ADAPTIVE DECISION-BOUNDARY PROBING FOR ROBUST DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Robust distillation has been shown to effectively transfer robustness from larger adversarially trained models to smaller student models. However, robust teacher models may not reliably classify all adversarial examples, and conventional distillation mostly do not explicitly account for the position of these examples relative to the teacher's decision boundary. In this work, we propose Adaptive Decision-Boundary Probing Distillation *(ADBD)*, a robust distillation framework that uses the teacher's classification behavior to construct sample-specific query points. Starting from an adversarial example generated through coordinated teacher-student information, we preserve its perturbation direction while adaptively scaling its magnitude. When the teacher misclassifies the adversarial example, the perturbation magnitude is reduced to obtain a query point closer to the teacher's decision boundary while remaining on the teacher-correct side. When the teacher correctly classifies the adversarial example, the perturbation is extended along the same direction to probe a farther region in which the teacher remains correct. The resulting query samples provide teacher supervision at locations selected according to the teacher's local decision-boundary behavior. We evaluate *ADBD* across multiple datasets and adversarial attacks. Our experimental results show that *ADBD* improves the adversarial robustness of distilled student models over existing robust distillation methods.

1 INTRODUCTION

It is well established that deep neural networks (DNNs) are susceptible to adversarial attacks (Szegedy et al., 2013; Goodfellow et al., 2014; Nguyen et al., 2015). Adversarial attacks introduce small, often imperceptible perturbations to input samples, causing the resulting adversarial examples to be misclassified. The existence of adversarial examples has called into question the practical deployment of deep learning models in safety-critical domains such as healthcare, defense, and finance. Consequently, considerable effort has been directed toward improving the robustness of DNNs to adversarial attacks.

Adversarial training has been shown to be one of the most effective methods for improving the robustness of DNNs to adversarial attacks (Goodfellow et al., 2016; Madry et al., 2017). Adversarial training is typically formulated as a min-max optimization problem in which adversarial examples obtained through inner loss maximization are used to train the model (Madry et al., 2017). The relative success of adversarial training has inspired several variants (Zhang et al., 2019; Wu et al., 2020; Ding et al., 2019; Fakorede et al., 2024). However, prior work has also highlighted a number of challenges associated with adversarially trained models. These include the exacerbation of robustness disparities across labeled classes (Xu et al., 2021; Fakorede et al., 2023), the natural-robust accuracy trade-off (Zhang et al., 2019), and the need for increased model capacity to achieve stronger robustness (Zhang et al., 2020), among others.

Adversarial robustness distillation, which involves transferring knowledge from a higher-capacity robust teacher model to a smaller student model, has been demonstrated to improve the robustness of smaller models beyond what may be achieved when such models are adversarially trained from scratch Goldblum et al. (2020); Zi et al. (2021); Zhu et al. (2021). The conventional setup for adversarial robustness distillation follows the teacher-student architecture commonly used in knowledge distillation, where the student leverages information from the teacher's logits, features,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

or both. Following the work of Goldblum et al. (2020), which demonstrated the effectiveness of transferring adversarial robustness through knowledge distillation, several variants have been proposed with varying degrees of success (Zi et al., 2021; Zhu et al., 2021; Jung et al., 2024; Yin et al., 2024; Dong et al., 2024). Many of these methods rely on matching the logits or intermediate representations of the student and teacher models. More recent approaches have explored adaptive teacher guidance, attribution-region alignment, intermediate adversarial samples, and indirect gradient matching, among other strategies (Huang et al., 2023; Yin et al., 2024; Lee et al., 2025).

Despite these advances, existing adversarial robustness distillation methods have only partially explored the rich geometric information available around the decision boundaries of robust teachers. Prior studies on adversarial training have made several observations about the local geometry of robust models (Zhang et al., 2020; Xu et al., 2021; Hameed & Buesser, 2022; Rade & Moosavi-Dezfooli, 2022; Xu et al., 2023). For example, Rade & Moosavi-Dezfooli (2022) argues that adversarial training may excessively enlarge margins along certain adversarial directions, while individual samples may exhibit substantially different distances to the learned decision boundary. Moreover, the evolution of these margins can itself be sample dependent, with the decision boundary moving away from some vulnerable examples while moving toward others (Xu et al., 2023).

These findings have primarily been investigated in the context of adversarial training. In adversarial robustness distillation, however, the classification behavior of an adversarially trained robust teacher could provide a useful signal for guiding how the perturbation magnitude of individual adversarial examples should be adjusted. Motivated by this perspective, we investigate whether the teacher's classification behavior on adversarial examples can be exploited to construct additional sample-specific training examples. Specifically, we preserve the original adversarial direction but adaptively increase or decrease the perturbation magnitude according to the teacher's classification behavior. The resulting auxiliary examples, called *query examples*, are then used alongside the original adversarial examples to provide additional robustness supervision during distillation.

We summarize our contributions as follows:

- We propose a novel set of teacher-guided auxiliary examples, called *query examples*, obtained by adaptively modifying the perturbation magnitude of the original adversarial examples generated during the inner maximization while preserving their initial adversarial directions. The adaptive modification is guided by probing the teacher's classification behavior relative to its local decision boundary.
- We propose augmenting the supervisory signals available to the student during robust distillation by transferring the teacher's predictions on the resulting *query examples*.
- We empirically demonstrate the effectiveness of our method, termed *Adaptive Decision-Boundary Probing Distillation (ADBD)*, across multiple datasets and adversarial attacks, showing improved adversarial robustness over existing robust distillation methods.

2 Related Work

2.1 Adversarial Robustness.

Adversarial robustness refers to a model's ability to withstand adversarial attacks. Various approaches Madry et al. (2017); Goodfellow et al. (2014); Zhang et al. (2019) were proposed to improve the adversarial robustness of neural networks. However, some of these methods are ineffective against stronger attacks. Adversarial training Madry et al. (2017), which involves training the model with adversarial examples obtained under a worst-case loss, has significantly improved robustness. Adversarial training solves the following min-max optimization problem:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\max_{\mathbf{x}'\in B_\epsilon(\mathbf{x})} L(f_{\boldsymbol{\theta}}(\mathbf{x}'), y) \right]. \tag{1}$$

Here, y is the true label of input $\mathbf{x}$, $L(\cdot)$ represents the loss function, $\boldsymbol{\theta}$ denotes the model parameters, and $B_\epsilon(\mathbf{x}) = \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x}' - \mathbf{x}\|_p \leq \epsilon\}$ represents the ℓ_p-norm ball centered around $\mathbf{x}$ with radius ϵ. In Eq. (1), the inner maximization seeks a worst-case adversarial version of $\mathbf{x}$ that maximizes the loss, while the outer minimization optimizes the model parameters to minimize this worst-case adversarial loss.

2.2 Robust Distillation

While adversarial training is among the most effective approaches for improving the robustness of deep neural networks, achieving high robust performance can be hard to obtain with smaller compact models. Adversarial robustness distillation addresses this limitation by transferring robustness from a high-capacity robust teacher to a smaller student. Goldblum et al. (2020) introduced adversarially robust distillation and showed that robustness obtained by a large adversarially trained teacher can be transferred to a smaller student, with distilled students outperforming adversarially trained models of comparable capacity. Zi et al. (2021) further showed that exploiting the soft labels produced by a robust teacher may provide useful supervision for the student on both natural and adversarial examples and improve robust distillation.

However, a robust teacher may not be reliable on every adversarial example generated against the student. A growing body of work has consequently focused on making teacher guidance adaptive. Huang et al. (2023) proposed *Adaptive Adversarial Distillation (AdaAD)*, which addresses the fixed teacher supervision used by earlier min-max distillation methods. Rather than treating teacher information as fixed during adversarial-example generation, *AdaAD* incorporates the teacher into the inner optimization so that teacher and student interact when determining the adversarial example. This allows the optimization target itself to adapt as the student evolves. Other methods adapt the supervision according to teacher errors or sample characteristics. For example, Park & Min (2024) proposed a *Dynamic Guidance Adversarial Distillation (DGAD)*, which partitions samples according to misclassification behavior, corrects unreliable teacher labels, and imposes consistency between clean and adversarial student predictions. The method explicitly recognizes that different samples should receive different forms of teacher guidance. Jung et al. (2024) approached teacher reliability from a different perspective in *PeerAiD*, arguing that a pretrained robust teacher may perform poorly on adversarial examples specifically generated against the evolving student. *PeerAiD* therefore jointly trains a peer network to specialize in the adversarial examples produced for the student, rather than relying exclusively on a fixed pretrained teacher.

Other methods enrich the type of knowledge transferred. Yin et al. (2024) proposed *SmaraAD*, arguing that strict KL-based output matching may be insufficient since teacher and student capacities differ substantially. *SmaraAD* combines relaxed output matching with attribution-region alignment so that the student learns where the robust teacher focuses in addition to matching its predictions. Lee et al. (2025) pursue an even more explicitly geometric signal. Instead of relying only on logits, their method transfers information associated with the teacher's input gradients, which describe the local sensitivity of the prediction to changes in the input. The method provides another indication that robustness transfer can benefit from information about how teacher predictions vary locally, rather than merely their values at a fixed input.

Decision-boundary information is particularly relevant because adversarial robustness is fundamentally related to the perturbation required to change a model's prediction. Recent work has begun incorporating such information explicitly into robust distillation. Yang et al. (2025) proposed an approach that generate samples near a student's decision boundary, and utilized in the later stages of distillation process. The goal, according to the authors, is to encourage the student prediction on boundary samples to be consistent with the teacher's prediction.

Like Yang et al. (2025), our method explores decision-boundary information, but from a different perspective. Our method probes the teacher's classification behavior along the adversarial direction obtained from the inner maximization. Query examples are generated based on whether the teacher correctly classifies the adversarial example. When the teacher remains correct, the perturbation magnitude is increased to further probe its competence along that direction. When the teacher misclassifies the adversarial example, the perturbation magnitude is reduced until a query example that the teacher correctly classifies is found. The resulting query examples are then used to provide additional supervision to the student.

3 Notations and Preliminaries

We use bold letters to represent vectors. We denote by $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ a dataset of input feature vectors $\mathbf{x}_i \in \mathcal{X} \subseteq \mathbb{R}^d$ and labels $y_i \in \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ represent the feature space and label set, respectively.

Let $f_\theta : \mathcal{X} \to \mathbb{R}^C$ denote a deep neural network (DNN) classifier with parameters θ, where C is the number of output classes. For any $\mathbf{x} \in \mathcal{X}$, the predicted class is $F_\theta(\mathbf{x}) = \arg\max_k f_\theta(\mathbf{x})_k$, where $f_\theta(\mathbf{x})_k$ denotes the k-th output of the network. When $f_\theta(\mathbf{x})$ represents logits, we denote the corresponding class probabilities by $p_\theta(\mathbf{x}) = \mathrm{softmax}(f_\theta(\mathbf{x}))$, where $p_\theta(\mathbf{x})_y$ is the predicted probability that $\mathbf{x}$ belongs to class y.

We denote by $\|\cdot\|_p$ the ℓ_p-norm over $\mathbb{R}^d$, where $\|\mathbf{x}\|_p = (\sum_{i=1}^d |x_i|^p)^{1/p}$. An ϵ-neighborhood of $\mathbf{x}$ is defined as $B_\epsilon(\mathbf{x}) = \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x}' - \mathbf{x}\|_p \leq \epsilon\}$.

An adversarial example corresponding to a natural input $\mathbf{x}$ is denoted by $\mathbf{x}'$. We use the terms *robust model* and *adversarially trained model* interchangeably to refer to a DNN trained using adversarial examples. The student and teacher models are denoted by $F_{\theta_{\text{std}}}(\mathbf{x})$ and $F_{\theta_{\text{teach}}}(\mathbf{x})$, respectively.

Logit margin. For a sample $\mathbf{x}$ with class label y, we define the logit margin as $\phi_\theta^y(\mathbf{x}) = f_\theta(\mathbf{x})_y - \max_{y' \neq y} f_\theta(\mathbf{x})_{y'}$. A positive logit margin indicates that class y has the largest logit, while $\phi_\theta^y(\mathbf{x}) = 0$ characterizes a decision-boundary point between class y and its strongest competing class.

Adversarial direction. Given a natural input $\mathbf{x}$ and its adversarial counterpart $\mathbf{x}'$, we define the adversarial perturbation as $\mathbf{r} = \mathbf{x}' - \mathbf{x}$. Points along the same adversarial direction are parameterized as $\mathbf{x}(\alpha) = \mathbf{x} + \alpha\mathbf{r}$, where α controls the perturbation magnitude.

4 Proposed Method

Here, we motivate and present our proposed distillation approach. Adversarial robustness distillation typically trains a student using information from a robust teacher as supervisory signals. Most existing approaches primarily exploit the teacher's output predictions, intermediate features, or related representations to guide the student. However, adversarially-trained models possess additional characteristics that could provide useful supervisory information beyond their outputs and features. For instance, adversarial training is known to induce distinctive decision-boundary geometry (Xu et al., 2023) and sample-dependent robustness behavior (Xu et al., 2021; Fakorede et al., 2023).

4.1 Decision boundary and sample-wise robustness

Compared with standard-trained models, robust models obtained through adversarial training can exhibit substantially larger margins in certain directions (Rade & Moosavi-Dezfooli, 2022). These increases are not necessarily confined to the directions emphasized by the adversarial perturbations used during training, suggesting that adversarial training can reshape the decision boundary more broadly than the perturbation directions alone might suggest. It is also noteworthy that the distance to the decision boundary can vary considerably across samples, even for a robust classifier. Consequently, using a fixed perturbation budget ϵ does not necessarily produce adversarial examples that occupy comparable positions relative to the learned decision boundary (Xu et al., 2021).

For samples whose margins are substantially smaller than ϵ, the corresponding ϵ-bounded perturbation region can extend beyond the decision boundary. In contrast, for samples with margins larger than ϵ, the same perturbation budget may be insufficient to reach the boundary. Existing works have investigated sample-specific margins and the relationship between adversarial perturbations and decision-boundary geometry to better understand variations in local robustness (Liu et al., 2021; Xu et al., 2023). These observations suggest that the position of an adversarial example relative to the decision boundary may provide useful information beyond the adversarial example itself. We hypothesize that such boundary-relative information can provide additional supervision within a robust knowledge distillation framework.

4.1.1 Selective Probing of the Local Decision Boundary

In addition to learning from the predictions of a teacher model, a student may benefit from information about the local geometry of the teacher's decision boundary, particularly around natural and adversarial examples that the teacher classifies correctly. Although exactly characterizing the decision boundary is generally difficult, probing its local geometry using carefully selected samples provides a practical alternative.

We rely on the adversarial direction $\mathbf{r}$ defined in Sec. 3, where $\mathbf{r} = \mathbf{x}' - \mathbf{x}$. The perturbation $\mathbf{r}$ captures both the direction and magnitude of the change from the natural example to its adversarial counterpart. While $\mathbf{r}$ is obtained subject to a predefined perturbation constraint, the direction identified by the attack also provides a meaningful trajectory along which the behavior of the teacher can be examined. To explore this trajectory, we consider

$$\mathbf{x}^q(\alpha) = \mathbf{x} + \alpha\mathbf{r}. \tag{2}$$

Here, α controls the magnitude of the perturbation while preserving its direction. The original adversarial example corresponds to $\alpha = 1$. Values $0 < \alpha < 1$ place $\mathbf{x}^q$ between the natural example $\mathbf{x}$ and the adversarial example $\mathbf{x}'$, whereas $\alpha > 1$ extends the perturbation beyond $\mathbf{x}'$ along the same adversarial direction. Thus, varying α provides a simple means of probing the behavior of a robust teacher $f_{\boldsymbol{\theta}_{teach}}$ along an attack-induced direction. In particular, we examine the teacher's logit margin as α varies. When a change in the sign of the teacher's logit margin is bracketed along this trajectory, we use bisection search to approximate the corresponding value of α, thereby identifying a point near the teacher's local decision boundary along the adversarial direction.

Our directional probing shares a geometric similarity with max-margin adversarial methods such as MMA (Ding et al., 2019), which search for the shortest adversarial perturbation that reaches the decision boundary. However, the objectives are different. MMA uses the estimated boundary location to maximize the sample-wise margin of the model being trained. In contrast, we probe the boundary of a pretrained robust teacher and use the resulting boundary-aware query point as an input for robust knowledge distillation. Thus, the boundary search is not itself the training objective, but a mechanism for selecting informative locations from which teacher knowledge is transferred to the student.

4.1.2 Sample-Dependent Boundary Probing

Existing works have shown that the effect of adversarial training is not uniform across samples, even when samples are perturbed using the same perturbation budget. In particular, different examples may exhibit substantially different distances to the decision boundary. Consequently, the same perturbation magnitude can have different geometric implications across samples.

We therefore adapt α on a sample-wise basis. We categorize the samples into three groups: (i) *adversarial examples that are correctly classified by the teacher;* (ii) *adversarial examples that are misclassified by the teacher, while their corresponding natural examples are correctly classified;* and (iii) *adversarial examples for which both the adversarial and corresponding natural examples are misclassified by the teacher*.

In the first group, where the teacher correctly classifies the adversarial example, we search for $\alpha > 1$, thereby increasing the magnitude of $\mathbf{r}$ while preserving its direction. The goal is to find the largest α for which $\mathbf{x}^q$ remains correctly classified by the teacher. This provides information about how far the teacher's correct decision region extends along the adversarial direction. Formally, we seek

$$\alpha^* = \max_{\alpha \in [1, \alpha_{\max}]} \alpha \quad \text{subject to} \quad \arg\max_k f_{\boldsymbol{\theta}_{\text{teach}}}(\mathbf{x} + \alpha\mathbf{r})_k = y. \tag{3}$$

Equivalently, using the teacher's true-class logit margin $\phi^y_{\theta_{teach}}$, we seek

$$\alpha^* = \max_{\alpha \in [1, \alpha_{\max}]} \alpha \quad \text{subject to} \quad \phi^y_{\theta_{\text{teach}}}(\mathbf{x} + \alpha\mathbf{r}) \geq 0. \tag{4}$$

The resulting query is then given by $\mathbf{x}^q = \mathbf{x} + \alpha^*\mathbf{r}$.

For the second group, where the teacher misclassifies the adversarial example but correctly classifies its natural counterpart, we search over $0 \leq \alpha \leq 1$. Here, $\alpha = 0$ gives $\mathbf{x}^q = \mathbf{x}$, while $\alpha = 1$ gives $\mathbf{x}^q = \mathbf{x}'$. The goal is to find the maximum value of α such that the teacher does not misclassify $\mathbf{x}^q(\alpha)$. This places the query as far as possible along the adversarial direction while remaining on the teacher-correct side of the decision boundary. Formally, we seek:

$$\alpha^* = \max_{\alpha \in [0, 1]} \alpha \quad \text{subject to} \quad \arg\max_k f_{\boldsymbol{\theta}_{\text{teach}}}(\mathbf{x} + \alpha\mathbf{r})_k = y. \tag{5}$$

Our approach refrains from adjusting α for samples where both the adversarial example and its corresponding natural example are misclassified by the teacher, since a reliable competence-guided search interval is unavailable. We therefore consider two treatments for these samples. The first treatment, ***ADBD-v1***, excludes these samples from teacher-guided distillation, avoiding supervision from a teacher that is incorrect at both endpoints. Intuitively, this treatment is akin to a student refraining from learning from a teacher who lacks reliable knowledge about certain examples. The second retains the adversarial point and directly distills the teacher's soft prediction, allowing the student to preserve dark knowledge and broader decision-structure information contained in the teacher's output distribution. We term this ***ADBD-v2***.

4.2 Robust Distillation Objective

Here, we present our *ADBD* distillation objective. The proposed approach is adaptive, as it relies on the sample-dependent query examples $\mathbf{x}^q$ defined in Eq. (2). Each $\mathbf{x}^q$ is constructed based on the teacher's classification behavior on the corresponding natural and adversarial examples.

The optimization consists of two steps: an inner maximization and an outer minimization. Following Huang et al. (2023), the inner maximization solves

$$\mathbf{x}' = \underset{\tilde{\mathbf{x}} \in B_\epsilon(\mathbf{x})}{\operatorname{argmax}} D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\tilde{\mathbf{x}}) \parallel f_{\boldsymbol{\theta}_{teach}}(\tilde{\mathbf{x}}))\,, \tag{6}$$

where D_{KL} denotes the Kullback–Leibler divergence between the output distributions of the standard and teacher models. The inner maximization yields the adversarial example $\mathbf{x}'$. The corresponding adversarial perturbation is then given by $\mathbf{r} = \mathbf{x}' - \mathbf{x}$.

The goal of the outer minimization is to encourage the student model to inherit both the predictive behavior of the teacher and useful information about the geometry of the teacher's decision boundary. The outer minimization is defined as

$$\mathcal{L}_{\mathrm{ADBD}} = D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\mathbf{x}^q), f_{\boldsymbol{\theta}_{teach}}(\mathbf{x}^q)) + D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\mathbf{x}'), f_{\boldsymbol{\theta}_{std}}(\mathbf{x})) \tag{7}$$

The first term distills the teacher's prediction at $\mathbf{x}^q$. Recall that $\mathbf{x}^q$ is obtained by adaptively scaling the adversarial perturbation according to the teacher's classification behavior along the adversarial direction. Consequently, these query examples provide supervision at locations that reflect information about the local decision geometry of the teacher.

The second term matches the student's prediction on the adversarial example $\mathbf{x}'$ with its corresponding prediction on the natural example $\mathbf{x}$. This term complements the teacher supervision provided by the first term by allowing the student to also learn from its own predictions. In this way, the student's prediction on the natural example provides an additional supervisory signal for learning a consistent prediction on its adversarial counterpart.

The pseudocode for the *ADBD* approach is provided in Appendix 1.

5 Experiments

In this section, we describe the experimental settings and compare our approach with existing state-of-the-art adversarial robustness distillation methods.

5.1 Implementation Details

Datasets. We evaluate our method on three standard image classification datasets: CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and SVHN (Netzer et al., 2011).

Teacher and Student Models. For CIFAR-10 and CIFAR-100, we adopt two robust teacher models, *LTD* (Chen & Lee, 2026) and *SCORE* (Pang et al., 2022). The choice of these teacher models follows prior adversarial robustness distillation works (Dong et al., 2024; Lee et al., 2025). Both teacher models are publicly available through *RobustBench* (Croce et al., 2020). For SVHN, we trained the teacher model from scratch using *TRADES* approach (Zhang et al., 2019), following the experimental setting of (Lee et al., 2025). The performance of the teacher models is reported in

Table 1: The performance (accuracy %) of teacher models for CIFAR-10, CIFAR-100, and SVHN datasets.

DATASET	TEACHER	ARCHITECTURE	NATURAL	PGD	AA
CIFAR-10	LTD (CHEN & LEE, 2026)	WIDERESNET-34-10	85.21	60.89	56.94
	SCORE (PANG ET AL., 2022)	WIDERESNET-28-10	88.61	64.95	61.03
CIFAR-100	LTD (CHEN & LEE, 2026)	WIDERESNET-34-10	64.07	36.61	30.57
	SCORE (PANG ET AL., 2022)	WIDERESNET-28-10	63.64	35.46	31.13
SVHN	TRADES (ZHANG ET AL., 2019)	WIDERESNET-34-10	92.93	62.82	55.98

Table 1. In line with prior works, we adopt smaller model architectures for the student models. Specifically, we use ResNet-18 and MobileNet-v2 (Sandler et al., 2018) for CIFAR-10 and CIFAR-100, and SVHN.

Implementation Details. The adversarial samples obtained in the inner maximization are generated based on the ℓ_∞-norm threat model with perturbation radius $\epsilon = 8/255$, step size $2/255$, and 10 iteration steps. The samples are generated using the formulation in Eqn. 6. We train the model using mini-batch gradient descent for 200 epochs, with momentum 0.9 and a batch size of 240. We use a weight decay of 5×10^{-4} for CIFAR-10/100 and SVHN. The initial learning rate is set to 0.1 and divided by 10 at the 90th, 150th, and 170th epochs.

Hyperparameters. While the exact value of α for the majority of individual query samples used for distillation is computed using the bisection method, the maximum value $\alpha_{\max}$ is set to 2.0. We use five bisection steps for our experiments on all datasets.

5.2 EVALUATION METRICS

We evaluate the robustness of the proposed method against strong white-box attacks, where the attacker has access to the student model parameters. For CIFAR-10/100 and SVHN using ResNet-18, we employ PGD-20 with $\epsilon = 8/255$, step size $\kappa = 2/255$, and $K = 20$ iterations. We also evaluate the student model against the CW attack, implemented using the CW loss (Carlini & Wagner, 2017) and optimized with PGD-20 using $\epsilon = 8/255$ and step size $2/255$. In addition, we evaluate against AutoAttack (AA) (Croce & Hein, 2020b), a strong ensemble of attacks consisting of APGD-CE, APGD-T (Croce & Hein, 2020b), FAB-T (Croce & Hein, 2020a), and Square, a strong black-box attack (Andriushchenko et al., 2020).

5.3 BASELINES

We compare our method with prominent robust distillation approaches, including *ARD* (Goldblum et al., 2020), *IAD* (Zhu et al., 2021), *RSLAD* (Zi et al., 2021), *AKD* (Maroto et al., 2022), *AdaAD* (Huang et al., 2023), and *Smara* (Yin et al., 2024). In addition, we compare against standard adversarial training methods, including *PGD-AT* (Madry et al., 2017) and *TRADES* (Zhang et al., 2019). For each baseline, we follow the method-specific hyperparameters reported in the corresponding original paper. To ensure a consistent experimental setting, however, we use the same learning rate, batch size, and weight decay across all methods as those used to train our proposed method.

5.4 PERFORMANCE EVALUATIONS

We summarize our results on CIFAR-10 and CIFAR-100 in Tables 2 and 3, respectively. Further results on the SVHN dataset, along with ablation studies on each term in our training objective, are provided in the Appendix in Tables 4 and 5, respectively.

It can be observed from the tables that adversarial distillation methods generally outperform adversarially trained models when using smaller models such as RN-18. Our proposed *ADBD-v1* and *ADBD-v2* also outperform these adversarial training baselines by a considerable margin. For instance, on CIFAR-10, *ADBD-v1* and *ADBD-v2*, distilled from *SCORE* (Pang et al., 2022), outperform *PGD-AT* under PGD by more than 9%, while *ADBD-v2* outperforms *TRADES* by almost 7%. Performance improvements can also be observed across the other robust evaluation metrics.

In addition to outperforming traditional adversarial training methods, *ADBD* also improves upon existing adversarial distillation methods on CIFAR-10 against *PGD*, *C&W*, and *AA*. For a fair comparison, both the baseline methods and our proposed methods are distilled from the same robust teachers.

Results in Table 2 show the effectiveness of *ADBD-v1* and *ADBD-v2* for robust distillation across the two teachers provided in Table 1, for both *RN-18* and *MN-v2*. *ADBD-v1* achieves a robust accuracy of 54.76% under AA, exceeding the other reported distillation baselines for *RN-18*. Similarly, *ADBD-v2* achieves a PGD robust accuracy of 59.17% on *RN-18*, which is also higher than the other reported distillation baselines. The improvements of our methods are also evident for *MN-v2*. In particular, *ADBD-v1* and *ADBD-v2* outperform *AdaAD* and *SmaraAD* under *C&W* by more than 1.5% and 1.7%, respectively.

Results in Table 3 also indicate that *ADBD* improves robustness on CIFAR-100 when applied to *RN-18* and *MN-v2* across the reported attacks.

Table 2: Adversarial distillation results using two teacher and two student models on CIFAR-10. Natural, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds.

Model	Method	LTD (Chen & Lee, 2026)				SCORE (Pang et al., 2022)			
		Natural	PGD	C&W	AA	Natural	PGD	C&W	AA
ResNet-18	PGD-AT	83.22	49.81	48.65	46.91	83.22	49.81	48.65	46.91
	TRADES	82.91	52.29	50.31	49.10	82.91	52.29	50.31	49.10
	ARD	83.89	53.09	50.32	48.95	84.35	53.27	51.58	49.40
	IAD	83.21	53.21	51.11	48.92	83.57	53.45	51.73	49.09
	RSLAD	83.38	55.95	53.07	51.93	84.11	54.52	53.39	51.36
	AKD	**85.05**	52.62	49.46	48.21	**86.03**	55.92	54.11	51.94
	AdaAD	84.53	56.76	53.55	52.56	85.41	56.91	54.01	52.90
	SmaraAD	84.59	56.35	54.75	52.64	86.11	57.42	55.14	52.55
	ADBD-v1	84.19	58.50	**56.16**	**54.31**	85.41	58.92	**56.54**	**54.76**
	ADBD-v2	84.21	**58.73**	56.09	54.26	85.65	**59.17**	56.42	54.62
MobileNetV2	PGD-AT	80.13	49.11	48.18	45.62	80.13	49.11	48.18	45.62
	TRADES	79.98	51.49	50.3	46.97	79.98	51.49	50.3	46.97
	ARD	82.34	51.96	49.08	48.41	83.51	52.11	50.61	49.31
	IAD	83.21	52.78	51.11	48.89	84.19	53.23	51.42	49.09
	RSLAD	81.91	52.35	49.02	47.29	82.80	52.29	49.85	47.94
	AKD	*84.23*	50.89	49.78	46.94	84.34	51.80	50.26	48.15
	AdaAD	84.13	54.78	51.67	49.39	**84.91**	55.04	51.92	50.15
	SmaraAD	83.61	54.85	52.65	51.57	84.86	55.72	52.41	51.84
	ADBD-v1	83.11	57.55	**54.57**	**53.00**	84.16	57.23	**54.45**	**53.05**
	ADBD-v2	83.22	**57.57**	54.41	52.85	84.32	**57.55**	54.31	52.70

5.5 Comparison of *ADBD-v1* and *ADBD-v2*

The main distinction between the two implementations is the treatment of data points whose natural and corresponding adversarially perturbed variants are both misclassified by the teacher. *ADBD-v1* abstains from distilling from these samples, since the teacher may lack useful supervisory signals that can be transferred to the student. In contrast, *ADBD-v2* retains these examples for distillation.

Tables 2 and 3 show that these treatments produce slightly different results. The exclusion of these samples does not degrade the overall performance of *ADBD-v1*. Compared with *ADBD-v2*, *ADBD-v1* performs better under *AA*, an ensemble of strong attacks. However, *ADBD-v2* performs better in terms of natural accuracy and robustness to *PGD*. We observe this behavior particularly on the CIFAR-10 and SVHN datasets. On CIFAR-100, the performances of *ADBD-v1* and *ADBD-v2* are comparatively similar.

Table 3: Adversarial distillation results using two teacher and two student models on CIFAR-100. Natural, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds.

Model	Method	TLD (Chen & Lee, 2026)				SCORE (Pang et al., 2022)			
		Natural	PGD	C&W	AA	Natural	PGD	C&W	AA
ResNet-18	PGD-AT	56.27	25.81	24.04	21.17	56.27	25.81	24.04	21.17
	TRADES	55.76	25.75	23.92	22.12	55.76	25.75	23.92	22.12
	ARD	60.95	31.21	27.69	25.71	60.11	31.43	27.97	26.04
	IAD	60.12	32.83	28.35	26.27	59.67	31.97	28.79	27.03
	RSLAD	59.95	32.39	28.75	26.81	60.05	32.73	29.37	27.52
	AKD	60.19	31.26	26.32	25.11	60.74	31.35	28.81	26.78
	AdaAD	**62.16**	33.24	29.69	27.42	**62.46**	32.87	29.19	27.37
	SmaraAD	61.49	32.50	28.84	27.16	61.16	31.82	28.57	27.27
	ADBD-v1	61.63	34.52	30.46	28.88	61.29	33.67	30.23	**29.02**
	ADBD-v2	61.78	**34.60**	**30.49**	**28.96**	61.51	**33.78**	**30.67**	28.81
MobileNetV2	PGD-AT	59.34	24.09	23.03	21.59	59.34	24.09	23.03	21.59
	TRADES	52.12	24.95	23.45	21.12	52.12	24.95	23.45	21.12
	ARD	60.53	30.84	27.85	25.28	59.94	30.92	28.03	25.77
	IAD	56.09	29.54	26.42	24.22	56.88	30.54	27.51	25.60
	RSLAD	60.43	32.39	28.89	26.86	59.91	31.67	28.57	26.56
	AKD	60.68	29.81	26.11	24.82	59.39	30.17	27.57	25.51
	AdaAD	61.45	30.61	26.67	25.03	61.27	32.35	29.21	27.38
	SmaraAD	62.13	31.67	28.60	26.79	61.11	31.48	27.61	27.01
	ADBD-v1	61.05	33.52	30.01	**28.45**	60.95	33.21	30.17	**28.55**
	ADBD-v2	61.56	**33.61**	**30.11**	28.41	61.45	**33.56**	**30.25**	28.49

While the reason for this behavior is not entirely clear, the relatively better performance of *ADBD-v2* under *PGD*, together with its lower performance under *AA*, may suggest that retaining these doubly misclassified samples favors robustness against the training-aligned PGD attack more than robustness across a broader set of attacks. This observation raises questions about the usefulness of such samples for robust distillation. Our observation is also qualitatively consistent with the empirical findings of Dong et al. (2021), which suggest that low-quality training samples can contribute to robustness overestimation and that excluding such samples may help alleviate this effect.

6 Conclusion

In this paper, we argue that since robust teacher models may not reliably classify all adversarial examples, indiscriminately distilling from all examples may be inadequate. Therefore, we propose an adaptive decision-boundary probing distillation approach that leverages the teacher's classification behavior. Our method generates auxiliary examples, termed query examples. These sample-wise query examples are constructed by scaling the perturbation magnitude along the initial adversarial direction obtained from the inner maximization. For adversarial examples correctly classified by the teacher, the perturbation magnitude is further increased. For adversarial examples misclassified by the teacher, but whose corresponding natural examples are correctly classified, the perturbation magnitude is reduced to a point where the teacher can correctly classify the resulting query example. We also consider two different treatments for samples whose natural and corresponding adversarial examples are both misclassified by the teacher. Finally, we distill information from the teacher using the resulting query examples. Our extensive experiments across multiple datasets, teacher models, and student architectures validate the effectiveness of the proposed approach.

AI USE STATEMENT

In this work, we used generative AI tools for smoothing some of the sentences and paragraphs, and for the quick identification of punctuation or other errors. We have not used generative AI tools for generating any section or the empirical results published in our work. We carefully cross-checked, reviewed, and verified every piece of information provided by the generative tool we used. The use of the generative tool in this work is purely assistive. The content of this work represents our ideas.

REFERENCES

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

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

Erh-Chung Chen and Che-Rung Lee. Ltd: Low temperature distillation for gradient masking-free adversarial training. *ACM Transactions on Cyber-Physical Systems*, 10(2):1–22, 2026.

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

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

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. Mma training: Direct input space margin maximization through adversarial training. In *International Conference on Learning Representations*, 2019.

Chengyu Dong, Liyuan Liu, and Jingbo Shang. Data quality matters for adversarial training: An empirical study. *arXiv preprint arXiv:2102.07437*, 2021.

Junhao Dong, Piotr Koniusz, Junxi Chen, and Yew-Soon Ong. Adversarially robust distillation by reducing the student-teacher variance gap. In *European Conference on Computer Vision*, pp. 92–111. Springer, 2024.

Olukorede Fakorede, Ashutosh Kumar Nirala, Modeste Atsague, and Jin Tian. Vulnerability-aware instance reweighting for adversarial training. *arXiv preprint arXiv:2307.07167*, 2023.

Olukorede Fakorede, Modeste Atsague, and Jin Tian. Standard-deviation-inspired regularization for improving adversarial robustness. *arXiv preprint arXiv:2412.19947*, 2024.

Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 3996–4003, 2020.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

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

Muhammad Zaid Hameed and Beat Buesser. Boundary adversarial examples against adversarial overfitting. *arXiv preprint arXiv:2211.14088*, 2022.

Bo Huang, Mingyang Chen, Yi Wang, Junda Lu, Minhao Cheng, and Wei Wang. Boosting accuracy and robustness of student models via adaptive adversarial distillation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24668–24677. IEEE, 2023.

Jaewon Jung, Hongsun Jang, Jaeyong Song, and Jinho Lee. Peeraid: Improving adversarial distillation from a specialized peer tutor. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24482–24491. IEEE, 2024.

Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

Hongsin Lee, Seungju Cho, and Changick Kim. Indirect gradient matching for adversarial robust distillation. In *International Conference on Learning Representations*, volume 2025, pp. 70007–70028, 2025.

Feng Liu, Bo Han, Tongliang Liu, Chen Gong, Gang Niu, Mingyuan Zhou, Masashi Sugiyama, et al. Probabilistic margins for instance reweighting in adversarial training. *Advances in Neural Information Processing Systems*, 34:23258–23269, 2021.

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.

Javier Maroto, Guillermo Ortiz-JimÃŠnez, and Pascal Frossard. On the benefits of knowledge distillation for adversarial robustness. *arXiv preprint arXiv:2203.07159*, 2022.

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

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Tianyu Pang, Min Lin, Xiao Yang, Jun Zhu, and Shuicheng Yan. Robustness and accuracy could be reconcilable by (proper) definition. In *International conference on machine learning*, pp. 17258–17277. PMLR, 2022.

Hyejin Park and Dongbo Min. Dynamic guidance adversarial distillation with enhanced teacher knowledge. In *European Conference on Computer Vision*, pp. 204–219. Springer, 2024.

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

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 4510–4520. Ieee, 2018.

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.

Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33, 2020.

Han Xu, Xiaorui Liu, Yaxin Li, Anil Jain, and Jiliang Tang. To be robust or to be fair: Towards fairness in adversarial training. In *International Conference on Machine Learning*, pp. 11492–11501. PMLR, 2021.

Yuancheng Xu, Yanchao Sun, Micah Goldblum, Tom Goldstein, and Furong Huang. Exploring and exploiting decision boundary dynamics for adversarial robustness. *arXiv preprint arXiv:2302.03015*, 2023.

Fengyu Yang, Hong Yan, Yu Xiong, Xiaohui Wei, and Yihui Zhong. Dynamic boundary guided adversarial robust distillation to improve robustness. *Engineering Applications of Artificial Intelligence*, 159:111228, 2025.

Shenglin Yin, Zhen Xiao, Mingxuan Song, and Jieyi Long. Adversarial distillation based on slack matching and attribution region alignment. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24605–24614. IEEE, 2024.

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*, pp. 7472–7482. PMLR, 2019.

Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*, 2020.

Jianing Zhu, Jiangchao Yao, Bo Han, Jingfeng Zhang, Tongliang Liu, Gang Niu, Jingren Zhou, Jianliang Xu, and Hongxia Yang. Reliable adversarial distillation with unreliable teachers. *arXiv preprint arXiv:2106.04928*, 2021.

Bojia Zi, Shihao Zhao, Xingjun Ma, and Yu-Gang Jiang. Revisiting adversarial robustness distillation: Robust soft labels make student better. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16423–16432. IEEE, 2021.

A Further Experiments

To further validate the effectiveness of our method, we conduct additional experiments on the SVHN dataset. The distillation results presented in this section involve the RN-18 student architecture. The teacher model uses a WideResNet-34-10 architecture and is trained on SVHN using the *TRADES* method (Zhang et al., 2019). The performance of the teacher model is reported in Table 1.

A.1 Teacher Implementation Details

We train the WideResNet-34-10 teacher model for 100 epochs using mini-batch stochastic gradient descent with a momentum of 0.9 and a batch size of 128. We use a weight decay of 5×10^{-4} and an initial learning rate of 0.1, which is divided by 10 at the 75th epoch and again at the 90th epoch. The SVHN samples are normalized using their channel-wise mean and standard deviation. The regularization hyperparameter for *TRADES* is set to 6.0.

A.2 Results

Our experimental results further validate the effectiveness of our method on RN-18 using the SVHN dataset. Both *ADBD-v1* and *ADBD-v2* outperform the baseline methods under PGD, C&W, and AA. In particular, both variants outperform *AdaAD* under PGD and C&W by more than 2%. They also achieve better robustness under AA. Consistent with our observation on CIFAR-10, *ADBD-v1* performs slightly better than *ADBD-v2* under *AA*, whereas the performance difference under *PGD* is negligible.

Table 4: Adversarial distillation results on SVHN using a TRADES-trained teacher and a ResNet-18 student. Natural, PGD, C&W, and AA columns report accuracy (%) under each evaluation setting. Results are averaged over three random seeds.

Method	**Natural**	**PGD**	**C&W**	**AA**
PGD-AT	91.71	53.67	50.76	45.41
TRADES	89.95	57.27	51.67	49.57
ARD	91.28	55.77	53.27	48.66
IAD	90.65	56.73	53.19	48.10
RSLAD	91.35	57.04	54.15	50.85
AKD	89.62	54.25	50.30	49.35
AdaAD	92.14	58.31	55.19	53.02
SmaraAD	92.15	59.52	55.67	53.09
ADBD-v1	90.92	61.61	**58.11**	**54.81**
ADBD-v2	91.02	**61.64**	57.80	54.49

B ABLATION STUDIES

The proposed *ADBD* objective consists of two terms: $D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\mathbf{x}^q), f_{\boldsymbol{\theta}_{teach}}(\mathbf{x}^q))$ and $D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\mathbf{x}'), f_{\boldsymbol{\theta}_{std}}(\mathbf{x}))$. The first term represents the main idea of the proposed approach, while the second term complements it by allowing all adversarial examples obtained from the inner maximization step to contribute to training and encouraging the student to learn from its predictions on natural examples.

In this section, we conduct an ablation study on these two terms, with the results presented in Table 5. Using only $D_{\mathrm{KL}}(f_{\boldsymbol{\theta}_{std}}(\mathbf{x}^q), f_{\boldsymbol{\theta}_{teach}}(\mathbf{x}^q))$ with the *SCORE* model as the teacher already produces strong robustness. Specifically, *ADBD-v2* without the second term achieves robust accuracies of 58.62%, 56.16%, and 54.29% under *PGD*, *C&W*, and *AA*, respectively. Similarly, *ADBD-v1* without the second term achieves 58.48%, 56.28%, and 54.47% under the same attacks. These results outperform the corresponding existing distillation baselines reported in Table 2, demonstrating that the query-based distillation term alone provides substantial robustness.

The introduction of the second term further provides consistent, although modest, improvements in both natural accuracy and robustness across the evaluated attacks. For *ADBD-v1*, the PGD, C&W, and AA accuracies improve from 58.48%, 56.28%, and 54.47% to 58.92%, 56.54%, and 54.76%, respectively. A similar trend is observed for *ADBD-v2*, where the corresponding accuracies improve from 58.62%, 56.16%, and 54.29% to 59.17%, 56.42%, and 54.62%.

Table 5: Ablation results for *ADBD* using an RN-18 student and the SCORE teacher on CIFAR-10.

Objective	**Natural**	**PGD**	**C&W**	**AA**
ADBD-v1 w/o second term	85.31	58.48	56.28	54.47
ADBD-v2 w/o second term	85.54	58.62	56.16	54.29
ADBD-v1	85.41	58.92	56.54	54.76
ADBD-v2	85.65	59.17	56.42	54.62

C MAIN ALGORITHM

Algorithm 1 ADBD-v2 Algorithm.

Input: two neural networks $f_{\theta_{std}}$ and $f_{\theta_{teach}}$ with parameters θ_{std} and θ_{teach} representing the student and teacher parameters, respectively, a training dataset $\mathcal{D}$ of size n, and hyperparameter α.

Output: a robust student model with parameters θ^*_{std}

for $epoch = 1$ to num_epochs **do**
  for $batch = 1$ to num_batches **do**
    Sample a mini-batch $\{(\mathbf{x}_i, y_i)\}_{i=1}^{M}$ from $\mathcal{D}$; ▷ mini-batch of size M
    for $i = 1$ to M **do**
      $\mathbf{x}'_i \leftarrow \mathbf{x}_i + 0.001 \cdot \mathcal{N}(0, I)$ ▷ $\mathcal{N}(0, I)$ is a Gaussian distribution
      for $t = 1$ to T **do**
        $\mathbf{x}'_i \leftarrow \Pi_{B_\epsilon(\mathbf{x}_i)} \left(\mathbf{x}'_i + \kappa \cdot \text{sign}\left(\nabla_{\mathbf{x}'_i} D_{\text{KL}}\left(f_{\theta_{std}}(\mathbf{x}'_i), f_{\theta_{teach}}(\mathbf{x}'_i)\right)\right)\right)$
      end for
      $\mathbf{r}_i \leftarrow \mathbf{x}'_i - \mathbf{x}_i$
      if $\arg\max_c f^c_{\theta_{teach}}(\mathbf{x}'_i) = y_i$ **then**
        Search $\alpha^*_i \in [1, 2]$ using bisection search
        while $\arg\max_c f^c_{\theta_{teach}}(\mathbf{x}_i + \alpha^*_i \mathbf{r}_i) = y_i$ **do**
          Update α^*_i using bisection search
        end while
        $\mathbf{x}^q_i \leftarrow \mathbf{x}_i + \alpha^*_i \mathbf{r}_i$
      else if $\arg\max_c f^c_{\theta_{teach}}(\mathbf{x}'_i) \neq y_i$ **and** $\arg\max_c f^c_{\theta_{teach}}(\mathbf{x}_i) = y_i$ **then**
        Search $\alpha^*_i \in [0, 1]$ using bisection search
        while $\arg\max_c f^c_{\theta_{teach}}(\mathbf{x}_i + \alpha^*_i \mathbf{r}_i) = y_i$ **do**
          Update α^*_i using bisection search
        end while
        $\mathbf{x}^q_i \leftarrow \mathbf{x}_i + \alpha^*_i \mathbf{r}_i$
      else
        $\mathbf{x}^q_i \leftarrow \mathbf{x}'_i$
      end if
    end for
    $\theta_{std} \leftarrow \theta_{std} - \eta \nabla_{\theta_{std}} \frac{1}{M} \sum_{i=1}^{M} \left[D_{\text{KL}}\left(f_{\theta_{std}}(\mathbf{x}^q_i), f_{\theta_{teach}}(\mathbf{x}^q_i)\right) + D_{\text{KL}}\left(f_{\theta_{std}}(\mathbf{x}'_i), f_{\theta_{std}}(\mathbf{x}_i)\right)\right]$
  end for
end for