boldsymbol{\phi}(y)|^2/2\boldsymbol{\sigma}_\phi\right) + \boldsymbol{\mu} \cdot \exp\left(-|x-y|^2/2\boldsymbol{\sigma}_q\right)$, where $\boldsymbol{\mu}, \boldsymbol{\sigma}_\phi, \boldsymbol{\sigma}_q$ are learnable parameters, and $\boldsymbol{\phi}(\cdot)$ represents the deep neural network feature extractor. This approach enables KASD to detect and understand adversarial behavior.

The model swarm also employs sign (C2STS) and confidence (C2ST-L) classifier two-sample testing (C2ST) to differentiate between benign and adversarial inputs (Gretton et al. 2012; Lopez-Paz and Oquab 2016; Cheng and Cloninger 2022). These tests use a classifier to determine whether two distributions are different. For C2ST-S, kernel $\boldsymbol{k}_{\text{C-S}}(x,y) = \frac{1}{4}\mathbb{1}(f(x) > 0)\mathbb{1}(f(y) > 0)$ utilizes the symbols output by the classifier, while C2ST-L uses $\boldsymbol{k}_{\text{C-L}}(x,y) = \boldsymbol{f}(x)\boldsymbol{f}(y)$, focusing on The confidence of the discriminator. These techniques enable KASD to effectively distinguish between normal model behavior and adversary-influenced behavior.

Optimized Frequency Domain Analysis. To further enhance the detection capability, KASD uses a Smooth Feature Function (SCF) based on Gaussian kernel mean embedding to analyze adversarial strategies in the frequency domain (Sriperumbudur et al. 2010; Fukumizu et al. 2009; Jitkrittum et al. 2016). This approach $\mathbf{Q}_{\text{SCF}}(\cdot,\cdot)$ provides a new perspective on the structure and evolution of adversarial attacks, allowing the detection of high-dimensional patterns that may otherwise go unidentified.

KASD is integrated into the model population of defensive postures through a multi-layered feedback system that continuously updates and informs Dynamic Synergy Amplification (DSA). Integrating predictive components into defense strategies can significantly enhance their adaptability and robustness, ensuring that the foresight provided by KASD is effectively translated into viable defense strategies. In our framework, KASD's predictive insights are encoded as adjustments to the model synergy score, recalibrating the collective defense posture in real time $\hat{\boldsymbol{\varphi}}_{t+1} = \hat{\boldsymbol{\varphi}}_t + \eta_2 \cdot \nabla_{\hat{\boldsymbol{\varphi}}} \mathcal{L}\left(\mathcal{F}\left(\mathcal{H}_t\right), \mathcal{A}_t\right)$, where $\hat{\boldsymbol{\varphi}}_t$ is the current collaboration score, η_2 is the learning rate, $\mathcal{L}$ is the loss function that measures the deviation between the predicted and actual adversarial modes, $\mathcal{F}$ is the look-ahead function of KASD, $\mathcal{H}_t$ is the currently collected historical attack data, and $\mathcal{A}_t$ is the current data distribution.

KASD's proactiveness is achieved through continuous monitoring and analysis of adversarial patterns to detect and interpret the slightest signs of emerging threats. Inspired by the concepts of active learning and early warning systems (Settles 2011; Quansah, Engel, and Rochon 2010), KASD employs a series of anomaly detection and pattern recognition algorithms to identify potential threats before they arise. Analyzing the spectral properties of adversarial data distributions for early detection of coordinated attacks:

$$\boldsymbol{\mathcal{W}} = \sum_{i=1}^{n} \lambda_i \cdot \left(1 + \frac{||\tilde{\mathbf{v}}_i|| + \mathbf{v}_i^{\mathrm{T}} \mathbf{M}_i \mathbf{v}_i}{\vartheta + \boldsymbol{\alpha}_{i,t}}\right) \cdot \mathbf{H}\left(\frac{\lambda_i - \boldsymbol{\upsilon}}{\boldsymbol{\theta}_{\triangle}}\right) \tag{5}$$

where λ_i and $\mathbf{v}_i$ are the eigenvalues and eigenvectors of the adversarial data covariance matrix, and $\boldsymbol{\upsilon}$ is the threshold for determining important spectral components. $\tilde{\mathbf{v}}_i = \mathbf{v}_i - \mathbf{v}_{avg}$, where $\mathbf{v}_{avg}$ represents the mean of all feature vectors, $\mathbf{H}(\cdot)$ is the Sigmoid function. ϑ is a scale parameter.

KASD learns from every interaction with its adversary, iteratively improving its detection model. This iterative learning is similar to the online learning paradigm discussed in (Shalev-Shwartz et al. 2012), where the model continuously evolves based on new data. In our context, KASD updates its parameters after every adversarial encounter:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - a \cdot (\nabla_\theta \mathcal{L}(\mathcal{F}(\mathcal{H}_t), \mathcal{A}_t) + ||\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}||^2) \tag{6}$$

where $\boldsymbol{\theta}_t$ is the parameter of KASD at time t, and a is the adaptation rate. In summary, the role of Kernel Anomaly Smoothing Detection Module in combating OOD attacks is proactive, detecting new attack methods. These attack methods cannot be effectively detected using common methods, and KASD enhances the overall robustness of the model group.

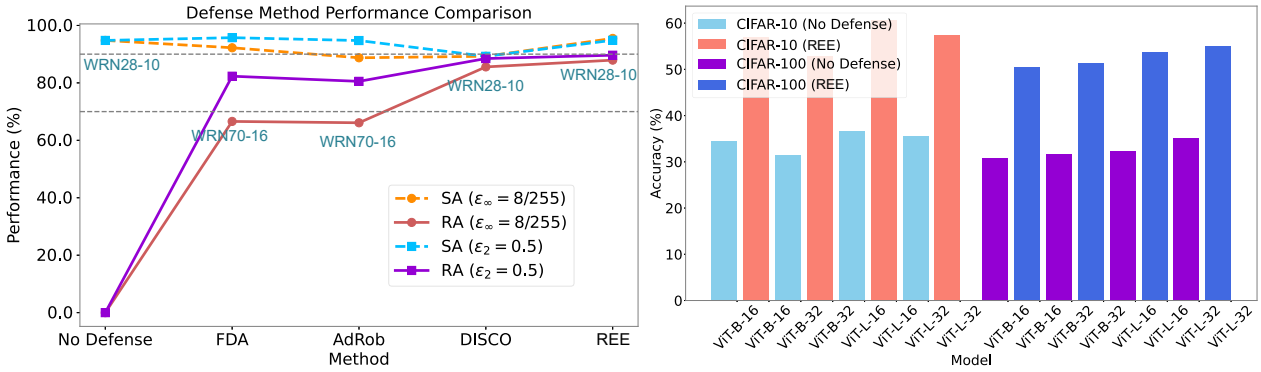
Collaborative Protocols and Training

Communication and Consensus. The framework is designed from the principles of distributed computing and peer-to-peer networks, where nodes (in this case, subgroup individuals in a model swarm) communicate directly with each other (Attiya and Welch 2004). The communication between any two models $\mathbf{M}_i$ and $\mathbf{M}_j$ is expressed as:

$$\mathcal{C}_{i,j} = \sum_{m=1}^{n} \sum_{\|\Theta_c\|=m} \Theta_c \cdot \mathbf{M}_i \cdot \mathbf{M}_j^{\mathrm{T}} \cdot (\mathbf{M}_j \cdot \Theta_c)^{m-1} \tag{7}$$

where $\boldsymbol{\mathcal{C}}_{i,j}$ is the communication channel between $\mathbf{M}_i$ and $\mathbf{M}_j$, and Θ_c represents the parameters that control the Communication Protocol (CP). This decentralized approach ensures that the network remains robust and can continue to operate effectively even if parts of the network are compromised. To achieve a unified defense, it is critical that models reach a consensus on the collective response to hostile threats. This is facilitated through a stochastic agreement protocol that guides the model to Consensus Update (CU) decisions over time. Inspired by work on stochastic optimization and multi-agent systems (Konečnỳ et al. 2016; Assran et al. 2019), the consensus process for a given defense strategy $\boldsymbol{\Gamma}_t$ at time t is formalized as $\boldsymbol{\Gamma}_{t+1} = \boldsymbol{\Gamma}_t + \xi_t \cdot \Delta\boldsymbol{\Gamma}_t$, where $\Delta\boldsymbol{\Gamma}_t$ is the update based on the current model state, and ξ_t is the learning rate parameter. This iterative process ensures that all models gradually adjust their strategies, resulting in a coherent and coordinated defense.

We use reinforcement learning techniques to continuously optimize the communication and consensus parameters in the network. This approach enables the network to gradually improve its collaboration strategy to better cope with new attacks. Under the framework of dynamic collaborative amplification, the communication and consensus modules ensure the overall processing efficiency within the network. Specifically, the design of these modules is based on the feature analysis of current attack methods, combined with



(a) Standard Accuracy and Robust Accuracy

(b) No-defense and Post-REE

Figure 2: (a) Standard accuracy (SA) and Robust Accuracy (RA) on WRN28-10 and WRN70-16 models under two attack scenarios ($\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$). (b) Comparison of the accuracy of the ViT model on the CIFAR-10 and CIFAR-100 datasets under no-defense and post-REE training scenarios. The 8 bars on the left represent the results on the CIFAR-10 dataset, while the 8 bars on the right correspond to the CIFAR-100 dataset.

the ensemble knowledge of all models in the network, thus providing a credible basis for the dynamic adjustment.

Consensus-Based Group Updates. The model swarm uses a consensus-based approach to model updates to ensure that all models in the model swarm contribute to the learning process. The approach draws on the concept of distributed learning, where multiple agents collaboratively learn a shared model (Li et al. 2020; Shayan et al. 2018). At each training step, model parameters are updated based on the weighted average of their neighbor parameters: $\mathbf{Z}_i^{(t+1)} = \sum_{j \in \mathcal{M}_i} w_{ij} \cdot \mathbf{Z}_j^{(t)}$, where $\mathbf{Z}_i^{(t+1)}$ is the update parameter of model i, $\mathbf{M}_i^{\mathcal{N}}$ is the adjacent model set of model i, w_{ij} is assigned to The weights of the model j parameters.

Experiments

Experimental Setup

Datasets and Training Configuration. The CIFAR-10 and CIFAR-100 datasets consist of images across 10 and 100 categories respectively, with the training and test sets comprising 50k and 10k images. The ImageNet dataset contains 1.2M training images and 50k test images (224×224), spanning a total of 1000 overall categories. The networks we utilize include ResNet-18 (He et al. 2016), WideResNet-28/32/70 (WRN28-10, WRN32-10 and WRN70-16) (Zagoruyko and Komodakis 2016). We opt for SGD as our optimizer, setting the momentum at 0.9. The weight decay and initial learning rate, adjusted using a piecewise decay scheduler, are set to 0.0005 and 0.1, respectively. Training is conducted over 200 epochs with a batch size of 128. The perturbation magnitude, measured by the L_p norm, is represented as ϵ_p. On these datasets, we generate training pairs with $\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$, using a step size of 2/255. The model implementation is based on the code from LIIF (Chen, Liu, and Wang 2021).

Quantitative Evaluation

Evaluation on RobustBench. We conducts robustness benchmark experiments on the CIFAR-10 dataset under adversarial conditions of $\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$, using WRN28-10 and WRN70-16 models as baselines. As shown in Fig. 2(a), the FDA (Rebuffi et al. 2021) and AdRob (Gowal et al. 2021, 2020) algorithms showed evident improvements in Robust Accuracy (RA), with FDA notably increasing RA to 66.58% and 82.32% under $\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$ attacks respectively, on the WRN70-16 model. On the WRN28-10 model, REE elevated RA to 87.92% and 89.63% under $\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$, respectively. This experiment shows the balance between maintaining standard model accuracy and defending against perturbations.

Robust evaluation of Vision Transformer. We analyze the accuracy performance of the Vision Transformer (Dosovitskiy et al. 2020) (ViT) model without defense and after REE training on the CIFAR-10 and CIFAR-100 datasets. Out-of-distribution knowledge training is performed by combining ResNet (He et al. 2016), Wide ResNet (Zagoruyko and Komodakis 2016), ResNeXt (Xie et al. 2017) and ConvNeXt (Liu et al. 2022), and the results are shown in Fig. 2(b). On the CIFAR-10 dataset, the ViT-B-16 model showed a significant improvement in accuracy, from 34.53% in the undefended scenario to 56.99% after REE training. Similarly, the larger model ViT-L-32 showed the 21.67% accuracy improvement, thus demonstrating the scalability of REE across different model sizes. For the CIFAR-100 dataset, ViT-L-16 was initially 32.24% and reached an accuracy of 53.77% after REE, a significant improvement of 21.53%.

Robustness Enhancement Comparison

We benchmark REE against five adversarial attack methods, including FGSM (Szegedy et al. 2013), PGD^{20} (Madry et al. 2017), C&W (Carlini and Wagner 2017), MIFGSM (Dong et al. 2017) and AutoAttack (Croce and Hein 2020). We compare with several common methods for improving model stability, namely BN (Ioffe and Szegedy 2015), LN

Dataset	Method	ResNet-18					WRN32-10				
		FGSM	PGD^{20}	C&W	MIFGSM	AutoAttack	FGSM	PGD^{20}	C&W	MIFGSM	AutoAttack
CIFAR-10	BN	56.70	52.16	78.46	54.96	47.69	60.65	55.06	82.24	58.47	52.24
	LN	53.88	45.44	75.51	50.72	41.48	57.80	49.74	79.39	54.68	46.44
	RNA	63.10	60.69	84.45	60.70	65.61	65.73	63.34	**85.68**	62.84	67.88
	REE (**Ours**)	**65.25**	**61.80**	**84.58**	**62.40**	**68.27**	**68.45**	**66.36**	83.61	**64.43**	**67.95**
CIFAR-100	BN	31.33	28.71	50.94	30.26	24.48	35.40	31.69	57.11	34.14	28.36
	LN	27.05	23.83	44.07	26.93	19.97	32.92	29.75	52.19	31.73	25.71
	RNA	36.76	35.55	**56.86**	34.00	42.12	37.87	36.04	**60.21**	35.58	42.43
	REE (**Ours**)	**37.63**	**36.81**	54.49	**35.70**	**45.72**	**38.48**	**38.60**	57.63	**36.70**	**43.29**

Table 1: Robustness comparison of REE and other methods on CIFAR-10 and CIFAR-100 datasets using ResNet-18 and WRN32-10 models.

Defense Complexity	#	Softmax						Voting					
		PGD 500	**CW** 500	**AA** 4.9k	**CAA** 1.8k	**MORA** 500	$MORA^{mt}$ 1.4k	**PGD** 500	**CW** 500	**AA** 4.9k	**CAA** 1.8k	**MORA** 500	$MORA^{mt}$ 1.4k
ADP	3	5.98	7.72	0.98	3.34	0.59	0.34	9.32	11.84	6.13	8.29	0.64	0.29
	5	7.10	8.70	2.18	4.25	0.97	0.67	12.42	12.05	10.13	0.67	1.17	0.62
	8	7.22	9.59	3.94	6.04	1.70	1.32	12.53	10.50	9.21	1.69	3.16	1.65
Dverage	3	44.49	40.17	30.58	32.98	25.77	25.26	31.48	28.00	24.98	27.65	23.57	22.91
	5	**54.61**	**52.83**	43.29	46.65	40.02	39.50	**44.28**	**42.28**	39.20	**40.85**	35.06	34.46
	8	59.13	58.25	56.71	56.89	55.68	55.57	53.72	52.35	50.04	51.15	47.12	46.10
GAL	3	8.13	11.57	0.85	1.00	0.67	0.51	5.85	7.64	0.56	0.78	0.87	0.35
	5	37.59	35.52	23.90	25.11	17.45	16.05	29.33	27.62	20.82	22.17	12.96	12.25
	8	53.39	52.56	37.46	35.30	28.71	27.44	49.56	48.02	31.39	30.93	21.66	20.16
TRS^{+}	3	14.01	10.87	8.46	9.75	8.11	7.60	10.19	8.71	6.69	8.08	5.73	5.44
	5	15.91	15.28	13.20	13.78	12.67	12.47	12.71	11.88	10.30	11.21	8.82	8.38
	8	18.02	17.59	16.51	16.73	15.90	15.64	14.57	13.48	11.85	12.80	11.39	10.69
REE (**Ours**)	3	**45.39**	**41.76**	**31.19**	**33.60**	**26.31**	**26.94**	**32.63**	**29.41**	**25.68**	**28.54**	**24.33**	**23.90**
	5	53.90	51.58	**44.11**	**46.85**	**40.31**	**39.72**	43.21	41.18	**39.61**	40.82	**36.35**	**35.96**
	8	**59.52**	**58.74**	**56.79**	**57.73**	**56.59**	**56.52**	**54.98**	**53.44**	**51.83**	**52.71**	**47.20**	**46.40**

Table 2: Compare the accuracy after being attacked by PGD (Madry et al. 2017), C&W (Carlini and Wagner 2017), AutoAttack (AA) (Croce and Hein 2020), CAA (Mao et al. 2021) and MORA (Gao, Xu et al. 2022) with different ensemble strategies in 2 ensemble modes (Softmax and Voting). The worst complexity for the number of iterations is shown in the "Complexity" row. The standard deviation of all results is within $\pm$ 0.05%. Consistent with the statement of the MORA (Gao, Xu et al. 2022), where $MORA^{mt}$ is represented as a multi-target attack, and 100 iterations are performed on the remaining 9 labels.

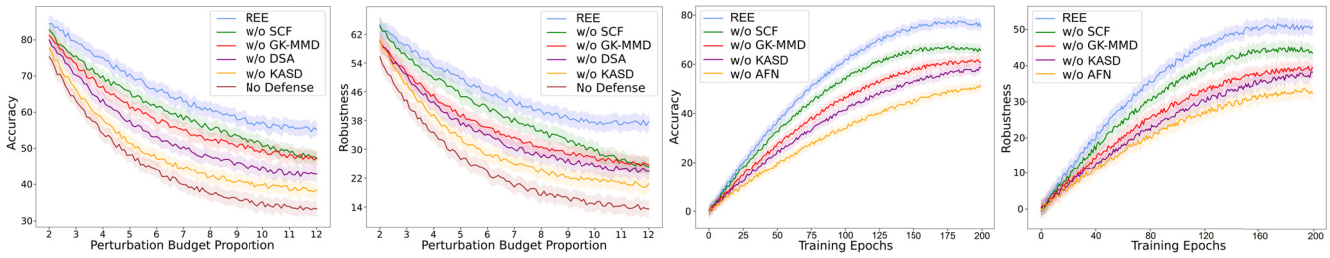
(Ba, Kiros, and Hinton 2016), and RNA (Dong et al. 2022). The experimental results are shown in Table 1, on the CIFAR-10 dataset, REE significantly outperforms other methods, especially the performance under adversarial attacks PGD^{20} and AutoAttack. Specifically, REE achieved a robustness of 61.80% and 66.36% on the ResNet-18 and WRN32-10 models respectively, which is the best among all comparison methods. On the CIFAR-100 dataset, under the most challenging AutoAttack attack, REE achieved 45.72% and 43.29% robustness on the ResNet-18 and WRN32-10 models, respectively. The robustness of REE was particularly highlighted under more challenging multi-vector attacks, showcasing significant improvements in both detection accuracy and mitigation effectiveness compared to existing methods.

Comparison of SOTA Methods

We compared the accuracy performance of different integration strategies (Softmax and Voting) when facing different attack methods (PGD (Madry et al. 2017), C&W (Carlini and Wagner 2017), AA (Croce and Hein 2020), CAA (Mao et al. 2021), MORA (Gao, Xu et al. 2022)). We compare with four defense methods, namely ADP (Pang et al. 2019), Dverage (Yang et al. 2020), GAL (Kariyappa and Qureshi 2019), and TRS^{+} (Yang et al. 2021), which are the most widely compared algorithms. As shown in Table 2, the REE defense method shows excellent robustness under various attack methods. In Softmax and Voting modes, the REE defense method has higher accuracy than other defense methods under different numbers of sub-models (3, 5, 8). In comparison, the REE defense method shows the best accuracy under various attacks in Softmax and Voting modes. Under different numbers of sub-models, the accuracy of the REE defense method is higher than other defense methods, especially in MORA (Gao, Xu et al. 2022). Excellent performance under $MORA^{mt}$ attacks.

Ablation Study

Our ablation study evaluated the impact of individual components within the REE on the overall model accuracy and robustness (as shown in Fig. 3). SubFig. (a) reveals a decrease



(a) Accuracy (b) Robustness (c) Accuracy (d) Robustness

Figure 3: Ablation studies on the influence of single component removal within the REE framework on model performance. SubFig. (a) and SubFig. (b) illustrate the impact of changes in Perturbation budget proportion on accuracy and robustness, respectively. SubFig. (c) and SubFig. (d) show the effects of varying Training Epochs on accuracy and robustness, respectively.

$(\boldsymbol{h}_1, \boldsymbol{h}_2)$	Re.	$\mathbf{C} \odot \mathbf{T}$	$\mathbf{C} \odot \mathbf{S}$	$\mathbf{C} \odot \mathbf{B}$	$\mathbf{C} \odot \mathbf{L}$	$(\boldsymbol{h}_1, \boldsymbol{h}_2)$	Re.	$\mathbf{C} \odot \mathbf{T}$	$\mathbf{C} \odot \mathbf{S}$	$\mathbf{C} \odot \mathbf{B}$	$\mathbf{C} \odot \mathbf{L}$
$(\mathbf{2.0}, \mathbf{0.5})$	$\mathbf{224^2}$	55.83	55.24	52.80	51.48	$(\mathbf{1.0}, \mathbf{2.0})$	$\mathbf{224^2}$	61.60	60.64	57.16	53.58
	$\mathbf{384^2}$	56.75	58.61	59.67	61.40		$\mathbf{384^2}$	62.25	63.96	64.95	65.85
$(\mathbf{2.0}, \mathbf{1.0})$	$\mathbf{224^2}$	55.32	54.17	51.64	50.18	$(\mathbf{0.5}, \mathbf{2.0})$	$\mathbf{224^2}$	57.47	54.30	51.31	48.69
	$\mathbf{384^2}$	56.70	57.48	60.43	62.83		$\mathbf{384^2}$	58.62	55.15	55.77	58.21
$(\mathbf{1.0}, \mathbf{1.0})$	$\mathbf{224^2}$	56.54	53.22	52.58	50.52	$(\mathbf{0.5}, \mathbf{3.0})$	$\mathbf{224^2}$	55.13	52.26	50.84	47.41
	$\mathbf{384^2}$	57.99	59.30	60.17	63.29		$\mathbf{384^2}$	56.54	52.86	53.19	55.32

Table 3: The experimental results of REE's hyperparameter combination $(\boldsymbol{h}_1, \boldsymbol{h}_2)$ on the ConvNeXt model show the percentage improvement in robustness compared to the undefended scenario. For ease of identification, optimal hyperparameter settings are represented with a gray background.

in accuracy as the perturbation budget proportion increases. Similarly, SubFig. (b) describes a parallel reduction in robustness with an increase in perturbation budget proportion, following the same relational hierarchy observed for accuracy. In SubFig. (c) and SubFig. (d), both accuracy and robustness are enhanced with an increase in training epochs. This trend indicates the models' learning efficacy, gradually improving their defensive stance against adversarial perturbations. This observation underscores the critical importance of balancing training duration and perturbation constraints to optimize both accuracy and robustness. It is worth mentioning that REE consistently outperforms variants that shield single or multiple components, demonstrating the overall contribution of each component to achieve a more superior performance.

Hyperparameter Combination Analysis

We performs a robustness analysis on ConvNeXt (trained on ImageNet-22K) (Liu et al. 2022) for the hyperparameter combination $(\boldsymbol{h}_1, \boldsymbol{h}_2)$ in REE, as shown in Table 3. We use four versions of ConvNeXt, namely ConvNeXt-Tiny ($\mathbf{C} \odot \mathbf{T}$), ConvNeXt-Small ($\mathbf{C} \odot \mathbf{S}$), ConvNeXt-Base ($\mathbf{C} \odot \mathbf{B}$) and ConvNeXt-Large ($\mathbf{C} \odot \mathbf{L}$). Experiments were conducted at two resolutions (224^2 and 384^2). The hyperparameter combination $(1.0, 2.0)$ performs best under all configurations, and its effect is highlighted with a gray background. At 384^2 resolution, this combination achieves a 65.85% robustness improvement on the ConvNeXt-Large model, and a 62.25% improvement on the Tiny variant, showing that REE can improve both large-scale networks and small-scale networks.

Conclusion

In this paper, we propose the Reliable Defense Ensemble (REE) to address the challenge of using ensemble knowledge to counter Out-of-Distribution (OOD) attacks. REE optimizes the ensemble knowledge of models through Dynamic Synergy Amplification (DSA) and the Kernel Anomaly Smoothing Detection Module (KASD). DSA allocates specific weights and adjusts strategies to enhance the robustness of the ensemble, thereby sharing critical knowledge between models. KASD detects new anomalies using a smoothing feature function based on Gaussian kernel mean embedding and a multi-layer feedback structure. To maintain effective knowledge sharing among models, we also propose using reinforcement learning to iteratively fine-tune communication and consensus parameters. Extensive experiments on benchmark datasets and various scenarios demonstrate that REE achieves state-of-the-art performance in utilizing ensemble model knowledge to counter OOD attacks.

Acknowledgments

This work was supported by the Jiangsu Province Science Foundation for Youths (BK20240463), the Laboratory for Advanced Computing and Intelligence Engineering Fund, the Xiaomi Young Talents Program, the China Postdoctoral Science Foundation (2024M753115), the National Natural Science Foundation of China (62172385), the Natural Science Foundation of Jiangsu Province (BK20241819), and the Innovation Program for Quantum Science and Technology (2021ZD0302900).

References

Assran, M.; Loizou, N.; Ballas, N.; and Rabbat, M. 2019. Stochastic gradient push for distributed deep learning. In *International Conference on Machine Learning*, 344–353. PMLR.

Attiya, H.; and Welch, J. 2004. *Distributed computing: fundamentals, simulations, and advanced topics*, volume 19. John Wiley & Sons.

Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bharti, S.; Zhang, X.; Singla, A.; and Zhu, J. 2022. Provable Defense against Backdoor Policies in Reinforcement Learning. *Advances in Neural Information Processing Systems*.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. Ieee.

Chen, C.; Ding, N.; and Carin, L. 2015. On the convergence of stochastic gradient MCMC algorithms with high-order integrators. *Advances in neural information processing systems*.

Chen, Y.; Liu, S.; and Wang, X. 2021. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8628–8638.

Cheng, X.; and Cloninger, A. 2022. Classification logit two-sample testing by neural networks for differentiating near manifold densities. *IEEE Transactions on Information Theory*, 68(10): 6631–6662.

Cheng, Y.; Diakonikolas, I.; and Ge, R. 2019. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the thirtieth annual ACM-SIAM symposium on discrete algorithms*, 2755–2771. SIAM.

Croce, F.; Gowal, S.; Brunner, T.; Shelhamer, E.; Hein, M.; and Cemgil, T. 2022. Evaluating the adversarial robustness of adaptive test-time defenses. In *International Conference on Machine Learning*, 4421–4435. PMLR.

Croce, F.; and Hein, M. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, 2206–2216. PMLR.

Diakonikolas, I.; Kamath, G.; Kane, D.; Li, J.; Moitra, A.; and Stewart, A. 2019. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2): 742–864.

Dong, M.; Chen, X.; Wang, Y.; and Xu, C. 2022. Random normalization aggregation for adversarial defense. *Advances in Neural Information Processing Systems*, 35: 33676–33688.

Dong, X.; Luu, A. T.; Ji, R.; and Liu, H. 2021. Towards robustness against natural language word substitutions. *arXiv preprint arXiv:2107.13541*.

Dong, Y.; Liao, F.; Pang, T.; Hu, X.; and Zhu, J. 2017. Discovering adversarial examples with momentum. *arXiv preprint arXiv:1710.06081*, 5.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fukumizu, K.; Gretton, A.; Lanckriet, G.; Schölkopf, B.; and Sriperumbudur, B. K. 2009. Kernel choice and classifiability for RKHS embeddings of probability distributions. *Advances in neural information processing systems*, 22.

Gao, X.; Xu, C.-Z.; et al. 2022. MORA: Improving ensemble robustness evaluation with model reweighing attack. *Advances in Neural Information Processing Systems*.

Ge, C.; Fu, X.; He, P.; Wang, K.; Cao, C.; and Zha, Z.-J. 2024. Neuromorphic Event Signal-Driven Network for Video Deraining. In *AAAI*, volume 38, 1878–1886.

Ge, C.; Fu, X.; and Zha, Z.-J. 2022. Learning Dual Convolutional Dictionaries for Image De-raining. In *ACM MM*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Gowal, S.; Qin, C.; Uesato, J.; Mann, T.; and Kohli, P. 2020. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*.

Gowal, S.; Rebuffi, S.-A.; Wiles, O.; Stimberg, F.; Calian, D. A.; and Mann, T. A. 2021. Improving robustness using generated data. *Advances in Neural Information Processing Systems*, 34: 4218–4233.

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. 2012. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1): 723–773.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Ho, C.-H.; and Vasconcelos, N. 2022. DISCO: Adversarial defense with local implicit functions. *Advances in Neural Information Processing Systems*, 35: 23818–23837.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. pmlr.

Jitkrittum, W.; Szabó, Z.; Chwialkowski, K. P.; and Gretton, A. 2016. Interpretable distribution features with maximum testing power. *Advances in Neural Information Processing Systems*, 29.

Kariyappa, S.; and Qureshi, M. K. 2019. Improving adversarial robustness of ensembles with diversity training. *arXiv preprint arXiv:1901.09981*.

Konečnỳ, J.; McMahan, H. B.; Ramage, D.; and Richtárik, P. 2016. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.

Li, Y.; Lyu, X.; Ma, X.; Koren, N.; Lyu, L.; Li, B.; and Jiang, Y.-G. 2023. Reconstructive Neuron Pruning for Backdoor Defense. *arXiv preprint arXiv:2305.14876*.

Liu, F.; Xu, W.; Lu, J.; Zhang, G.; Gretton, A.; and Sutherland, D. J. 2020. Learning deep kernels for non-parametric two-sample tests. In *International conference on machine learning*, 6316–6326. PMLR.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11976–11986.

Lopez-Paz, D.; and Oquab, M. 2016. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Mao, X.; Chen, Y.; Wang, S.; Su, H.; He, Y.; and Xue, H. 2021. Composite adversarial attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Mazeika, M.; Li, B.; and Forsyth, D. 2022. How to steer your adversary: Targeted and efficient model stealing defenses with gradient redirection. In *International Conference on Machine Learning*, 15241–15254. PMLR.

Pang, T.; Xu, K.; Du, C.; Chen, N.; and Zhu, J. 2019. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*. PMLR.

Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z. B.; and Swami, A. 2016. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, 372–387. IEEE.

Quansah, J. E.; Engel, B.; and Rochon, G. L. 2010. Early warning systems: a review. *Journal of Terrestrial Observation*, 2(2): 5.

Rebuffi, S.-A.; Gowal, S.; Calian, D. A.; Stimberg, F.; Wiles, O.; and Mann, T. 2021. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*.

Seligmann, F.; Becker, P.; Volpp, M.; and Neumann, G. 2024. Beyond Deep Ensembles: A Large-Scale Evaluation of Bayesian Deep Learning under Distribution Shift. *Advances in Neural Information Processing Systems*, 36.

Settles, B. 2011. From theories to queries: Active learning in practice. In *Active learning and experimental design workshop in conjunction with AISTATS 2010*, 1–18. JMLR Workshop and Conference Proceedings.

Shalev-Shwartz, S.; et al. 2012. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2): 107–194.

Shayan, M.; Fung, C.; Yoon, C. J.; and Beschastnikh, I. 2018. Biscotti: A ledger for private and secure peer-to-peer machine learning. *arXiv preprint arXiv:1811.09904*.

Shen, G.; Liu, Y.; Tao, G.; Xu, Q.; Zhang, Z.; An, S.; Ma, S.; and Zhang, X. 2022. Constrained optimization with dynamic bound-scaling for effective NLP backdoor defense. In *International Conference on Machine Learning*. PMLR.

Shi, Z.; Chen, Z.; Xu, Z.; Yang, W.; Yu, Z.; and Huang, L. 2022. Shape prior guided attack: Sparser perturbations on 3d point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8277–8285.

Sitawarin, C.; Golan-Strieb, Z. J.; and Wagner, D. 2022. Demystifying the adversarial robustness of random transformation defenses. In *International Conference on Machine Learning*, 20232–20252. PMLR.

Sriperumbudur, B. K.; Gretton, A.; Fukumizu, K.; Schölkopf, B.; and Lanckriet, G. R. 2010. Hilbert space embeddings and metrics on probability measures. *The Journal of Machine Learning Research*, 11: 1517–1561.

Sutherland, D. J.; Tung, H.-Y.; Strathmann, H.; De, S.; Ramdas, A.; Smola, A.; and Gretton, A. 2016. Generative models and model criticism via optimized maximum mean discrepancy. *arXiv preprint arXiv:1611.04488*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tramèr, F.; Kurakin, A.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*.

Wilson, A. G.; Hu, Z.; Salakhutdinov, R.; and Xing, E. P. 2016. Deep kernel learning. In *Artificial intelligence and statistics*, 370–378. PMLR.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500.

Yang, H.; Zhang, J.; Dong, H.; Inkawhich, N.; Gardner, A.; Touchet, A.; Wilkes, W.; Berry, H.; and Li, H. 2020. Dverge: diversifying vulnerabilities for enhanced robust generation of ensembles. *Advances in Neural Information Processing Systems*, 33: 5505–5515.

Yang, Z.; Li, L.; Xu, X.; Zuo, S.; Chen, Q.; Zhou, P.; Rubinstein, B.; Zhang, C.; and Li, B. 2021. Trs: Transferability reduced ensemble via promoting gradient diversity and model smoothness. *Advances in Neural Information Processing Systems*, 34: 17642–17655.

Zagoruyko, S.; and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zhang, Y.; Albarghouthi, A.; and D'Antoni, L. 2022. BagFlip: A Certified Defense against Data Poisoning. *Advances in Neural Information Processing Systems*, 35: 31474–31483.

Zhou, D.; Wang, N.; Yang, H.; Gao, X.; and Liu, T. 2023a. Phase-aware adversarial defense for improving adversarial robustness. In *International Conference on Machine Learning*, 42724–42741. PMLR.

Zhou, T.; Luo, Y.; Ren, S.; and Xu, X. 2023b. NNSplitter: An Active Defense Solution to DNN Model via Automated Weight Obfuscation. *arXiv preprint arXiv:2305.00097*.

Zhu, C.; Roos, S.; and Chen, L. Y. 2023. LeadFL: client self-defense against model poisoning in federated learning. In *International Conference on Machine Learning*. PMLR.

Ziegel, E. R. 2003. The elements of statistical learning. *Technometrics*, 45(3): 267.

Zou, H.; and Hastie, T. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2): 301–320.