arXiv:2509.11525v1 [cs.LG] 15 Sep 2025

DARD: Dice Adversarial Robustness Distillation against Adversarial Attacks

Jing Zou[1], Shungeng Zhang[1], Meikang Qiu[1*] Chong Li[2]

[1] Augusta University, Augusta, GA 30901, USA
[2] Electrical Engineering Dept., Columbia University, New York, NY 10027, USA
jizou@augusta.edu szhang2@augusta.edu mqiu@augusta.edu
cl3607@columbia.edu

Abstract. Deep learning models are vulnerable to adversarial examples, posing critical security challenges in real-world applications. While Adversarial Training (*AT*) is a widely adopted defense mechanism to enhance robustness, it often incurs a trade-off by degrading performance on unperturbed, natural data. Recent efforts have highlighted that larger models exhibit enhanced robustness over their smaller counterparts. In this paper, we empirically demonstrate that such robustness can be systematically distilled from large teacher models into compact student models. To achieve better performance, we introduce Dice Adversarial Robustness Distillation (*DARD*), a novel method designed to transfer robustness through a tailored knowledge distillation paradigm. Additionally, we propose *Dice Projected Gradient Descent* (*DPGD*), an adversarial example generalization method optimized for effective attack. Our extensive experiments demonstrate that the *DARD* approach consistently outperforms adversarially trained networks with the same architecture, achieving superior robustness and standard accuracy.

Keywords: Adversarial examples · Adversarial attacks · Robustness · Knowledge distillation · Adversarial training · Deep neural networks

1 Introduction

Deep learning has achieved remarkable success across various domains, from computer vision [8] and natural language processing [9] to the control of autonomous systems [10]. These advancements are driven by the unparalleled capacity of Deep Neural Networks (*DNNs*) to extract complex features and learn rich representations in diverse real-world applications. Despite their impressive performance, DNNs remain critically susceptible to adversarial examples – subtle, intentional perturbations that can lead to erroneous predictions while remaining undetected by human perception [24]. This vulnerability poses significant security risks in critical applications, such as autonomous driving [12] [11], large language models [14], and medical diagnosis [13], where adversarial attacks can lead to life-threatening outcomes.

Adversarial Training (*AT*) has emerged as one of the most effective approaches for improving robustness by incorporating adversarial examples into

the training process [15]. Despite its efficacy, *AT* suffers from three inherent limitations: First, *AT*-trained models often exhibit overfitting to specific attack patterns, limiting generalization to novel or adaptive threat vectors. Second, *AT* frequently leads to degraded performance on clean data (i.e., model's standard accuracy), creating a robustness-accuracy trade-off. Third, the iterative generation of adversarial examples during training imposes significant computational demands, making *AT* computationally intensive. These limitations hinder the deployment of robust models in resource-constrained environments.

In response to these challenges, researchers have explored various strategies to improve adversarial robustness. Regularization-based methods, such as adversarial weight perturbation and *TRADES* [4], aim to balance robustness and generalization by imposing constraints on model updates. Data augmentation approaches, leveraging diverse attack strategies or generative models, focus on enhancing defenses against unforeseen adversaries [5]. Certified defenses offer theoretical robustness guarantees within defined perturbation bound, but often suffer from high computational complexity and limited scalability [6]. Despite these efforts, the inherent limitations of adversarial training remain largely unaddressed, highlighting the urgent need for innovative solutions to improve both robustness and generalization.

Recent studies indicate a positive correlation between model size and adversarial robustness, suggesting that larger models tend to exhibit greater resistance to adversarial perturbations compared to smaller counterparts. This is potentially attributed to their capacity for capturing richer feature representations and more effectively mitigating the impact of adversarial noise. Knowledge Distillation (*KD*), a technique for transferring the knowledge of large, high-performing models to smaller, more efficient models, has proven to be effective for maintaining accuracy while reducing computational overhead [7]. Motivated by these insights, we introduce Dice Adversarial Robustness Distillation (*DARD*), a novel framework specifically designed to enhance the adversarial robustness of compact models beyond the capabilities of standard adversarial training.

Our *DARD* framework employs a teacher-student paradigm, where a robustly trained large model guides the learning of a smaller model by distilling knowledge through soft-label probabilities derived from the teacher's output distribution. Unlike traditional hard labels, soft labels retain inter-class relationships, enabling the student model to learn richer and more discriminative representations. Moreover, *DARD* leverages both natural and adversarial soft outputs in a joint objective, carefully balancing the influence of clean and perturbed examples. This balanced approach effectively mitigates the typical degradation of standard accuracy associated with adversarial training while achieving superior adversarial robustness in the student model.

Through extensive experimentation, we demonstrate that our *DARD* framework achieves state-of-the-art robustness-accuracy trade-offs, outperforming adversarially trained baselines of equivalent architecture on the standard datasets. These results highlight the efficacy of knowledge distillation as a powerful mechanism to decouple adversarial robustness from computational overhead, offering

valuable insights into the design of deep learning models that are simultaneously robust and computationally efficient. Our key contributions can be summarized as follows:

- Dice Adversarial Robustness Distillation (*DARD*): A novel framework that systematically transfers robustness from large, adversarially trained teachers to compact student models via dual supervision from natural and adversarial soft labels, eliminating the robustness-accuracy trade-off inherent in standard adversarial training.
- Generalization of *DPGD*: We adapt *Dice Projected Gradient Descent* – an attack strategy originally designed for semantic segmentation – to image classification, optimizing its perturbation efficiency and attack success rate through dynamic step-size tuning and channel-wise gradient masking.
- Empirical Validation: Comprehensive experiments show that *DARD*-trained lightweight models (e.g., ResNet-18) achieve better robust accuracy on CIFAR-10 and CIFAR-100, outperforming pre-trained small models of the same architecture, while maintaining near-identical natural accuracy.

The rest of this paper is structured as follows. Section 2 reviews related work on adversarial attacks, adversarial training, and knowledge distillation, highlighting key differences between existing approaches and our proposed framework. Section 3 introduces *DPGD* adversarial attack and Dice Adversarial Robustness Distillation (*DARD*) framework designed to enhance the student model's acquisition of adversarial features. Section 4 describes the experimental setup, including datasets, evaluation metrics, and implementation details, and presents a comprehensive analysis of the experimental results, demonstrating the effectiveness of *DARD* in improving both robust and standard accuracy. Finally, Section 5 concludes the paper.

2 Related Work

This section reviews existing research on adversarial attacks, training, and knowledge distillation. This review established the theoretical background and motivation for our proposed Dice Adversarial Robustness Distillation (*DARD*) method.

2.1 Adversarial Attacks

In 2013, *Szegedy et al.* [24] first introduced the concept of adversarial examples. Their research demonstrated that subtle, imperceptible to human eyes, can mislead deep neural networks into making incorrect decisions. The process of adding perturbations to the original input is defined as an adversarial attack, and the perturbed inputs are referred to as adversarial examples. Building up this foundation, *Goodfellow et al.* [17] proposed the Fast Gradient Sign Method (*FGSM*), a groundbreaking approach related to the Gradient Descent algorithm. The goal of generating adversarial examples is to maximize the loss function to induce wrong predictions in the model, as shown in Equation1.

$$\arg \max_{\|\delta\| \leq \epsilon} \ell(h_\theta(x+\delta), y) \tag{1}$$

FGSM efficiently generates adversarial examples by leveraging the gradients of the loss function to the input data, perturbing the inputs in the direction that maximizes the loss, thereby causing the model to make wrong predictions. FGSM is a single-step attack, while unable to achieve optimal adversarial loss. To overcome this limitation, subsequent attacks, such as Projected Gradient Descent (*PGD*) [15], *DeepFool* [26] and the Carlini & Wagner (*CW*) [25], these methods adopted multiple-step iterative strategies, gradually approaching more effective adversarial examples through multiple small-step updates. Moreover, for specialized tasks like semantic segmentation, the *SegPGD* [3] method distinguished between pixels that are correctly segmented and those that are not in its loss function design, and iteratively adjusts the weighting parameters for these two components to achieve enhanced attack performance. Inspired by these approaches, our proposed Dice Projected Gradient Descent (*PGD*) introduces a Dice Loss coupled with a dynamic weighting strategy to generate even more disruptive adversarial examples in our classification tasks.

2.2 Adversarial Training

Goodfellow et al. [17] proposed that deep neural networks are highly susceptible to adversarial examples and adversarial training as a regularization strategy to mitigate overfitting to natural data, similar to the effect of dropout. In their proposed adversarial training method, the loss function was defined as:

$$J_{\text{adv}}(\theta, x, y) = \alpha J(\theta, x, y) + (1-\alpha) J(\theta, x_{\text{adv}}, y). \tag{2}$$

This approach incorporates both natural and adversarial examples into the training process, thereby enhancing the model's robustness. Generally, α is set to 0.5. Specifically, it employs the *FGSM* to generate adversarial examples. *Madry et al.* [15] extended this research and introduced a more powerful adversarial training method, namely *PGD* adversarial training. Compared to *FGSM*, *PGD* optimizes adversarial examples through multiple iterative steps, making them more destructive and capable of exploring a broader perturbation space. This advancement significantly improves the robustness of adversarial training, enabling *PGD*-trained models to withstand stronger adversarial attacks while avoiding overfitting to specific perturbation patterns.

2.3 Knowledge Distillation

Knowledge distillation was originally introduced as a method for model compression and knowledge transfer by *Hinton et al.* [7]. They proposed that a compact student model can be trained to mimic the behavior of a larger, well-trained teacher network. This approach allowed the student model to maintain much of the teacher's predictive capability with lower computational cost. *Romero et*

al. [27] introduced *FitNets*, which leverage intermediate representations, named hints, to facilitate the transfer of knowledge from teacher to student. *Zagoruyko et al.* [28] proposed an attention transfer method, wherein the spatial attention maps derived from the teacher network guide the training of the student model, to bolster its representational capability. Researchers integrated adversarial training into the distillation framework to address the adversarial vulnerability in deep neural networks, transferring the robustness from teacher to student [2] [29] [30]. In these adversarial robustness distillation methods, the student model is trained to learn the teacher's capability on both clean and adversarial examples, enhancing defense against adversarial attacks.

3 Proposed Method

In this section, we introduce our proposed *DARD* method, which integrates our *DPGD* adversarial attack strategy devised into a teacher-student framework to improve adversarial robustness while maintaining high precision on natural inputs. Our *DARD* framework is shown in Fig.2.

3.1 *DPGD*: Dice Projected Gradient Descent

SegPGD [3] was originally proposed for generating adversarial perturbations in semantic segmentation tasks. Its core idea is to use Projected Gradient Descent (PGD) to introduce fine-grained pixel-level perturbations that disrupt the model's semantic understanding of an image. Inspired by this method, we adapt its iterative perturbation strategy and adversarial loss formulation to the image classification task.

Similar to the standard Projected Gradient Descent (*PGD*) framework, *DPGD* seeks to perturb the natural input X_{nat} within an ℓ_∞-ball of radius ϵ. However, unlike basic *PGD* attacks, *DPGD* introduces a novel application of Dice Loss and a dynamic weighting scheme that distinguishes between correctly and wrongly predicted samples. *DPGD* can be mathematically defined as follows:

$$\mathbf{X}_{adv}^{\mathrm{t}+1} = \phi^{\epsilon}\Big(\mathbf{X}_{adv}^{\mathrm{t}} + \alpha \cdot \mathrm{sign}\big(\nabla_{\mathbf{X}_{adv}^{\mathrm{t}}} L\big(f(\mathbf{X}_{adv}^{\mathrm{t}}), \mathbf{y}\big)\big)\Big), \tag{3}$$

where X_{adv}^{t} represents the adversarial example at iteration t, $L\big(f(\mathbf{X}_{adv}^{\mathrm{t}}), \mathbf{y}\big)$ denotes the model's output when given the current iteration adversarial input X_{adv}, $\nabla_{\mathbf{X}_{adv}^{\mathrm{t}}} L\big(f(\mathbf{X}_{adv}^{\mathrm{t}}), \mathbf{y}\big)$ computes the gradient of the loss function to X_{adv}^{t}, α is the step size controlling the magnitude of each update, $sign(\cdot)$ determines the direction of maximum loss increase, $\phi^{\epsilon}(\cdot)$ is the projection operator that ensures X_{adv}^{t} remains within an ℓ_∞-ball of radius ϵ. The process of *DPGD* is illustrated in Fig.1.

We define the loss function as two parts, as shown in Equation4. The first component calculates the loss for correctly classified samples, while the second component calculates the loss for misclassified samples. To generate highly effective adversarial examples, we introduce dynamically updated weights λ and

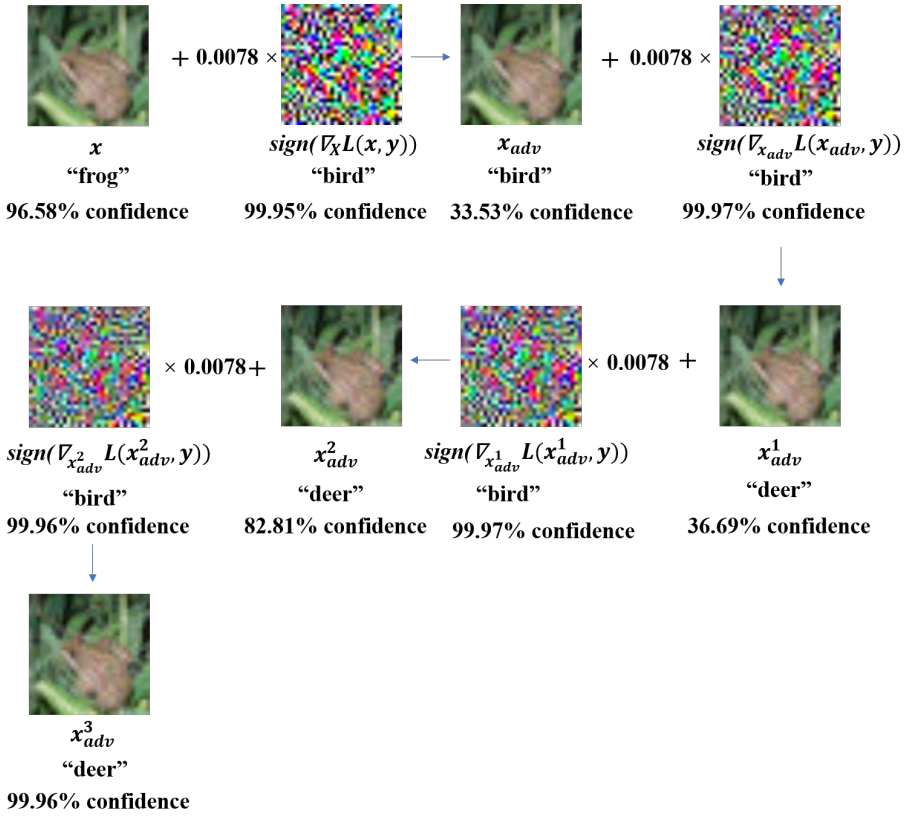


Fig. 1. Using CIFAR-10 samples and the DPGD algorithm with a step size of $\alpha = 2/255$, adversarial examples are generated by iteratively adding small perturbations that increase the model's loss, eventually leading to high-confidence misclassifications.

$1 - \lambda$. Through this approach, in the early stages of the attack iterations, we primarily perturb correctly classified samples. As the iterations progress, the number of misclassified samples gradually increases. In the later stages of the attack, we focus more on misclassified samples to prevent adversarial examples from becoming ineffective.

$$\mathcal{L}\big(f(\mathbf{x}_{\text{adv}}), \mathbf{y}\big) = (1-\lambda)\mathcal{L}_{p_T}\big(\mathbf{x}_{\text{adv}}, \mathbf{y}\big) + \lambda\mathcal{L}_{\rho_F}\big(\mathbf{x}_{\text{adv}}, \mathbf{y}\big). \tag{4}$$

To ensure that the weight λ dynamically adjusts throughout the attack iterations, reflecting different focuses in the early and later stages, we define λ as formulated in Equation5. This design allows the attack to prioritize perturbing correctly classified samples in the early iterations while gradually shifting its focus toward misclassified samples in the later iterations, preventing adversarial examples from becoming ineffective. In this equation, t represents the current iteration steps, and T denotes the total iteration steps.

$$\lambda = \frac{t-1}{2T}. \tag{5}$$

3.2 Adversarial Robustness Distillation

DeepSeek R1 open source enables researchers to distill small models with logical reasoning capabilities at a low cost. Inspired by this approach and supported by previous studies [1] [2] indicating that larger models exhibit greater robustness than smaller ones, we explore methods to transfer robustness from large models to compact models effectively. Enhanced robustness can be transferred from a teacher model to a student model, making knowledge distillation a promising approach for improving the robustness of small models. Despite the effectiveness of certain methods in enhancing the robustness of student models, their ability to accurately classify natural images still requires further improvement.

Our proposed method *DARD* aims to not only improve the robustness of the model but also maintain the high accuracy of classifying natural images. Traditional supervised learning typically uses one-hot hard labels to train models. Assuming there is a five-class classification problem, the true label of the input x is a one-hot vector:

$$y = [0, 1, 0, 0, 0] \tag{6}$$

In our *DARD* framework, we replace one-hot hard labels with soft labels, meaning that we use output probabilities as labels instead:

$$y' = [0.1, 0.7, 0.1, 0.05, 0.05] \tag{7}$$

A one-hot hard label only conveys the correct class to the model, whereas a soft label can transfer information about the relationships between different classes. In *DARD*, we leverage not only the probability output vectors of natural images from the large model but also those of adversarial images, using them as labels for adversarial examples in the small model. This design is motivated by the goal of transferring knowledge from the large model to enhance the small model's performance on both standard and adversarial input.

Specifically, for both natural examples and adversarial examples, we extract their probability distributions from a large teacher model and jointly use these soft labels to guide the training of a small student model. This approach not only preserves the student model's accuracy on clean samples but also enables it to leverage the extensive knowledge embedded within the teacher model, thereby enhancing its generalization capabilities. By integrating the teacher model's insights on natural examples and adversarial perturbations, the student model becomes better equipped to withstand adversarial attacks, ultimately resulting in improved robustness. The student model is trained via an adversarial training framework, wherein adversarial examples are generated using *DPGD* and simultaneously refined through teacher model distillation. The basic optimization

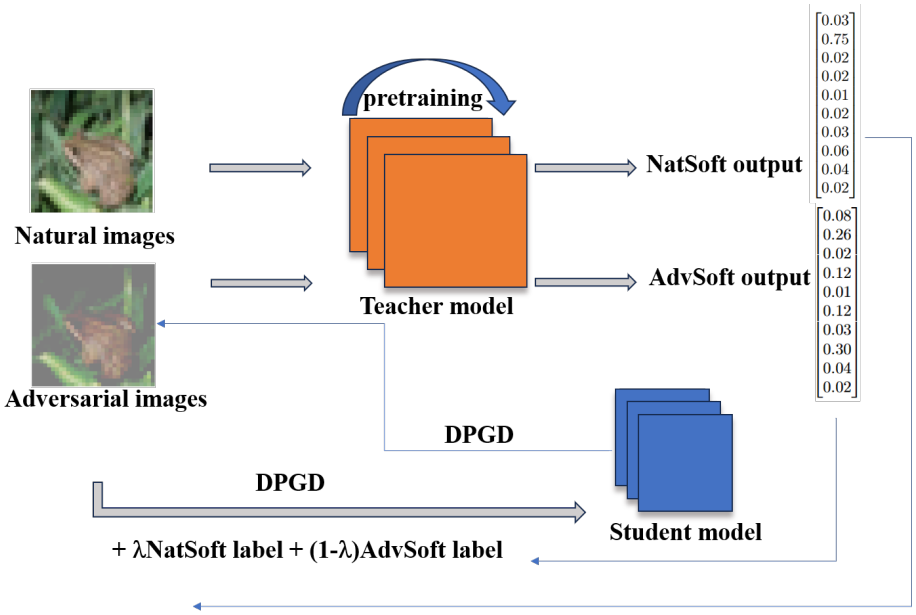


Fig. 2. The framework of our *DARD* method. First, we generate adversarial examples from the student model using the proposed *DPGD* attack. Next, these adversarial examples are used to train (pretrain) the teacher model adversarially. Finally, we derive a balanced soft probability from the teacher model, combining both natural and adversarial examples, and use this probability as the soft label to train the student model.

framework of *DARD* can be defined as follows:

$$\arg\min_{\theta_S} \big((1-\alpha)\, CE(S(x), y) + \alpha\tau^2 KL(S(x_{\text{adv}}), P(x)\big), \tag{8}$$

where θ_S denotes the student model parameters to be optimized, $S(\cdot)$ is the student model's output, y is the ground-truth label associated with x, $CE(S(x_{\text{adv}}, y)$ is the cross-entropy loss between the student's prediction on the adversarial examples and the true label, $KL(S(x), P(x))$ denotes the Kullback-Leibler divergence between the student's prediction on the natural input and the teacher model's soft distribution, $P(x)$ is the teacher model's probability distribution on the natural input, and α is the weighting coefficient that balance the distillation loss against the adversarial training loss, τ is the distillation temperature.

4 Experiment

4.1 Experiment setting

In our experiments, we evaluate the performance of the *DARD* method on two widely used image classification datasets: *CIFAR-10* and *CIFAR-100* [16]. We

compare our method with natural training and two state-of-the-art adversarial robustness methods: Standard Adversarial Training (*SAT*) [15] and Adversarially Robustness Distillation (*ARD*) [2]. All experiments run on CloudLab using its c240g5 node, each equipped with two Intel Xeon Silver 4114 CPUs (10 cores each, 2.20 GHz), 192 GB of ECC DDR4-2666 memory, an NVIDIA P100 GPU (12 GB), a 1 TB 7200 RPM SAS hard drive, and a 480 GB Intel DC S3500 SSD.

Student and Teacher Networks. In our experiments, we consider *ResNet-18* [22] as the student model, *ResNet-56* [22] as the teacher model following previous work. The teacher model is pre-trained before adversarial robustness distillation.

Defense	Clean	FGSM	PGD20	T-PGD	BIM	AutoAttack
Natural	**93.49**%	17.03%	0.00%	0.55%	0.00%	0.00%
SAT	81.33%	55.62%	49.68%	78.42%	50.27%	46.63%
ARD	84.01%	57.40%	50.71%	76.42%	50.12%	45.78%
TDARD	85.88%	54.78%	44.35%	**82.55**%	44.12%	42.58%
DARD	83.56%	**57.53**%	**52.63**%	81.11%	**53.07**%	**47.75**%

Table 1. White-box robustness of ResNet-18 on CIFAR-10

Defense	Clean	FGSM	PGD20	T-PGD	BIM	AutoAttack
Natural	**75.01**%	7.05%	0.00%	0.26%	0.00%	0.00%
SAT	58.59%	28.18%	23.79%	55.64%	23.80%	20.85%
ARD	59.50%	29.30%	25.61%	53.64%	21.60%	21.54%
TDARD	65.06%	29.90%	24.33%	**57.96**%	23.80%	19.60%
DARD	61.13%	**32.42**%	**28.80**%	57.40%	**28.74**%	**23.44**%

Table 2. White-box robustness of ResNet-18 on CIFAR-100

Adversarial Attacks. To comprehensively evaluate the robustness of our proposed method, we compare it against five widely used adversarial attack methods: *FGSM* (Fast Gradient Sign Method), *PGD* (Projected Gradient Descent), *T-PGD* (Target Projected Gradient Descent) [18], *BIM* (Basic Iterative Method) [19], and *AutoAttack* [20]. These attack methods cover a range of adversarial attack strategies, including single-step attacks (*FGSM*) and multi-step iterative attacks (*PGD*, *T-PGD*, *BIM*, and *AutoAttack*), ensuring diversity in adversarial examples. *AutoAttack* acts as a comprehensive attack framework that combines mul-

tiple adversarial strategies, including *Auto-PGD*, *APGD-CE*, *PGD-DLR*, and *Square Attack*. As a popular strategy for rigorously testing model robustness, *AutoAttack* subjects defense mechanisms to a broad and stringent evaluation. In our experiments, we standardized the hyper parameters for all multi-step iterative adversarial attacks to ensure a fair comparison. Specifically, we set the number of iterations to 20, the maximum perturbations value $\epsilon = 8/255$, and the step size $\alpha = 2/255$. These setting ensure a controlled and consistent evaluation of the model's robustness against iterative adversarial attacks.

Attack	**Defense**	**Clean**	**Robust**	**W-Robust**
FGSM	Natural	93.49%	17.03%	55.26%
	SAT	81.33%	55.62%	68.48%
	ARD	84.01%	57.40%	70.71%
	TDARD	85.88%	54.78%	**70.33%**
	DARD	83.56%	57.53%	**70.55%**
PGD20	Natural	93.49%	0.00%	46.75%
	SAT	81.33%	49.68%	65.51%
	ARD	84.01%	50.71%	67.36%
	TDARD	85.88%	44.35%	**65.12%**
	DARD	83.56%	52.63%	**68.10%**
T-PGD	Natural	93.49%	0.55%	47.02%
	SAT	81.33%	78.42%	79.88%
	ARD	84.01%	76.42%	80.22%
	TDARD	85.88%	82.55%	**84.22%**
	DARD	83.56%	81.11%	**82.34%**
BIM	Natural	93.49%	0.00%	46.75%
	SAT	81.33%	50.27%	65.80%
	ARD	84.01%	50.12%	67.07%
	TDARD	85.88%	44.12%	**65.00%**
	DARD	83.56%	53.07%	**68.32%**
AutoAttack	Natural	93.49%	0.00%	46.75%
	SAT	81.33%	46.63%	63.98%
	ARD	84.01%	45.78%	64.90%
	TDARD	85.88%	42.58%	**64.23%**
	DARD	83.56%	47.75%	**65.66%**

Table 3. White-box trade-off robustness of ResNet-18 on CIFAR-10

4.2 Training and Evaluation

We employ the Stochastic Gradient Descent (*SGD*) optimizer to train our student model, setting the learning rate to 0.1, momentum to 0.9, and weight decay

to 5e-4. This configuration is commonly used in training neural networks as it effectively balances convergence speed and generalization. In our *DARD* method, we enhance the adversarial robustness of the student model by adopting an adversarial strategy that employs only adversarial examples. Specifically, for each original sample, we generate its corresponding adversarial example by *DPGD* and use these adversarial examples exclusively to train the student model. Meanwhile, we evaluated various values for the hyperparameter λ during the construction of soft labels for the student model and found that setting λ to 0.5 yielded the preferred performance. In other words, for a given sample, we obtain the teacher model's prediction distribution on the natural sample and on the adversarial example, then average these two distributions to form the soft label. This allows the student model to simultaneously absorb the teacher's knowledge of both clean and adversarial distributions, thereby preserving an understanding of the clean data distribution despite training solely on adversarial examples and ultimately improving overall robustness and generalization. In our experiments, we also evaluated another adversarial training strategy within the *DARD* framework, referred to as *TDARD*, which trains the student model using both natural and adversarial examples simultaneously. Although both strategies operate under the same framework, they emphasize different aspects. Our experimental results indicate that the first strategy demonstrates a more comprehensive performance. The results are presented in Table1 and 2.

In our comparative experiments, to ensure consistency with *DARD* approach, we adopted the core design principles of the *SAT* and *ARD* methods and implemented them on the same *ResNet18* architecture used by our student model. Additionally, to better compare these methods and *DARD* balance accuracy on both clean and adversarial examples, we introduce Weighted Robust Accuracy [23] as our evaluation metric. Specifically, this metric takes into account the accuracies on clean and adversarial examples by averaging them, thus providing a comprehensive reflection of the model's performance under both natural and adversarial examples. Table3 shows the results.

4.3 Ablation Study

To gain deeper insights into the role of each part in the *DARD* framework and their contributions to adversarial robustness distillation, we conducted a series of systematic ablation studies. The results are shown in Fig.3, and the change of total loss is shown in Fig.4.

Effect of the *DPGD* Module. Specifically, we replaced our proposed *DPGD* with the traditional 20-step *PGD* to quantitatively assess the impact of *DPGD* on the overall adversarial robustness and classification performance of *DARD*. The results highlight the effectiveness of the *DPGD* module within the *DARD* framework. Specifically, our *DARD* method, enhanced with *DPGD*, achieves a more balanced performance across both natural and adversarial examples.

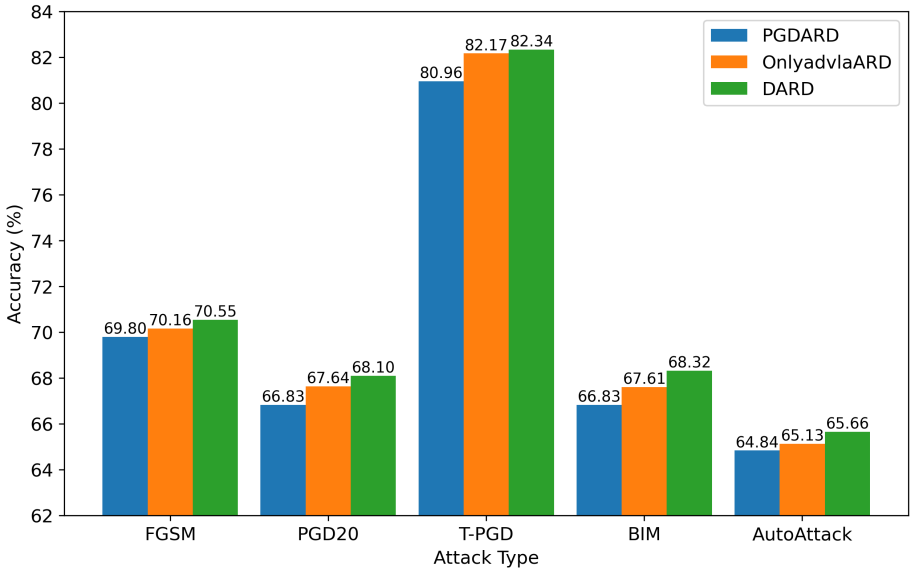


Fig. 3. Based on ablation experiments using the ResNet-18 model, we compare the performance differences among three adversarial robustness distillation strategies. Specifically, the PGDARD method employs the conventional PGD algorithm to generate adversarial examples, which are then used for adversarial knowledge distillation. In contrast, the OnlyadvlaARD method solely utilizes the teacher model's soft outputs on adversarial examples as guidance for the student model. Our proposed *DARD* approach, on the other hand, leverages the *DPGD* algorithm to generate adversarial examples and integrates the teacher model's soft outputs from both clean and adversarial examples during the distillation process.

Effect of Combining Clean and Adversarial Soft Label. In addition, during adversarial robustness distillation, we used only the soft output of the teacher model on adversarial examples as soft labels for the student model to investigate whether incorporating the teacher model's soft output on both natural and adversarial examples is necessary. The ablation experimental results demonstrate that the *DARD* method leverages the guidance of the teacher model on both natural and adversarial examples, enabling it to capture the teacher model's behavior more comprehensively. As a result, *DARD* achieves superior and more balanced performance across both natural and adversarial examples.

Through these two ablation studies, we obtained a more comprehensive understanding of the crucial role played by *DPGD* in *DARD* and verified the effectiveness of incorporating the soft outputs of both natural and adversarial examples in enhancing model robustness. Based on the results shown in Fig. 3 and 4, we conclude that *DARD* exhibits optimal performance in terms of both training convergence and adversarial robustness across all ablation studies.

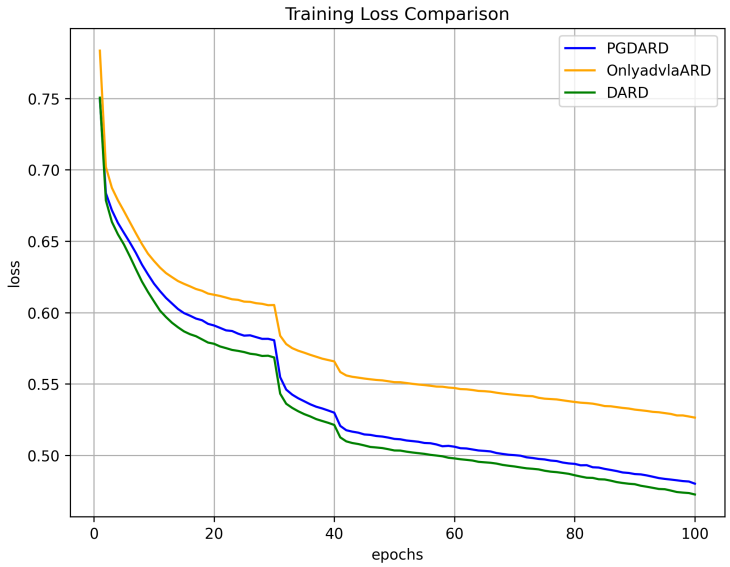


Fig. 4. The training loss with ResNet-18 student network distilled using PGDARD, OnlyadvlaARD, and DARD on CIFAR-10.

5 Conclusion

In this work, we presented *Dice Adversarial Robustness Distillation* (*DARD*), a novel knowledge distillation framework that simultaneously enhances the adversarial robustness of deep learning models and maintains high natural accuracy on clean data. A key insight from our study within *DARD* is the surprising effectiveness of adversarial-example-only training in achieving a superior equilibrium between standard and robust accuracy, providing a promising perspective for further research aimed at optimizing model performance across diverse operational conditions.

Our extensive empirical evaluations on *CIFAR-10* and *CIFAR-100* datasets demonstrated that *DARD* significantly outperforms traditional adversarial training baselines, including *SAT* and *ARD*, in various attack scenarios. These results underscore the efficacy of our approach in addressing the robustness-accuracy trade-off. Nevertheless, *DARD* does exhibit certain limitations, notably the computational demands associated with knowledge distillation and adversarial sample generation. Future research will focus on improving the efficiency of the distillation process and minimizing computational overhead without compromising robustness. Additionally, we plan to extend the applicability of the *DARD* framework by evaluating its performance on larger-scale datasets and in real-world adversarial settings, ensuring its relevance and efficacy across a broader range of applications. Moreover, exploring robustness against emerging stealthy threats such as clean-label backdoor attacks [21] is a critical direction for fu-

ture research, highlighting the importance of developing more comprehensive defensive framework.

References

1. Z. Zhao, S. Jie, J. Yu, Z. Sun, B. Zhang, and X. Wei, "Enhanced accuracy and robustness via multi-teacher adversarial distillation," in Proc. Eur. Conf. Comput. Vis. (ECCV), Cham, Switzerland: Springer Nature, 2022, pp. 585–602.
2. M. Goldblum, L. Fowl, S. Feizi, and T. Goldstein, "Adversarially robust distillation," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 4, pp. 3996–4003, Apr. 2020, doi: 10.1609/aaai.v34i04.5816.
3. J. Gu, H. Zhao, V. Tresp, and P. H. Torr, "SegPGD: An effective and efficient adversarial attack for evaluating and boosting segmentation robustness," in Proceedings of the European Conference on Computer Vision (ECCV), Cham, Switzerland: Springer Nature, Oct. 2022, pp. 308–325.
4. H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in Proc. Int. Conf. Mach. Learn. (ICML), Long Beach, CA, USA, Jun. 2019, pp. 7472–7482.
5. S. Gowal, C. Qin, J. Uesato, T. Mann, and P. Kohli, "Uncovering the limits of adversarial training against norm-bounded adversarial examples," arXiv preprint arXiv:2010.03593, 2020.
6. J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in Proc. Int. Conf. Mach. Learn. (ICML), May 2019, pp. 1310-1320.
7. G. Hinton, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.
8. P. Mettes, M. Ghadimi Atigh, M. Keller-Ressel, J. Gu, and S. Yeung, "Hyperbolic deep learning in computer vision: A survey," International Journal of Computer Vision, pp. 1–25, 2024.
9. S. Chen, Y. Zhang, and Q. Yang, "Multi-task learning in natural language processing: An overview," ACM Computing Surveys, vol. 56, no. 12, pp. 1–32, Apr. 2024.
10. T. Okuyama, T. Gonsalves and J. Upadhay, "Autonomous Driving System based on Deep Q Learnig," 2018 International Conference on Intelligent Autonomous Systems (ICoIAS), Singapore, 2018, pp. 201-205, doi: 10.1109/ICoIAS.2018.8494053.
11. L. Chi, M. Msahli, Q. Zhang, H. Qiu, T. Zhang, G. Memmi, and M. Qiu, "Adversarial attacks on autonomous driving systems in the physical world: A survey," IEEE Trans. Intell. Vehicles, pp. 1–22, 2024, doi: 10.1109/TIV.2024.3484152.
12. H. Chen, H. Yan, X. Yang, H. Su, S. Zhao and F. Qian, "Efficient Adversarial Attack Strategy Against 3D Object Detection in Autonomous Driving Systems," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 11, pp. 16118-16132, Nov. 2024, doi: 10.1109/TITS.2024.3410038.
13. M. Surekha, A. K. Sagar and V. Khemchandani, "Adversarial Attack and Defense Mechanisms in Medical Imaging: A Comprehensive Review," 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), Greater Noida, India, 2024, pp. 1657-1661, doi: 10.1109/IC2PCT60090.2024.10486235.
14. J. Zou, S. Zhang, and M. Qiu, "Adversarial attacks on large language models," in International Conference on Knowledge Science, Engineering and Management, Singapore: Springer Nature Singapore, Jul. 2024, pp. 85–96.

15. A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, Apr. 30–May 3, 2018.
16. A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Technical Report TR-2009, University of Toronto, 2009.
17. I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014.
18. A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," arXiv preprint arXiv:1611.01236, 2016.
19. A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in Artificial Intelligence Safety and Security, R. V. Yampolskiy, Ed. Boca Raton, FL, USA: Chapman and Hall/CRC, 2018, pp. 99–112.
20. F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in Proceedings of the 37th International Conference on Machine Learning, vol. 119, pp. 2206–2216, Jul. 2020.
21. Y. Zeng, M. Pan, H. A. Just, L. Lyu, M. Qiu, and R. Jia, "Narcissus: A practical clean-label backdoor attack with limited information," in Proc. 2023 ACM SIGSAC Conf. Comput. Commun. Secur., Nov. 2023, pp. 771–785.
22. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
23. N. M. Gürel, X. Qi, L. Rimanic, C. Zhang, and B. Li, "Knowledge enhanced machine learning pipeline against diverse adversarial attacks," in Proceedings of the 38th International Conference on Machine Learning, vol. 139, pp. 3976–3987, Jul. 2021.
24. C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," arXiv preprint arXiv:1312.6199, 2013.
25. N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in Proc. IEEE Symposium on Security and Privacy (SP), May 2017, pp. 39–57.
26. S. M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: a simple and accurate method to fool deep neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2574–2582.
27. A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for Thin Deep Nets," arXiv preprint arXiv:1412.6550, 2014.
28. S. Zagoruyko and N. Komodakis, "Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer," arXiv preprint arXiv:1612.03928, 2016.
29. B. Zi, S. Zhao, X. Ma, and Y. Jiang, "Revisiting adversarial robustness distillation: Robust soft labels make student better," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 16443–16452.
30. X. Yue, N. Mou, Q. Wang, and L. Zhao, "Revisiting adversarial robustness distillation from the perspective of robust fairness," Adv. Neural Inf. Process. Syst., vol. 36, pp. 30390–30401, 2023.