Decoupled Kullback-Leibler Divergence Loss

Jiequan Cui[1] **Zhuotao Tian**[4] **Zhisheng Zhong**[2] **Xiaojuan Qi**[3] **Bei Yu**[2] **Hanwang Zhang**[1]
Nanyang Technological University[1] The Chinese University of Hong Kong[2]
The University of Hong Kong[3] HIT(SZ)[4]

Abstract

In this paper, we delve deeper into the Kullback–Leibler (KL) Divergence loss and mathematically prove that it is equivalent to the Decoupled Kullback-Leibler (DKL) Divergence loss that consists of 1) a weighted Mean Square Error ($\mathbf{w}$MSE) loss and 2) a Cross-Entropy loss incorporating soft labels. Thanks to the decomposed formulation of DKL loss, we have identified two areas for improvement. Firstly, we address the limitation of KL/DKL in scenarios like knowledge distillation by breaking its asymmetric optimization property. This modification ensures that the $\mathbf{w}$MSE component is always effective during training, providing extra constructive cues. Secondly, we introduce class-wise global information into KL/DKL to mitigate bias from individual samples. With these two enhancements, we derive the Improved Kullback–Leibler (IKL) Divergence loss and evaluate its effectiveness by conducting experiments on CIFAR-10/100 and ImageNet datasets, focusing on adversarial training, and knowledge distillation tasks. The proposed approach achieves new state-of-the-art adversarial robustness on the public leaderboard — *RobustBench* and competitive performance on knowledge distillation, demonstrating the substantial practical merits. Our code is available at `https://github.com/jiequancui/DKL`.

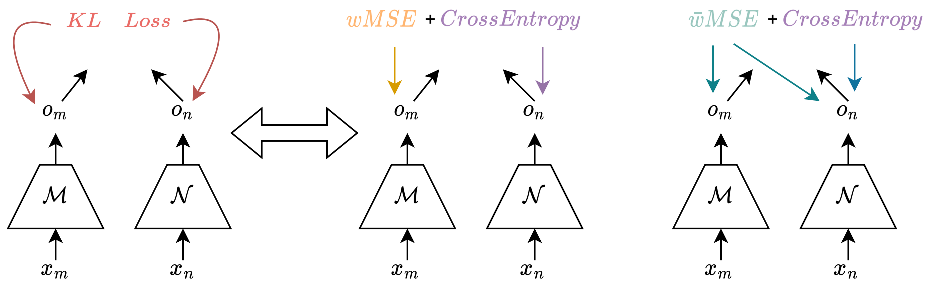
1 Introduction

Loss functions are a critical component of training deep models. Cross-Entropy loss is particularly important in image classification tasks [28, 55, 59, 20, 44, 12], while Mean Square Error (MSE) loss is commonly used in regression tasks [51, 27, 25]. Contrastive loss [7, 26, 8, 23, 4, 16, 17] has emerged as a popular objective for representation learning. The selection of an appropriate loss function can exert a substantial influence on a model's performance. Therefore, the development of effective loss functions [3, 43, 73, 63, 36, 2, 65, 58, 15] remains a critical research topic in the fields of computer vision and machine learning.

Kullback-Leibler (KL) Divergence quantifies the degree of dissimilarity between a probability distribution and a reference distribution. As one of the most frequently used loss functions, it finds application in various scenarios, such as adversarial training [71, 66, 14, 35], knowledge distillation [33, 6, 73], incremental learning [5, 42], and robustness on out-of-distribution data [29]. Although many of these studies incorporate KL Divergence loss as part of their algorithms, they may not thoroughly investigate the underlying mechanisms of the loss function. To bridge this gap, our paper aims to elucidate the working mechanism of KL Divergence regarding gradient optimization.

Our study focuses on the analysis of Kullback–Leibler (KL) Divergence loss from the perspective of gradient optimization. For models with *softmax* activation, we provide theoretical proof that it is equivalent to the Decoupled Kullback–Leibler (DKL) Divergence loss which comprises a weighted Mean Square Error ($\mathbf{w}$MSE) loss and a Cross-Entropy loss with soft labels. Figures 1(a) and (b) reveal the equivalence between KL and DKL losses regarding gradient backpropagation. With the decomposed formulation, it becomes more convenient to analyze how the KL loss works in training optimization.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) KL Loss (b) DKL Loss (c) IKL Loss

Figure 1: **Comparisons of gradient backpropagation between KL, DKL, and IKL losses.** DKL loss is equivalent to KL loss regarding backward optimization. $\mathcal{M}$ and $\mathcal{N}$ can be the same one (like in adversarial training) or two separate (like in knowledge distillation) models determined by application scenarios. Similarly, $x_m, x_n \in X$ can also be the same one (like in knowledge distillation) or two different (like in adversarial training) images. o_m, o_n are logits output with which the probability vectors are obtained when applying the *softmax* activation. Black arrows represent the forward process while colored arrows indicate the backward process driven by the corresponding loss functions in the same color. "$\mathbf{w}$MSE" is a weighted Mean Square Error (MSE) loss. "$\bar{\mathbf{w}}$MSE" is incorporated with class-wise global information.

We have identified potential issues of KL loss with the newly derived DKL loss. Specifically, its gradient optimization is asymmetric regarding the inputs. As illustrated in Figure 1(b), the gradients on o_m and o_n are asymmetric and driven by the $\mathbf{w}$MSE and Cross-Entropy individually. This optimization asymmetry can lead to the $\mathbf{w}$MSE component being ignored in certain scenarios, such as knowledge distillation where o_m is the logits of the teacher model and detached from gradient backpropagation. Fortunately, it is convenient to address this issue with the decoupled formulation of DKL loss by breaking the asymmetric optimization property. As evidenced by Figure 1(c), enabling gradient on o_n from $\mathbf{w}$MSE alleviates this problem.

Moreover, $\mathbf{w}$MSE component is guided by sample-wise predictions. Hard examples with incorrect prediction scores can lead to challenging optimization. We thus insert class-wise global information to regularize the training process. Integrating DKL with these two points, we derive the Improved Kullback–Leibler (IKL) Divergence loss.

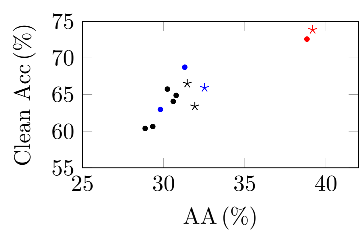


Figure 2: **We achieve SOTA robustness on CIFAR-100.** "star" represents our method while "circle" denotes previous methods. "Black" means adversarial training with image preprocessing only including random crop and flip, "Blue" is for methods with AutoAug or CutMix, and "red" represents methods using synthesized data. AA is short for Auto-Attack [10].

To demonstrate the effectiveness of our proposed IKL loss, we evaluate it with adversarial training and knowledge distillation tasks. Our experimental results on CIFAR-10/100 and ImageNet show that the IKL loss achieves new state-of-the-art robustness on the public leaderboard of *RobustBench* [1]. Comparisons with previous methods on adversarial robustness are shown in Figure 2.

In summary, the main contributions of our work are:

- We reveal that the KL loss is mathematically equivalent to a composite of a weighted MSE ($\mathbf{w}$MSE) loss and a Cross-Entropy loss employing soft labels.
- Based on our analysis, we propose two modifications for enhancement: breaking its asymmetric optimization and incorporating class-wise global information, deriving the Improved Kullback–Leibler (IKL) loss.
- With the proposed IKL loss, we obtain the state-of-the-art adversarial robustness on *RobustBench* and competitive knowledge distillation performance on CIFAR-10/100 and ImageNet.

[1] https://robustbench.github.io/

2 Related Work

Adversarial Robustness. Since the identification of adversarial examples by Szegedy et al. [57], the security of deep neural networks (DNNs) has gained significant attention, and ensuring the reliability of DNNs has become a prominent topic in the machine learning community. Adversarial training [46], being the most effective method, stands out due to its consistently high performance.

Adversarial training incorporates adversarial examples into the training process. Madary et al. [46] propose the adoption of the universal first-order adversary, specifically the PGD attack, in adversarial training. Zhang et al. [71] trade off the accuracy and robustness by the KL loss. Wu et al. [66] introduce adversarial weight perturbation to explicitly regulate the flatness of the weight loss landscape. Cui et al. [14] leverage guidance from naturally-trained models to regularize the decision boundary in adversarial training. Additionally, various other techniques [35] focusing on optimization or training aspects have also been developed. Besides, recently, several works [22, 64, 1] have explored the use of data augmentation techniques to improve adversarial training. We have explored the mechanism of KL loss for adversarial robustness in this paper. The effectiveness of the proposed IKL loss is tested in both settings with and without synthesized data [38].

Knowledge Distillation. The concept of Knowledge Distillation (KD) was first introduced by Hinton et al. [33]. It involves extracting "dark knowledge" from accurate teacher models to guide the learning process of student models. This is achieved by utilizing the KL loss to regularize the output probabilities of student models, aligning them with those of their teacher models when given the same inputs. This simple yet effective technique significantly improves the generalization ability of smaller models and finds extensive applications in various domains. Since the initial success of KD [33], several advanced methods, including logits-based [9, 21, 48, 67, 72, 73, 34] and features-based approaches [53, 61, 30, 70, 6, 31, 32, 39, 49, 50, 68], have been introduced. This paper decouples the KL loss into a new formulation, *i.e.*, DKL, and addresses the limitation of KL loss for application scenarios like knowledge distillation.

Other Applications of KL Divergence Loss. In semi-supervised learning, the KL loss acts as a consistency loss between the outputs of weakly and strongly augmented images [56, 60]. In continual learning, KL loss helps retain previous knowledge by encouraging consistency between the outputs of pre-trained and newly updated models [5, 42]. Additionally, KL loss is also applied to enhance model robustness to out-of-distribution data [29, 74, 76].

3 Method

In this section, we begin by introducing the preliminary mathematical notations in Section 3.1. Theoretical analysis of the equivalence between KL and DKL losses is presented in Section 3.2. Finally, we propose the IKL loss to address potential limitations of KL/DKL in Section 3.3, followed by a case study with additional analysis in Section 3.4.

3.1 Preliminary

Definition of KL Divergence. Kullback-Leibler (KL) Divergence measures the differences between two probability distributions. For distributions P and Q of a continuous random variable, It is defined to be the integral:

$$D_{KL}(P||Q) = \int_{-\infty}^{+\infty} p(x) * \log \frac{p(x)}{q(x)} dx, \tag{1}$$

where p and q denote the probability densities of P and Q.

The KL loss is one of the most widely used objectives in deep learning, applied across various contexts involving categorical distributions. This paper primarily examines its role in adversarial training and knowledge distillation tasks.

In adversarial training, the KL loss improves model robustness by aligning the output probability distribution of adversarial examples with that of their corresponding clean images, thus minimizing output changes despite input perturbations. In knowledge distillation, the KL loss enables a student model to mimic the behavior of a teacher model, facilitating knowledge transfer that enhances the student model's generalization performance.

Applications of KL Loss in Deep Learning. We consider image classification models that predict probability vectors using the *softmax* activation. Let $\mathbf{o}_i \in \mathbb{R}^C$ represent the logits output from a model given an input image $x_i \in X$, where C denotes the number of classes. The predicted probability vector is $\mathbf{s}_i \in \mathbb{R}^C$, computed as $\mathbf{s}_i = \textit{softmax}(\mathbf{o}_i)$. The values $\mathbf{o}_i^j$ and $\mathbf{s}_i^j$ correspond to the logits and probabilities for the j-th class, respectively. The KL loss is often used to encourage similarity between $\mathbf{s}_m$ and $\mathbf{s}_n$ in various scenarios, resulting in the following objective:

$$\mathcal{L}_{KL}(x_m, x_n) = \sum_{j=1}^{C} \mathbf{s}_m^j * \log \frac{\mathbf{s}_m^j}{\mathbf{s}_n^j}. \tag{2}$$

For example, in adversarial training, x_m represents a clean image, while x_n is its corresponding adversarial example. In knowledge distillation, x_m and x_n are the same image, but they are input separately to the teacher and student models. Notably, in the knowledge distillation process, $\mathbf{s}_m$ is detached from gradient backpropagation, as the teacher model is pre-trained and fixed during training.

3.2 Decoupled Kullback-Leibler Divergence Loss

Previous works [33, 73, 71, 14] have incorporated the KL loss into their algorithms without investigating its underlying mechanism. This paper aims to uncover the driving force behind gradient optimization by analyzing the KL loss function. With the backpropagation rule in training optimization, the derivative gradients are as follows,

$$\frac{\partial \mathcal{L}_{KL}}{\partial \mathbf{o}_m^j} = \sum_{k=1}^{C} ((\Delta \mathbf{m}_{j,k} - \Delta \mathbf{n}_{j,k}) * (\mathbf{s}_m^k * \mathbf{s}_m^j)), \tag{3}$$

$$\frac{\partial \mathcal{L}_{KL}}{\partial \mathbf{o}_n^j} = \mathbf{s}_n^j - \mathbf{s}_m^j, \tag{4}$$

where $\Delta \mathbf{m}_{j,k} = \mathbf{o}_m^j - \mathbf{o}_m^k$, and $\Delta \mathbf{n}_{j,k} = \mathbf{o}_n^j - \mathbf{o}_n^k$.

Leveraging the antiderivative technique alongside the structured gradient information, we introduce a novel formulation called the Decoupled Kullback-Leibler (DKL) Divergence loss, as presented in Theorem 1. The DKL loss is designed to be equivalent to the KL loss while offering a more analytically tractable alternative for further exploration and study.

Theorem 1 From the perspective of gradient optimization, the Kullback-Leibler (KL) Divergence loss is equivalent to the following Decoupled Kullback-Leibler (DKL) Divergence loss when $\alpha = 1$ and $\beta = 1$.

$$\mathcal{L}_{DKL}(x_m, x_n) = \underbrace{\frac{\alpha}{4} \| \sqrt{\mathcal{S}(\mathbf{w}_m)} (\Delta \mathbf{m} - \mathcal{S}(\Delta \mathbf{n})) \|^2}_{\textbf{weighted MSE (wMSE)}} \underbrace{- \beta \cdot \mathcal{S}(\mathbf{s}_m^\top) \cdot \log \mathbf{s}_n}_{\textbf{Cross-Entropy}}, \tag{5}$$

where $\mathcal{S}(\cdot)$ represents *stop gradients* operation, $\mathbf{s}_m^\top$ is transpose of $\mathbf{s}_m$, $\mathbf{w}_m^{j,k} = \mathbf{s}_m^j * \mathbf{s}_m^k$, $\Delta \mathbf{m}_{j,k} = \mathbf{o}_m^j - \mathbf{o}_m^k$, and $\Delta \mathbf{n}_{j,k} = \mathbf{o}_n^j - \mathbf{o}_n^k$. Summation is used for the reduction of $\| \cdot \|^2$.

Proof See Appendix A.1.

Interpretation. With Theorem 1, we know that KL loss is equivalent to DKL loss regarding gradient optimization, *i.e., DKL loss produces the same gradients as KL loss given the same inputs.* Therefore, KL loss can be interpreted as a composition of a $\mathbf{w}$MSE loss and a Cross-Entropy loss. This is the first work to reveal the precise quantitative relationships between KL, Cross-Entropy, and MSE losses. Upon examining this new formulation, we identify two potential issues with the KL loss.

Asymmetirc Optimization. As shown in Eqs. (3) and (4), gradient optimization is asymmetric for $\mathbf{o}_m$ and $\mathbf{o}_n$. The $\mathbf{w}$MSE and Cross-Entropy losses in Theorem 1 are complementary and collaboratively work together to make $\mathbf{o}_m$ and $\mathbf{o}_n$ similar. Nevertheless, the asymmetric optimization can cause the $\mathbf{w}$MSE component to be neglected or overlooked when $\mathbf{o}_m$ is detached from gradient backpropagation, which is the case for knowledge distillation, potentially leading to performance degradation.

Sample-wise Prediction Bias. As shown in Eq. (5), $\mathbf{w}_m$ in $\mathbf{w}$MSE component is conditioned on the prediction score of x_m. However, sample-wise predictions can be subject to significant variance.

Incorrect prediction of hard examples or outliers will mislead the optimization and result in unstable training. Our study in Sections 3.4 and 4.4 indicates that the choice of $\mathbf{w}_m$ significantly affects adversarial robustness.

3.3 Improved Kullback-Leibler Divergence Loss

Based on the analysis in Section 3.2, we propose an Improved Kullback-Leibler (IKL) Divergence loss. Distinguished from DKL in Theorem 1, we make the following improvements: 1) *breaking the asymmetric optimization property*; 2) *inserting class-wise global information to mitigate sample-wise bias*. The details are presented as follows.

Breaking the Asymmetric Optimization Property. As shown in Eq. (5), the $\mathbf{w}$MSE component encourages $\mathbf{o}_n$ to resemble $\mathbf{o}_m$ by capturing second-order information, specifically the differences between logits for each pair of classes. Each addend in $\mathbf{w}$MSE only involves logits of two classes. We refer to this property as *locality*. On the other hand, the Cross-Entropy loss ensures that $\mathbf{s}_n$ and $\mathbf{s}_m$ produce similar predicted scores. Each addend in the Cross-Entropy gathers all class logits. We refer to this property as *globality*. Two loss terms collaboratively work together to make $\mathbf{o}_n$ and $\mathbf{o}_m$ similar in *locality* and *globality*. Discarding any one of them can lead to performance degradation.

However, because of the asymmetric optimization property of KL/DKL, the unexpected case can occur when $\mathbf{s}_m$ is detached from the gradient backpropagation (scenarios like knowledge distillation), in which the formulation will be:

$$\mathcal{L}_{DKL-KD}(x_m, x_n) = \underbrace{\frac{\alpha}{4}\|\sqrt{\mathcal{S}(\mathbf{w}_m)}(\mathcal{S}(\Delta\mathbf{m}) - \mathcal{S}(\Delta\mathbf{n}))\|^2}_{\textbf{weighted MSE (wMSE)}} \underbrace{-\beta \cdot \mathcal{S}(\mathbf{s}_m^\top) \cdot \log \mathbf{s}_n}_{\textbf{Cross-Entropy}} \tag{6}$$

As indicated by Eq. (6), the $\mathbf{w}$MSE component loss takes no effect on training optimization since all sub-components of $\mathbf{w}$MSE are detached from gradient propagation, which can potentially hurt the model performance. Knowledge distillation exactly matches this case because the teacher model is fixed during knowledge distillation training. Thanks to the decomposition of DKL formulation, we address this issue by breaking the asymmetric optimization property, *i.e.*, enabling the gradients of $\mathcal{S}(\Delta\mathbf{n})$ in Eq. (5). Then, the updated formulation of Eq. (6) becomes,

$$\widehat{\mathcal{L}}_{DKL-KD}(x_m, x_n) = \underbrace{\frac{\alpha}{4}\|\sqrt{\mathcal{S}(\mathbf{w}_m)}(\mathcal{S}(\Delta\mathbf{m}) - \Delta\mathbf{n})\|^2}_{\textbf{weighted MSE (wMSE)}} \underbrace{-\beta \cdot \mathcal{S}(\mathbf{s}_m^\top) \cdot \log \mathbf{s}_n}_{\textbf{Cross-Entropy}}. \tag{7}$$

After enabling the gradients of $\mathcal{S}(\Delta\mathbf{n})$ in Eq. (5), $\mathbf{w}$MSE will produce symmetric gradients on o_n and o_m. Regarding the knowledge distillation, $\mathbf{w}$MSE can output gradient on o_n and promote the training optimization demonstrated by Eq. (7).

Inserting Class-wise Global Information. Recall in Theorem 1, $\mathbf{w}_m$ in Eq. (5) is calculated as:

$$\mathbf{w}_m^{j,k} = \mathbf{s}_m^j * \mathbf{s}_m^k. \tag{8}$$

It indicates that $\mathbf{w}_m$ depends on the sample-wise prediction scores. Nevertheless, the model cannot output correct predictions when dealing with outliers or hard examples in training. In this case, $\mathbf{w}$MSE will attach the most importance on the predicted class $\hat{y} = \arg\max o_m$ rather than the ground-truth class, which misleads the optimization and makes the training unstable.

We thus insert class-wise global information into $\mathbf{w}$MSE component, replacing $\mathbf{w}_m$ with $\bar{\mathbf{w}}_y$:

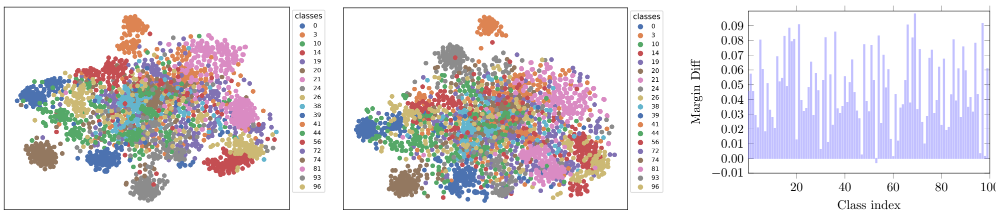
$$\bar{\mathbf{w}}_y^{j,k} = \bar{\mathbf{s}}_y^j * \bar{\mathbf{s}}_y^k, \tag{9}$$

where y is ground-truth label of x_m, $\bar{\mathbf{s}}_y = \frac{1}{|X_y|}\sum_{x_i \in X_y} \mathbf{s}_i$.

The class-wise global information injected by $\bar{\mathbf{w}}_y$ can act as a regularization to enhance intra-class consistency and mitigate biases that may arise from sample noises. Especially, in the late stage of training, $\bar{\mathbf{w}}_y$ can always provide correct predictions, benefiting the optimization of $\bar{\mathbf{w}}$MSE component.

Table 1: **Ablation study on “GI” and “BA” with DKL loss.** “GI” represents “Inserting Global Information”, and “BA” indicates “Breaking Asymmetric Optimization”. “Clean” is the test accuracy of clean images and “AA” is the robustness under Auto-Attack. CIFAR-100 is used for the adversarial training task and ImageNet is adopted for the knowledge distillation task.

Index	GI	BA	Adversarial Training Clean (%)	Adversarial Training AA (%)	Knowledge Distillation Top-1 (%)	Descriptions
(a)	Na	Na	62.87	30.29	71.03	baseline with KL loss.
(b)	✗	✗	62.54	30.20	71.03	DKL, equivalent to KL loss.
(c)	✗	✔	62.69	30.42	71.80	(b) with BA.
(d)	✔	✔	65.76	**31.91**	**71.91**	(c) with GI, *i.e.*, IKL.



(a) IKL-AT (b) TRADES (c) Margin differences

Figure 3: **Visualization comparisons.** (a) t-SNE visualization of the model trained by IKL-AT on CIFAR-100; (b) t-SNE visualization of the model trained by TRADES on CIFAR-100. (c) Class margin differences between models trained by IKL-AT and TRADES.

To this end, we derive the IKL loss in Eq. (10) by incorporating the two designs,

$$\mathcal{L}_{IKL}(x_m, x_n) = \underbrace{\frac{\alpha}{4}\|\sqrt{\mathcal{S}(\bar{\mathbf{w}}_y)}(\Delta\mathbf{m} - \Delta\mathbf{n})\|^2}_{\textbf{weighted MSE } (\bar{\mathbf{w}}\textbf{MSE})} \underbrace{-\beta \cdot \mathcal{S}(\mathbf{s}_m^\top) \cdot \log \mathbf{s}_n}_{\textbf{Cross-Entropy}}, \tag{10}$$

where y is the ground-truth label for x_m, $\bar{\mathbf{w}}_y \in \mathbb{R}^{C\times C}$ is the weights for class y calculated with Eq. (9).

3.4 A Case Study and Analysis

A Case Study. We empirically examine each component of IKL on CIFAR-100 with the adversarial training task and on ImageNet with the knowledge distillation task. Ablation experimental results and their setting descriptions are listed in Table 1. In the implementation, for adversarial training, we use improved TRADES [71] as our baseline that combines with AWP [66] and uses an increasing epsilon schedule [1]. For knowledge distillation, we use the official code from DKD. The comparison between (a) and (b) shows that DKL can achieve comparable performance, confirming the equivalence to KL. The comparisons among (b), (c), and (d) confirm the effectiveness of the “GI” and “BA".

Analysis on Inserting Class-wise Global Information. As evidenced by Table 1, class-wise global information plays an important role in adversarial robustness. The mean probability vector $\bar{s}_y$ of all samples in the class y is more robust than the sample-wise probability vector. During training, once the model gives incorrect predictions for hard samples or outliers, $\mathbf{w}_m$ in Eq. (5) will wrongly guide the optimization. Adoption of $\bar{\mathbf{w}}_y$ in Eq. (10) can mitigate the issue and meanwhile enhance intra-class consistency.

To visualize the effectiveness of inserting class-wise global information, we define the boundary margin for class y as:

$$\text{Margin}_y = \bar{s}_y[y] - \max_{k\neq y} \bar{s}_y[k]. \tag{11}$$

We plot the margin differences between models trained by IKL-AT and TRADES on CIFAR-100. As shown in Figure 3c, almost all class margin differences are positive, demonstrating that there are larger decision boundary margins for the IKL-AT model. Such larger margins lead to stronger robustness. This phenomenon is coherent with our experimental results in Section 4.1.

Table 2: **Test accuracy (%) of clean images and robustness (%) under AutoAttack on CIFAR-100.** All results are the average over three trials.

Dataset	Method	Architecture	Augmentation Type	Clean	AA
CIFAR-100 ($\ell_\infty, \epsilon = 8/255$)	AWP	WRN-34-10	Basic	60.38	28.86
	LBGAT	WRN-34-10	Basic	60.64	29.33
	LAS-AT	WRN-34-10	Basic	64.89	30.77
	ACAT	WRN-34-10	Basic	65.75	30.23
	IKL-AT	WRN-34-10	Basic	**65.76**	**31.91**
	ACAT	WRN-34-10	AutoAug	**68.74**	31.30
	IKL-AT	WRN-34-10	AutoAug	66.08	**32.53**
	DM-AT [64]	WRN-28-10	50M Generated Data	72.58	38.83
	IKL-AT	WRN-28-10	50M Generated Data	**73.85**	**39.18**

Table 3: **Test accuracy (%) of clean images and robustness (%) under AutoAttack on CIFAR-10.** Average over three trials are listed.

Dataset	Method	Architecture	Augmentation Type	Clean	AA
CIFAR-10 ($\ell_\infty, \epsilon = 8/255$)	Rice et al. [52]	WRN-34-20	Basic	85.34	53.42
	LBGAT	WRN-34-20	Basic	**88.70**	53.57
	AWP	WRN-34-10	Basic	85.36	56.17
	LAS-AT	WRN-34-10	Basic	87.74	55.52
	ACAT	WRN-34-10	Basic	82.41	55.36
	IKL-AT	WRN-34-10	Basic	84.80	**57.09**
	ACAT	WRN-34-10	AutoAug	88.64	57.05
	IKL-AT	WRN-34-10	AutoAug	85.20	**57.62**
	DM-AT [64]	WRN-28-10	20M Generated Data	92.44	67.31
	IKL-AT	WRN-28-10	20M Generated Data	92.16	**67.75**

We also randomly sample 20 classes in CIFAR-100 for t-SNE visualization. The numbers in the pictures are class indexes. For each sampled class, we collect the feature representation of natural images and adversarial examples with the validation set. The visualization by t-SNE is shown in Figures 3b and 3a. Compared with TRADES that trained with KL loss, features by IKL-AT models are more compact and separable.

4 Experiments

To verify the effectiveness of our IKL loss, we conduct experiments on CIFAR-10, CIFAR100, and ImageNet for adversarial training (Section 4.1) and knowledge distillation (Sections 4.2 and 4.3). More ablation studies are included in Section 4.4.

4.1 Adversarial Robustness

Experimental Settings. We use an improved version of TRADES [71] as our baseline, which incorporates AWP [66] and adopts an increasing epsilon schedule. SGD optimizer with a momentum of 0.9 is used. We use the cosine learning rate strategy with an initial learning rate of 0.2 and train models 200 epochs. The batch size is 128, the weight decay is 5e-4 and the perturbation size ϵ is set to 8/255. Following previous work [71, 14], standard data augmentation including random crops and random horizontal flip is performed for data preprocessing. Models are trained with 4 Nvidia GeForce 3090 GPUs.

Under the setting of training with synthesized data by generative models, we strictly follow the training configurations in DM-AT [64] for fair comparisons. Our implementations are based on their open-sourced code. We only replace the KL loss with our IKL loss.

Datasets and Evaluation. Following previous work [66, 14], CIFAR-10 and CIFAR-100 are used for the adversarial training task. we report the clean accuracy on natural images and adversarial robustness under Auto-Attack [10] with epsilon 8/255.

Table 4: **Top-1 accuracy (%) on the ImageNet validation and training speed (sec/iteration) comparisons.** Training speed is calculated on 4 Nvidia GeForce 3090 GPUs with a batch of 512 224x224 images. All results are the average over three trials.

Distillation Manner	Teacher / Student	Extra Parameters	ResNet34 73.31 / ResNet18 69.75		ResNet50 76.16 / MobileNet 68.87	
Features	AT	✗	70.69		69.56	
	OFD	✔	70.81		71.25	
	CRD	✔	71.17		71.37	
	ReviewKD	✔	71.61	0.319 s/iter	72.56	0.526 s/iter
Logits	DKD	✗	71.70		72.05	
	KD	✗	71.03		70.50	
	IKL-KD	✗	**71.91**	**0.197 s/iter**	**72.84**	**0.252 s/iter**

Comparison Methods. To compare with previous methods, we categorize them into two groups according to the different types of data preprocessing:

- Methods with basic augmentation, *i.e.*, random crops and random horizontal flip.
- Methods using augmentation with generative models or Auto-Aug [11], CutMix [69].

Comparisons with State-of-the-art on CIFAR-100. On CIFAR-100, with the basic augmentations setting, we compare with AWP, LBGAT, LAS-AT, and ACAT. The experimental results are summarized in Table 2. Our WRN-34-10 models trained with IKL loss do a better trade-off between natural accuracy and adversarial robustness. With $\frac{\alpha}{4} = 5$ and $\beta = 5$, the model achieves **65.76%** top-1 accuracy on natural images while **31.91%** adversarial robustness under Auto-Attack. An interesting phenomenon is that IKL-AT is complementary to data augmentation strategies, like AutoAug, without any specific designs, which is different from the previous observation that the data augmentation strategy hardly benefits adversarial training [66]. With AutoAug, we obtain **32.53%** adversarial robustness, achieving new state-of-the-art under the setting without extra real or generated data.

We follow DM-AT [64] to take advantage of synthesized images generated by the popular diffusion models [38]. With 50M generated images, we create new state-of-the-art with WideResNet-28-10, achieving **73.85%** top-1 natural accuracy and **39.18%** adversarial robustness under Auto-Attack.

Comparison with State-of-the-art on CIFAR-10. Experimental results on CIFAR-10 are listed in Table 3. With the basic augmentation setting, our model achieves 84.80% top-1 accuracy on natural images and 57.09% robustness, outperforming AWP by 0.92% on robustness. With extra generated data, we improve the state-of-the-art by 0.44%, achieving **67.75%** robustness.

4.2 Knowledge Distillation

Datasets and Evaluation. Following previous work [6, 61], we conduct experiments on CIFAR-100 [40] and ImageNet [54] to show the advantages of IKL on knowledge distillation. For evaluation, we report top-1 accuracy on CIFAR-100 and ImageNet validation. The training speed of different methods is also discussed.

Experimental Settings. We follow the experimental settings in DKD. Our implementation for knowledge distillation is based on their open-sourced code. Models are trained with 1 and 8 Nvidia GeForce 3090 GPUs on CIFAR and ImageNet separately.

Specifically, on CIFAR-100, we train all models for 240 epochs with a learning rate that decayed by 0.1 at the 150th, 180th, and 210th epoch. We initialize the learning rate to 0.01 for MobileNet and ShuffleNet, and 0.05 for other models. The batch size is 64 for all models. We train all models three times and report the mean accuracy. On ImageNet, we use the standard training that trains the model for 100 epochs and decays the learning rate for every 30 epochs. We initialize the learning rate to 0.2 and set the batch size to 512.

For both CIFAR-100 and ImageNet, we consider the distillation among the architectures having the same unit structures, like ResNet56 and ResNet20, VGGNet13 and VGGNet8. On the other

Table 5: **Peformance (%) on imbalanced data, *i.e.*, the ImageNet-LT.**

Method	Teacher	Student	Many(%)	Medium(%)	Few(%)	All(%)
Baseline	-	ResNet-18	63.16	33.47	5.88	41.15
Baseline	-	ResNet-50	67.25	38.56	8.21	45.47
Baseline	-	ResNet-101	68.91	42.32	11.24	48.33
KL-KD	ResNeXt-101	ResNet-18	64.6	37.88	9.53	44.32
KL-KD	ResNeXt-101	ResNet-50	68.83	42.31	11.37	48.31
IKL-KD	ResNeXt-101	ResNet-18	66.60	38.53	8.19	**45.21**
IKL-KD	ResNeXt-101	ResNet-50	70.06	43.47	10.99	**49.29**

hand, we also explore the distillation among architectures made up of different unit structures, like WideResNet and ShuffleNet, VggNet and MobileNet-V2.

Comparison Methods. According to the information extracted from the teacher model in distillation training, knowledge distillation methods can be divided into two categories:

- Features-based methods [53, 61, 6, 30]. This kind of method makes use of features from different layers of the teacher model, which can need extra parameters and high training computational costs.
- Logits-based methods [33, 73]. This kind of method only makes use of the logits output of the teacher model, which does not require knowing the architectures of the teacher model and thus is more general in practice.

Comparison with State-of-the-art on CIFAR-100. Experimental results on CIFAR-100 are summarized in Table 13 and Table 14 (in Appendix). Table 13 lists the comparisons with previous methods under the setting that the architectures of the teacher and student have the same unit structures. Models trained by IKL-KD can achieve comparable or better performance in all considered settings. Specifically, we achieve the best performance in 4 out of 6 training settings. Table 14 shows the comparisons with previous methods under the setting that the architectures of the teacher and student have different unit structures. We achieve the best performance in 3 out of 5 training configurations.

Comparison with State-of-the-art on ImageNet. We empirically show the comparisons with other methods on ImageNet in Table 4. With a ResNet34 teacher, our ResNet18 achieves **71.91%** top-1 accuracy. With a ResNet50 teacher, our MobileNet achieves **72.84%** top-1 accuracy. Models trained by IKL-KD surpass all previous methods while saving **38%** and **52%** computation costs for ResNet34–ResNet18 and ResNet50–MobileNet distillation training respectively when compared with ReviewKD [6].

4.3 Knowledge Distillation on Imbalanced Data

Data often follows a long-tailed distribution. Tackling the long-tailed recognition problem is essential for real-world applications. Lots of research has contributed to algorithms and theories [3, 19, 37, 47, 17, 16, 18, 77, 75] on the problem. In this work, we examine how the knowledge distillation with our IKL loss affects model performance on imbalanced data, *i.e.*, ImageNet-LT [45]. We train ResNets models 90 epochs with *Random-Resized-Crop* and horizontal flip as image pre-processing. Following previous work [13], we report the top-1 accuracy on Many-shot, Meidum-shot, Few-shot, and All classes. As shown in Table 5, IKL-KD consistently outperforms KL-KD on imbalanced data.

4.4 Ablation Studies

Ablation on α and β for Adversarial Robustness. Thanks to the decomposition of the DKL loss formulation, the two components (wMSE and Cross-Entropy) of IKL can be manipulated independently. We empirically study the effects of hyper-parameters of α and β on CIFAR-100 for adversarial robustness. Clean accuracy on natural data and robustness under AA [10] are reported in Table 7 and Table 8. Reasonable α and β should be chosen for the best trade-off between natural accuracy and adversarial robustness.

Ablation on Temperature (τ) for Global Information. As discussed in Section 3.3, the incorporated class-wise global information is proposed to promote intra-class consistency and mitigate the biases

Table 6: **Ablation study on hyper-parameters of IKL.**

$\frac{\alpha}{4}$	Clean	AA
$\frac{\alpha}{4}=3$	67.52	31.29
$\frac{\alpha}{4}=4$	66.26	31.33
$\frac{\alpha}{4}=5$	65.76	31.91
$\frac{\alpha}{4}=6$	65.14	31.64

Table 7: Effects of $\frac{\alpha}{4}$.

β	Clean	AA
$\beta=2$	66.13	30.95
$\beta=3$	66.31	31.33
$\beta=4$	66.00	31.57
$\beta=5$	65.76	31.91

Table 8: Effects of β.

τ	Clean	AA
$\tau=1$	59.99	31.35
$\tau=2$	63.77	31.88
$\tau=3$	65.28	31.69
$\tau=4$	65.76	31.91

Table 9: Effects of τ.

Table 10: **Ablation study of ϵ.**

Method	Clean	AA						
		$\frac{2}{255}$	$\frac{4}{255}$	$\frac{6}{255}$	$\frac{8}{255}$	$\frac{10}{255}$	$\frac{12}{255}$	Avg.
TRADES	62.87	53.88	45.31	37.28	30.29	24.28	19.17	35.04
IKL-AT	**63.40**	**55.31**	**46.76**	**38.98**	**31.91**	**25.33**	**19.98**	**36.38**

Table 11: **Evaluation under PGD and CW attacks.**

Method	Acc	PGD-10	PGD-20	CW-10	CW-20	Auto-Attack	Worst
KL(TRADES)	62.87	36.01	35.84	40.03	39.86	30.29	30.29
IKL(Ours)	63.40	36.78	36.55	40.72	40.47	31.91	31.92
IKL(Ours with autoaug)	65.93	38.15	37.75	41.10	40.86	32.53	32.52
IKL(Ours with synthetic data)	73.85	44.43	44.12	47.59	47.53	39.18	39.18

from sample noises. When calculating the $\bar{w}_y$ and $\bar{s}_y$, a temperature τ could be applied before getting sample probability vectors. We summarize the experimental results in Table 9 for ablation of τ. Interestingly, we observe that models usually exhibit higher performance on clean images with a higher τ. There are even 5.75% improvements of clear accuracy while keeping comparable robustness when changing $\tau = 1$ to $\tau = 4$, which implies the vast importance of weights in $\mathbf{w}$MSE component of DKL/KL for adversarial robustness. To achieve the strongest robustness, we finally choose $\tau = 4$ as illustrated by empirical study.

Ablation on Various Perturbation Size ϵ. We evaluate model robustness with unknown perturbation size ϵ in training under Auto-Attack. The experimental results are summarized in Table 10. As shown in Table 10, model robustness decreases significantly as the ϵ increases for both the TRADES model and our model. Nevertheless, our model achieves stronger robustness than the TRADES model under all of ϵ, outperforming TRADES by 1.34% on average robustness. The experimental results demonstrate the super advantages of models adversarially trained with our IKL loss.

Robustness under Other Attacks. Auto-Attack is currently one of the strongest attack methods. It ensembles several adversarial attack methods including APGD-CE, APGD-DLR, FAB, and Square Attack. To show the effectiveness of our IKL loss, we also evaluate our models under PGD and CW attacks with 10 and 20 iterations. The perturbation size and step size are set to 8/255 and 2/255 respectively. As shown in Table 11, with increasing iterations from 10 to 20, our models show similar robustness, demonstrating that our models don't suffer from obfuscated gradients problem.

5 Conclusion and Limitation

In this paper, we have investigated the mechanism of Kullback-Leibler (KL) Divergence loss in terms of gradient optimization. Based on our analysis, we decouple the KL loss into a weighted Mean Square Error ($\mathbf{w}$MSE) loss and a Cross-Entropy loss with soft labels. The new formulation is named Decoupled Kullback-Leibler (DKL) Divergence loss. To address the spotted issues of KL/DKL, we make two improvements that break the asymmetric optimization property and incorporate class-wise global information, deriving the Improved Kullback-Leibler (IKL) Divergence loss. Experimental results on CIFAR-10/100 and ImageNet show that we create new state-of-the-art adversarial robustness and competitive performance on knowledge distillation. This underscores the efficacy of our Innovative KL (IKL) loss technique. The KL loss exhibits a wide range of applications. As part of our future work, we aim to explore and highlight the versatility of IKL in various other scenarios, like robustness on out-of-distribution data, and incremental learning.

References

[1] Sravanti Addepalli, Samyak Jain, and Venkatesh Babu Radhakrishnan. Efficient and effective augmentation strategy for adversarial training. In *NeurIPS*, 2022.

[2] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *CVPR*, pages 4413–4421, 2018.

[3] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *NeurIPS*, 2019.

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020.

[5] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *ECCV*, pages 532–547, 2018.

[6] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *CVPR*, 2021.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[8] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, 2020.

[9] Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *ICCV*, pages 4794–4802, 2019.

[10] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*. PMLR, 2020.

[11] Ekin D. Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019.

[12] Jiequan Cui, Pengguang Chen, Ruiyu Li, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Fast and practical neural architecture search. In *ICCV*, pages 6509–6518, 2019.

[13] Jiequan Cui, Shu Liu, Zhuotao Tian, Zhisheng Zhong, and Jiaya Jia. Reslt: Residual learning for long-tailed recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2022.

[14] Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial training. In *ICCV*, 2021.

[15] Jiequan Cui, Yuhui Yuan, Zhisheng Zhong, Zhuotao Tian, Han Hu, Stephen Lin, and Jiaya Jia. Region rebalance for long-tailed semantic segmentation. *arXiv preprint arXiv:2204.01969*, 2022.

[16] Jiequan Cui, Zhisheng Zhong, Shu Liu, Bei Yu, and Jiaya Jia. Parametric contrastive learning. In *ICCV*, pages 715–724, 2021.

[17] Jiequan Cui, Zhisheng Zhong, Zhuotao Tian, Shu Liu, Bei Yu, and Jiaya Jia. Generalized parametric contrastive learning. *arXiv preprint arXiv:2209.12400*, 2022.

[18] Jiequan Cui, Beier Zhu, Xin Wen, Xiaojuan Qi, Bei Yu, and Hanwang Zhang. Classes are not equal: An empirical study on image recognition fairness. In *CVPR*, pages 23283–23292, 2024.

[19] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, pages 9268–9277, 2019.

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[21] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *ICML*, pages 1607–1616. PMLR, 2018.

[22] Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. Improving robustness using generated data. 2021.

[23] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *NeurIPS*, 2020.

[24] Zhiwei Hao, Jianyuan Guo, Kai Han, Han Hu, Chang Xu, and Yunhe Wang. Vanillakd: Revisit the power of vanilla knowledge distillation from small scale to large scale. *arXiv preprint arXiv:2305.15781*, 2023.

[25] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022.

[26] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *CVPR*, pages 2961–2969, 2017.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[29] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

[30] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *ICCV*, 2019.

[31] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1921–1930, 2019.

[32] Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *AAAI*, pages 3779–3787, 2019.

[33] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. 2015.

[34] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. In *NeurIPS*, 2022.

[35] Xiaojun Jia, Yong Zhang, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. Las-at: Adversarial training with learnable attack strategy. In *CVPR*, 2022.

[36] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711. Springer, 2016.

[37] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

[38] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

[39] Jangho Kim, SeongUk Park, and Nojun Kwak. Paraphrasing complex network: Network compression via factor transfer. *NeurIPS*, 2018.

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

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

[42] Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. Overcoming catastrophic forgetting by incremental moment matching. *NeurIPS*, 2017.

[43] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[44] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021.

[45] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019.

[46] 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.

[47] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020.

[48] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *AAAI*, pages 5191–5198, 2020.

[49] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *CVPR*, pages 3967–3976, 2019.

[50] Baoyun Peng, Xiao Jin, Jiaheng Liu, Dongsheng Li, Yichao Wu, Yu Liu, Shunfeng Zhou, and Zhaoning Zhang. Correlation congruence for knowledge distillation. In *ICCV*, pages 5007–5016, 2019.

[51] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[52] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *ICML*. PMLR, 2020.

[53] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

[54] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. ImageNet large scale visual recognition challenge. *IJCV*, 2015.

[55] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[56] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33:596–608, 2020.

[57] 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.

[58] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *CVPR*, pages 11662–11671, 2020.

[59] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V. Le. Mnasnet: Platform-aware neural architecture search for mobile. In *CVPR*, 2019.

[60] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *NeurIPS*, 30, 2017.

[61] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *ICLR*, 2020.

[62] Zhuotao Tian, Pengguang Chen, Xin Lai, Li Jiang, Shu Liu, Hengshuang Zhao, Bei Yu, Ming-Chang Yang, and Jiaya Jia. Adaptive perspective distillation for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1372–1387, 2022.

[63] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2020.

[64] Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638*, 2023.

[65] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, pages 499–515. Springer, 2016.

[66] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

[67] Chenglin Yang, Lingxi Xie, Chi Su, and Alan L Yuille. Snapshot distillation: Teacher-student optimization in one generation. In *CVPR*, pages 2859–2868, 2019.

[68] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *CVPR*, pages 4133–4141, 2017.

[69] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

[70] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017.

[71] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *ICML*. PMLR, 2019.

[72] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *CVPR*, pages 4320–4328, 2018.

[73] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *CVPR*, 2022.

[74] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15659–15669, 2023.

[75] Beier Zhu, Yulei Niu, Xian-Sheng Hua, and Hanwang Zhang. Cross-domain empirical risk minimization for unbiased long-tailed classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 3589–3597, 2022.

[76] Beier Zhu, Yulei Niu, Saeil Lee, Minhoe Hur, and Hanwang Zhang. Debiased fine-tuning for vision-language models by prompt regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 3834–3842, 2023.

[77] Beier Zhu, Kaihua Tang, Qianru Sun, and Hanwang Zhang. Generalized logit adjustment: Calibrating fine-tuned models by removing label bias in foundation models. *Advances in Neural Information Processing Systems*, 36, 2024.

A Appendix

A.1 Proof to Theorem 1

To demonstrate that DKL in Eq. (5) is equivalent to KL in Eq. (2) for training optimization, we prove that DKL and KL produce the same gradients when given the same inputs.

For KL loss, we have the following derivatives according to the chain rule:

$$
\begin{aligned}
\frac{\partial \mathbf{s}_m^i}{\partial \mathbf{o}_m^i} &= \mathbf{s}_m^i * \sum_{j!=i}^{C} \mathbf{s}_m^j, \\
\frac{\partial \mathbf{s}_m^j}{\partial \mathbf{o}_m^i} &= -\mathbf{s}_m^i * \mathbf{s}_m^j, \\
\frac{\partial \mathcal{L}_{KL}}{\partial \mathbf{s}_m^i} &= \log \mathbf{s}_m^i - \log \mathbf{s}_n^i + 1, \\
\frac{\partial \mathcal{L}_{KL}}{\partial \mathbf{o}_n^i} &= \mathbf{s}_n^i - \mathbf{s}_m^i
\end{aligned} \tag{12}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{KL}}{\partial \mathbf{o}_m^i} &= \frac{\mathcal{L}_{KL}}{\partial \mathbf{s}_m^i} * \frac{\partial \mathbf{s}_m^i}{\partial \mathbf{o}_m^i} + \sum_{j!=i}^{C} \frac{\mathcal{L}_{KL}}{\partial \mathbf{s}_m^j} * \frac{\partial \mathbf{s}_m^j}{\partial \mathbf{o}_m^i} \\
&= (\log \mathbf{s}_m^i - \log \mathbf{s}_n^i + 1) * \mathbf{s}_m^i * \sum_{j!=i}^{C} \mathbf{s}_m^j + \sum_{j!=i}^{C} (\log \mathbf{s}_m^j - \log \mathbf{s}_n^j + 1) * (-\mathbf{s}_m^j * \mathbf{s}_m^i) \\
&= \sum_{i!=j}^{C} ((\log \mathbf{s}_m^i - \log \mathbf{s}_m^j) - (\log \mathbf{s}_n^i - \log \mathbf{s}_n^j)) * (\mathbf{s}_m^j * \mathbf{s}_m^i) \\
&= \sum_{i!=j}^{C} ((\mathbf{o}_m^i - \mathbf{o}_m^j) - (\mathbf{o}_n^i - \mathbf{o}_n^j)) * (\mathbf{s}_m^j * \mathbf{s}_m^i) \\
&= \sum_{i!=j}^{C} (\Delta \mathbf{m}_{i,j} - \Delta \mathbf{n}_{i,j}) * \mathbf{w}_m^{i,j} \\
&= \sum_{j}^{C} (\Delta \mathbf{m}_{i,j} - \Delta \mathbf{n}_{i,j}) * \mathbf{w}_m^{i,j}
\end{aligned} \tag{13}
$$

For DKL los, we expand the Eq. (5) as:

$$
\begin{aligned}
\mathcal{L}_{DKL}(x_m, x_n) &= \underbrace{\frac{\alpha}{4} \| \sqrt{\mathcal{S}(\mathbf{w}_m)} (\Delta \mathbf{m} - \mathcal{S}(\Delta \mathbf{n})) \|^2}_{\textbf{weighted MSE (wMSE)}} \underbrace{- \beta \cdot \mathcal{S}(\mathbf{s}_m^\top) \cdot \log \mathbf{s}_n}_{\textbf{Cross-Entropy}} \\
&= \underbrace{\frac{\alpha}{4} \sum_{j=1}^{C} \sum_{k=1}^{C} ((\Delta \mathbf{m}_{j,k} - \mathcal{S}(\Delta \mathbf{n}_{j,k}))^2 * \mathcal{S}(\mathbf{w}_m^{j,k}))}_{\textbf{weighted MSE (wMSE)}} \underbrace{- \beta \sum_{j=1}^{C} \mathcal{S}(\mathbf{s}_m^j) * \log \mathbf{s}_n^j}_{\textbf{Cross-Entropy}},
\end{aligned} \tag{14}
$$

According to the chain rule, we obtain the following equations:

$$
\frac{\partial \mathcal{L}_{DKL}}{\partial \mathbf{o}_n^i} = \beta * (\mathbf{s}_n^i - \mathbf{s}_m^i) \tag{15}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{DKL}}{\partial \mathbf{o}_m^i} &= \frac{\alpha}{4} * 2 * (\sum_{j}^{C} (\Delta \mathbf{m}_{j,i} - \Delta \mathbf{n}_{j,i}) * (-\mathbf{w}_m^{j,i}) + \sum_{k}^{C} (\Delta \mathbf{m}_{i,k} - \Delta \mathbf{n}_{i,k}) * \mathbf{w}_m^{i,k}) \\
&= \alpha * \sum_{j}^{C} (\Delta \mathbf{m}_{i,j} - \Delta \mathbf{n}_{i,j}) * \mathbf{w}_m^{i,j}
\end{aligned} \tag{16}
$$

Table 12: **New state-of-the-art on public leaderboard *R*obustBench [10].**

Experimental Settings	augmentation strategy	Clean	AA	Computation saving
w/o Generated Data (Previous best results)	Basic	62.99	31.20	
w/o Generated Data (Ours)	Basic	**65.76(+2.67)**	**31.91(+0.71)**	**33.3%**
w/o Generated Data (Previous best results)	Autoaug	**68.75**	31.85	
w/o Generated Data (Ours)	Autoaug	66.08	**32.53(+0.68)**	**33.3%**
w/ Generated Data (Previous best results)	Genreated data	72.58	38.83	
w/ Generated Data (Ours)	Generated data	**73.85(+1.27)**	**39.18(+0.35)**	0%

Table 13: **Top-1 accuracy (%) on the CIFAR-100 validation.** Teachers and students are in the **same** architectures. All results are the average over three trials.

Distillation Manner							
	Teacher	ResNet56 72.34	ResNet110 74.31	ResNet32×4 79.42	WRN-40-2 75.61	WRN-40-2 75.61	VGG13 74.64
	Student	ResNet20 69.06	ResNet32 71.14	ResNet8×4 72.50	WRN-16-2 73.26	WRN-40-1 71.98	VGG8 70.36
Features	FitNet	69.21	71.06	73.50	73.58	72.24	71.02
	RKD	69.61	71.82	71.90	73.35	72.22	71.48
	CRD	71.16	73.48	75.51	75.48	74.14	73.94
	OFD	70.98	73.23	74.95	75.24	74.33	73.95
	ReviewKD	71.89	73.89	75.63	76.12	**75.09**	74.84
Logits	DKD	**71.97**	74.11	76.32	76.24	74.81	74.68
	KD	70.66	73.08	73.33	74.92	73.54	72.98
	IKL-KD	71.44	**74.26**	**76.59**	**76.45**	74.98	**74.98**

Table 14: **Top-1 accuracy (%) on the CIFAR-100 validation.** Teachers and students are in **different** architectures. All results are the average over 3 trials.

Distillation Manner						
	Teacher	ResNet32×4 79.42	WRN-40-2 75.61	VGG13 74.64	ResNet50 79.34	ResNet32×4 79.42
	Student	ShuffleNet-V1 70.50	ShuffleNet-V1 70.50	MobileNet-V2 64.60	MobileNet-V2 64.60	ShuffleNet-V2 71.82
Features	FitNet	73.59	73.73	64.14	63.16	73.54
	RKD	72.28	72.21	64.52	64.43	73.21
	CRD	75.11	76.05	69.73	69.11	75.65
	OFD	75.98	75.85	69.48	69.04	76.82
	ReviewKD	**77.45**	77.14	70.37	69.89	**77.78**
Logits	DKD	76.45	76.70	69.71	70.35	77.07
	KD	74.07	74.83	67.37	67.35	74.45
	IKL-KD	76.64 $\pm$ 0.02	**77.19** $\pm$ 0.01	**70.40** $\pm$ 0.03	**70.62** $\pm$ 0.08	77.16 $\pm$ 0.04

Comparing Eq. (12) and Eq. (15), Eq. (13) and Eq. (16), we conclude that DKL loss and KL loss have the same derivatives given the same inputs. Thus, DKL loss is equivalent to KL loss in terms of gradient optimization.

A.2 New state-of-the-art robustness on CIFAR-100/10

Robustbench is the most popular benchmark for adversarial robust models in the community. It evaluates the performance of models by the Auto-Attack. Auto-Attack [10] is an ensemble of different kinds of attack methods and is considered the most effective method to test the robustness of models.

We achieve new state-of-the-art robustness on CIFAR-10 and CIFAR-100 under both settings w/ and w/o generated data. As shown in Table 12, on CIFAR-100 without extra generated data, we achieve 32.53% robustness, outperforming the previous best result by **0.68%** while saving **33.3%** computational cost. With generated data, our model boosts performance to 73.85% natural accuracy, surpassing the previous best result by **1.27%** while maintaining the **strongest robustness**. More detailed comparisons can be accessed on the public leaderboard `https://robustbench.github.io/`.

Table 15: **Comparisons with strong training settings on ImageNet for knowledge distillation.**

Method	KD	DKD	DIST	IKL-KD
Top-1 Accuracy (%)	80.89	80.77	80.70	**80.98**

A.3 Comparisons on CIFAR-100 for Knowledge Distillation

We experiment on CIFAR-100 with the following cases: 1) the teacher and student models have the same unit network architectures; 2) the teacher and student models have different unit network architectures. The results are listed in Table 13 and Table 14. We have achieved the best results in 4 out of 6 and 3 out of 5 experimental settings respectively.

Moreover, we follow the concurrent work [24] and conduct experiments with BEiT-Large as the teacher and ResNet-50 as the student under a strong training scheme, the experimental results are summarized in Table 15. The model trained by IKL-KD shows slightly better results.

A.4 Other Applications with IKL

Semisupervised learning. We use the open-sourced code from `https://github.com/microsoft/Semi-supervised-learning` and conduct semi-supervised experiments on CIFAR-100 with FixMatch and Mean-Teacher methods. Specifically, each class has 2 labeled images and 500 unlabeled images. All default training hyper-parameters are used for fair comparisons. We only replace the consistency loss with our IKL loss. As shown in Table 16, with our IKL loss, the Mean-Teacher method even surpasses the FixMatch.

Table 16: **Semi-supervised Learning on CIFAR-100 with ViT-small backbone.**

Method	Pseudo-label	Consistency Loss	Last epoch Top-1 Acc(%)
FixMatch			
FixMatch	hard	Cross-entropy Loss	69.20
FixMatch	soft	Cross-entroy/KL Loss	69.09
FixMatch	soft	IKL Loss	**70.00**
Mean-Teacher			
Mean-Teacher	soft	MSE Loss	67.38
Mean-Teacher	soft	IKL Loss	**70.05**

Semantic segmentation distillation. We conduct ablation on the semantic segmentation distillation task. We use the APD [62] as our baseline for their open-sourced code. All default hyper-parameters are adopted. We only replace the original KL loss with our IKL loss. As shown in Table 17, we achieve better performance with the IKL loss function, demonstrating that the IKL loss can be complementary to other techniques in semantic segmentation distillation.

Table 17: **Semantic segmentation distillation with APD on ADE20K.**

Method	Teacher	Student	Teacher mIoU	Student mIoU
Baseline	-	ResNet-18	-	37.19
APD with KL loss	ResNet-101	ResNet-18	43.44	39.25
APD with IKL loss	ResNet-101	ResNet-18	43.44	**39.75**

A.5 Complexity of IKL

Compared with the KL divergence loss, IKL loss is required to update the global class-wise prediction scores $W \in \mathbb{R}^{C \times C}$ where C is the number of classes during training. This extra computational cost can be nearly ignored when compared with the model forward and backward. Algorithm 1 shows the implementation of our IKL loss in Pytorch style. On dense prediction tasks like semantic segmentation, Δ_a and Δ_b can require large GPU Memory. Here, we also provide the memory-efficient implementations for wMSE loss component, which is listed in Algorithm 2.

Algorithm 1 Pseudo code for DKL/IKL loss in Pytorch style.

Input: $logits_a, logits_b \in \mathbb{R}^{B \times C}$, one-hot label Y, $W \in \mathbb{R}^{C \times C}$, α, β.
class_scores = one-hot @ W;
Sample_weights = class_scores.view(-1, C, 1) @ class_scores.view(-1, 1, C);
Δ_a = $logits_a$.view(-1, C, 1) - $logits_a$.view(-1, 1, C);
Δ_b = $logits_b$.view(-1, C, 1) - $logits_b$.view(-1, 1, C);
wMSE_loss = (torch.pow(Δ_n - Δ_a, 2) * Sample_weights).sum(dim=(1,2)).mean() * $\frac{1}{4}$;
score_a = F.softmax($logits_a$, dim=1).detach();
log_score_b = F.log_softmax($logits_b$, dim=-1);
CE_loss = -(score_a * log_score_b).sum(1).mean();
return β * CE_loss + α * wMSE_loss.

Algorithm 2 Memory efficient implementation for wMSE_loss in Pytorch style.

Input: $logits_a, logits_b \in \mathbb{R}^{B \times C}$, one-hot label Y, $W \in \mathbb{R}^{C \times C}$;
class_scores = one-hot @ W;
loss_a = (class_scores * $logits_a$ * $logits_a$).sum(dim=1) * 2 - torch.pow((class_scores * $logits_a$).sum(dim=1), 2) * 2;
loss_b = (class_scores * $logits_b$ * $logits_b$).sum(dim=1) * 2 - torch.pow((class_scores * $logits_b$).sum(dim=1), 2) * 2;
loss_ex = (class_scores * $logits_a$ * $logits_b$).sum(dim=1) * 4 - (class_scores * $logits_a$).sum(dim=1) * (class_scores * $logits_b$).sum(dim=1) * 4;
wMSE_loss = $\frac{1}{4}$ * (loss_a + loss_b - loss_ex).mean();
return wMSE_loss.

A.6 Connection between IKL and the Jensen-Shannon (JS) Divergence

With the following JS divergence loss,

$$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M), \quad M = \frac{1}{2}P + \frac{1}{2}Q. \tag{17}$$

We calculate its derivatives regarding o_n (the student logits),

$$\frac{\partial \mathcal{L}_{JSD}}{\partial \mathbf{o}_n^i} = \sum_{j=1}^{C} \mathbf{w}_n^{i,j} (\Delta \mathbf{n}_{i,j} - \Delta \mathbf{m}'_{i,j}) \tag{18}$$

$$\mathit{Softmax}(o_{m'}) = \frac{1}{2}s_n + \frac{1}{2}s_m \tag{19}$$

where $\mathbf{o}_m$ is the logits from the teacher model, $\mathbf{o}_{m'}$ is a virtual logits satisfying Eq. (19), $\mathbf{s}_m = \mathit{Softmax}(\mathbf{o}_m)$, $\mathbf{s}_n = \mathit{Softmax}(\mathbf{o}_n)$, $\Delta \mathbf{m}'_{i,j} = \mathbf{o}_{m'}^i - \mathbf{o}_{m'}^j$, $\Delta \mathbf{n}_{i,j} = \mathbf{o}_n^i - \mathbf{o}_n^j$.

Correspondingly, the derivatives of IKL loss regrading o_n (the student logits),

$$\frac{\partial \mathcal{L}_{IKL}}{\partial \mathbf{o}_n^i} = \underbrace{\alpha \sum_{j=1}^{C} \mathbf{w}_m^{i,j} (\Delta \mathbf{n}_{i,j} - \Delta \mathbf{m}_{i,j})}_{\textbf{Effects of wMSE}} + \underbrace{\beta * \mathbf{s}_m^i * (\mathbf{s}_n^i - 1) + \mathbf{s}_n^i * (1 - \mathbf{s}_m^i)}_{\textbf{Effects of Cross-Entropy}} \tag{20}$$

Compared with IKL loss, the problem for JSD divergence in knowledge distillation is that: *The soft labels from the teacher models often embed dark knowledge and facilitate the optimization of the student models. However, there are no effects of the cross-entropy loss with the soft labels from the teacher model, which can be the underlying reason that JSD is worse than KD and IKL-KD.*

As shown in Table 18, we also empirically demonstrate that IKL loss performs better than JSD divergence on the knowledge distillation task.

Table 18: **Comparisons between KL, IKL, and JSD on ImageNet-LT**.

Method	Student	Teacher	Teacher Acc(%)	Student Acc(%)
Self-distillation on Imbalanced Data				
KL	ResNet-50	ResNet-50	45.47	47.04
JSD	ResNet-50	ResNet-50	45.47	46.64
Ours	ResNet-50	ResNet-50	45.47	**47.50**
Knowledge distillation on Imbalanced Data				
KL	ResNet-50	ResNeXt-101	48.33	48.31
JSD	ResNet-50	ResNeXt-101	48.33	47.82
Ours	ResNet-50	ResNeXt-101	48.33	**49.22**

A.7 Licenses

All the datasets we considered are publicly available, we list their licenses and URLs as follows:

- **CIFAR-10** [41]: MIT License, `https://www.cs.toronto.edu/~kriz/cifar.html`.
- **CIFAR-100** [41]: MIT License, `https://www.cs.toronto.edu/~kriz/cifar.html`.
- **ImageNet** [54]: Non-commercial, `http://image-net.org`.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We theoretically prove that the Kullback-Leibler (KL) Divergence loss is equivalent to a Decoupled Kullback-Leibler (DKL) loss regarding gradient optimization. Based on this analysis, we improve KL/DKL loss on adversarial training and knowledge distillation tasks.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The conclusion and limitation section is included.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Our proof to Theorem 1 is in Appendix A.1.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide detailed experimental settings in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide a link for our code and models in the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide detailed experimental settings in Section 4 and ablations for hyper-parameters in the Appendix 4.4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Error bars are included in Table 14 of the Appendix. Due to the heavy computation, other tables don't provide error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: On adversarial training, Each run with basic augmentations takes around 2 days using 4GPUs while 5 days using 8 GPUs for adversarial training with generated data. On knowledge distillation, 8 Nvidia GeForce 3090 GPUs are used on ImageNet. Each run takes about 1 day for our method.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research conforms NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our paper is fundamental research in adversarial robustness and knowledge distillation and there is no obvious societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our models don't suffer from a high risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Proper citations are added to the paper. Licenses for used data is included in Appendix A.7.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Our paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.