REVISITING FAST ADVERSARIAL TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Fast Adversarial Training (FAT) not only improves the model robustness but also reduces the training cost of standard adversarial training. However, FAT often suffers from Catastrophic Overfitting (CO), which results in poor robustness performance. CO describes the phenomenon that model robust accuracy can decrease dramatically and suddenly during the training of FAT. Many effective techniques have been developed to prevent CO and improve the model robustness from different perspectives. However, these techniques adopt inconsistent training settings and require different training costs, *i.e,* training time and memory costs, resulting in an unfair comparison. In this paper, we first conduct a comprehensive study of more than 10 FAT methods in terms of adversarial robustness and training costs. We revisit the effectiveness and efficiency of FAT techniques in preventing CO from the perspective of model local nonlinearity and propose an effective Lipschitz regularization method for FAT. Furthermore, we explore the effect of data augmentation and weight averaging in FAT and propose a simple yet effective auto weight averaging method to improve robustness further. By assembling these techniques, we propose a **FGSM**-based fast adversarial training method equipped with **L**ipschitz regularization and **A**uto **W**eight averaging, abbreviated as FGSM-LAW. Experimental evaluations on four benchmark databases demonstrate the superiority of the proposed method.

1 INTRODUCTION

Deep Neural Networks (DNNs) can be easily fooled by Adversarial Examples (AEs) which are carefully crafted by adding imperceptible perturbations to benign samples. Adversarial Training (AT) has been demonstrated as one of the most effective methods to improve the model robustness against AEs. In Standard Adversarial Training (SAT) (Wang et al., 2019; Zhang et al., 2020; Cui et al., 2021), multi-step attack methods such as Projected Gradient Descent (PGD) are often applied to generate AEs, which are computationally costly. This makes it challenging to apply AT on large-scale datasets. To reduce training expenses, Fast Adversarial Training (FAT) (Tramèr et al., 2017) has been proposed. FAT applies a single-step attack such as Fast Gradient Sign Method (FGSM) to approximate the multi-step attack for AE generation. Recent work (Wong et al., 2020) has revealed that FGSM-AT does not increase the model's resilience against AEs. In detail, Wong et al. (2020) first discover that FGSM-based AT is prone to Catastrophic Overfitting (CO), in which model robustness increases at the beginning of FGSM-AT but sharply falls down after a few training epochs.

Some FAT variants (Kim et al., 2021; Jia et al., 2022c) have been proposed to prevent CO and improve the model robustness. Sample initialization (Wong et al., 2020; Jia et al., 2022c) and loss regularization (Sriramanan et al., 2020) are two commonly used strategies to prevent CO and improve the model robustness. Sample initialization (Zheng et al., 2020; Jia et al., 2022a) improves the quality of the AEs by generating better sample initialization. Loss regularization (Sriramanan et al., 2021) adopts regularity to guide the generation of stronger adversaries and smooths the loss surface for better robustness. Although these techniques can solve CO effectively, we find some of them introduce more training burden, contrary to the original intention of FAT. Such inconsistent training time and memory requirements also result in an unfair comparison.

To have a comprehensive study, we investigate the used techniques of more than tens of papers working on the FAT methods, which are summarized in Appendix A. We find the robustness improvement of these techniques is still correlated with training costs even in FAT. For example, Andriushchenko & Flammarion (2020) propose a gradient alignment regularization method that can

relieve CO and improve the model robustness. But it requires high consumption of time to calculate the regular term, which requires extra computational cost. And Zheng et al. (2020) propose to use the adversarial perturbations from the last epoch as the sample initialization to improve the model robustness. This method requires extra computational cost and high consumption of memory to restore the previous perturbations. It would be important to comprehensively conduct empirical studies of evaluating different FAT-related techniques quantitatively.

In this paper, we focus on exploring and finding efficient and effective FAT-related strategies. We roughly divide existing FAT-related techniques into two categories: sample initialization and regularization. In addition, we also explore the effect of several techniques on the robustness improvement for FAT, *i.e,* data augmentation, and weight averaging. We conduct extensive experiments to evaluate the effectiveness and computational cost of various techniques in each category and conclude the following findings:

- **Sample Initialization** Prior/learning-based initialization prevents CO and achieves better model robustness, which requires more training costs. Random sample initialization with an appropriate step size also prevents CO but achieves limited model robustness improvement without extra training cost.
- **Regularization** Regularization is unnecessary during the inner maximization step in FAT, but for outer loss minimization, regularization is critical.
- **Data Augmentation** Different from SAT (Rebuffi et al., 2021), FAT can achieve better model robustness when equipped with several sophisticated data augmentations.
- **Weight Averaging** In contrast to SAT (Wang & Wang, 2022), FAT cannot boost robustness by simply introducing the weight averaging technique.

Based on the findings above, a) for sample initialization, considering the time consumption, we adopt random initialization. b) For regularization, motivated by Lipschitz constraint, we propose a novel regularization approach only use in minimization optimization to achieve better performance and less time overhead. c) For data augmentation, we find data augmentation can improve the quality of AEs and adopt the Cutout for FAT. d) For weight averaging, we discover that the reason WA fails on FAT is that model parameters accumulate non-robust parameters trained on the low-quality AEs and propose a simple yet effective WA method to automatically select the robust parameters to conduct WA. By assembling the above techniques, we conclude our FAT method equipped with **L**ipschitz regularization and **A**uto **W**eight averaging, called FGSM-LAW. Our main contributions are in three aspects: **1)** We explore existing FAT-related techniques and find efficient and effective FAT techniques for further research. **2)** We propose a novel regularization approach motivated by the Lipschitz constraint, which can further improve the model robustness for FAT. **3)** We find WA fails to boost FAT and propose a simple yet effective WA method for FAT to improve model robustness further. **4)** By assembling these techniques, we conclude our fast adversarial training method, called FGSM-LAW. Compared with state-of-the-art FAT methods across various network architectures and datasets, FGSM-LAW achieves higher robustness performance with less training costs.

2 Related Work

2.1 Adversarial Attack Methods

DNNs are vulnerable to AEs. Several studies (Gu et al., 2021; Wang & He, 2021) focus on generating AEs. In detail, Goodfellow et al. (2014) propose to adopt Fast Gradient Sign Method (FGSM) to attack DNNs, which makes use of the model gradient to generate AEs. Projected Gradient Descent (PGD) known as a strong adversarial attack method, is proposed by Madry et al. (2017). It is an iterative extension of FGSM, which conducts FGSM with the project operation multiple times in a small attack step. Carlini & Wagner (2017) introduce three distance metrics that are used in the previous literature and propose some powerful adversarial attack methods, called C&W. Andriushchenko et al. (2020) propose a score-based black-box adversarial attack method based on a random search, called Square. Croce & Hein (2020a) propose a fast adaptive boundary attack method that aims to find the minimal adversarial perturbation to attack DNNs, called FAB. Moreover, Croce & Hein (2020b) propose two powerful adversarial attack methods (APGD-CE and APGD-DLR), which can

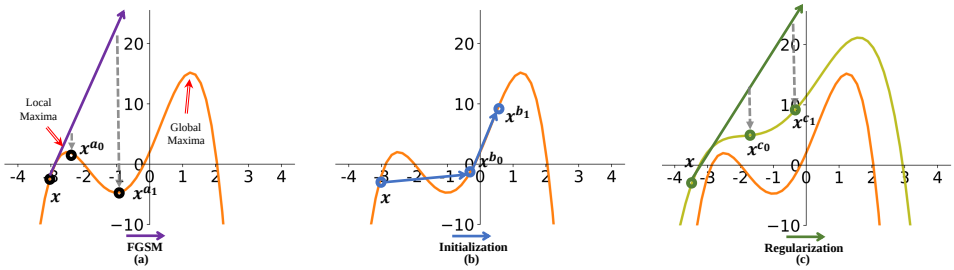


Figure 1: Generation process of adversarial examples generated by FGSM in FAT on the maximization loss function. (a) Using a zero initialization. (b) Using the sample initialization. (c) Using the regularization.

automatically select the attack step size. Further, they combine the two previous adversarial attack methods (Square and FAB) with their own attack methods, called AutoAttack (AA).

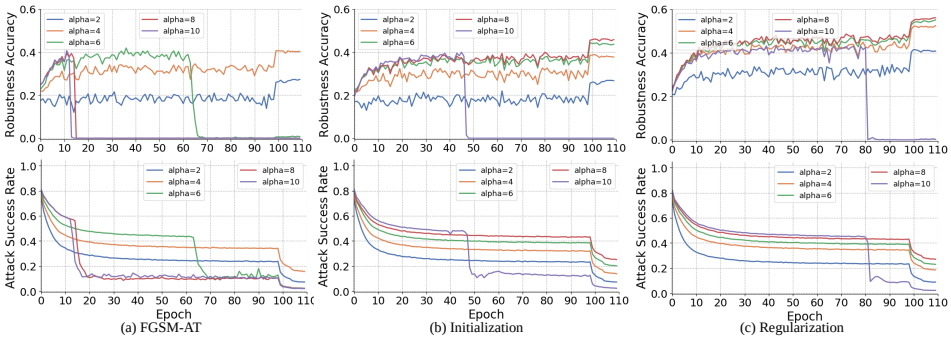
2.2 Fast Adversarial Training

Even though adversarial training (Zhang et al., 2019; Wu et al., 2020; Cui et al., 2021; Jia et al., 2022b) have been proposed to be one of the most effective defense methods, it requires high costs to generate AEs, resulting in its limited practical application. To reduce the training costs, Goodfellow et al. (2014) propose to adopt FGSM to generate AEs for training, called FGSM-AT. Tramèr et al. (2017) propose to initialize the clean samples with the Gaussian distribution to conduct FGSM-AT, called R+FGSM. Wong et al. (2020) indicate that both FGSM-AT and R+FGSM easily encounter catastrophic overfitting (CO) and propose to initialize the clean samples with the uniform distribution instead of Gaussian distribution to conduct FGSM-AT, called FGSM-RS. Several FAT methods are proposed to improve the model robustness from the regularization perspective. In detail, Andriushchenko & Flammarion (2020) propose a gradient alignment regularization to prevent CO and improve the model robustness. Also, Sriramanan et al. (2020) propose a ℓ_2 distance regularization to improve the quality of AEs. Sriramanan et al. (2021) propose a Nuclear-Norm regularization to enforce function smoothing. Moreover, some research focuses on sample initialization to improve the performance of FGSM-AT. Specifically, Jia et al. (2022c) propose a learnable sample initialization generated by a generative model to boost FAT. Jia et al. (2022a) propose to use the prior adversarial perturbation as the sample initialization to improve robustness performance.

3 Techniques to Prevent Catastrophic Overfitting

3.1 Rethinking Catastrophic Overfitting

Catastrophic overfitting (CO) indicates the phenomenon that the robust accuracy dramatically and suddenly decreases during the training parse of FAT, which is first discovered by Wong et al. (2020). A series of FAT variants have been proposed to prevent CO and improve the robustness from two aspects, *i.e.,* sample initialization and regularization. Rethink that FGSM attack can be regarded as a closed-form solution to the maximization optimization problem, *i.e.,* $\boldsymbol{\delta} = \arg\max_{\|\boldsymbol{\delta}\|_\infty < \epsilon} \langle \nabla_{\mathbf{x}} \ell(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})), \boldsymbol{\delta} \rangle$. If the loss function is locally linear, the output of $\nabla_{\mathbf{x}} \ell(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})$ is constant within the ℓ_∞-ball around the input sample $\mathbf{x}$, which can provide the attacker with the optimal adversarial perturbation to attack the model. On the contrary, if the loss function is locally nonlinear, as shown in Fig 1 (a), the generated AEs based on one large step FGSM may not be able to reach the local maximum region of the loss function, resulting in low-quality AEs. Jia et al. (2022a;c) have indicated that CO happens when the AE quality becomes worse. As shown in Fig 1 (b), better initialization can improve the quality of AEs, which can reduce the effect of local nonlinearity to prevent CO. Adding regularization to the loss function is another way of reducing local nonlinearity. In detail, as shown in Fig 1 (c), using the regularization can promote the flatness of the loss surface, which mitigates local nonlinearity and meanwhile improves the quality of AEs generated by FGSM. Hence, CO is directly related to the quality of the solution to the inner maximization and it is intrinsically caused by the model's local non-linearity. Sample initialization and regularization improve the quality of the AEs in different ways to prevent CO.



(a) FGSM-AT (b) Initialization (c) Regularization

Figure 2: The robustness accuracy under PGD-10 and attack success rate of FGSM-AT, FGSM with Bernoulli random initialization and FGSM with the regularization of different step sizes on the CIFAR-10 during the training phase.

3.2 Sample Initialization

The sample initialization used in FAT is quite different. For a given clean sample $\mathbf{x}$ initialized by $\boldsymbol{\eta}$, FAT methods adopt FGSM to generate the adversarial perturbation $\boldsymbol{\delta}$. It can be defined as:

$$\boldsymbol{\delta} = \Pi_{[-\epsilon,\epsilon]} \left[\boldsymbol{\eta} + \alpha \cdot \operatorname{sign} \left(\nabla_{\mathbf{x}} \mathcal{L}(f(\mathbf{x} + \boldsymbol{\eta}; \boldsymbol{\theta}), \mathbf{y}) \right) \right], \tag{1}$$

where ϵ represents the maximal perturbation, α represents the attack step size, $f(\cdot; \boldsymbol{\theta})$ represents the trained model with the weight parameters $\boldsymbol{\theta}$, $\mathbf{y}$ represents the ground truth label. And the $\mathcal{L}(f(\mathbf{x} + \boldsymbol{\eta}; \boldsymbol{\theta}), \mathbf{y})$ is the loss function for training. We summarize existing initialization methods used in FAT and divide them into three categories: **a) random-based initialization, b) prior-based initialization and c) learning-based initialization**. More details are presented in Appendix B.

Random-based Initialization Most FAT methods (Goodfellow et al., 2014; Tramèr et al., 2017; Sriramanan et al., 2021) adopt random-based initialization methods to conduct AT. They always use a random sample-independent distribution to initialize the sample for the AE generation, *i.e.,* $\boldsymbol{\eta} \sim \mathrm{Random}(\epsilon)$, where $\mathrm{Random}(\epsilon)$ represents a random distribution related to the maximum perturbation strength ϵ. In detail, Goodfellow et al. (2014) conduct FGSM with a zero initialization to generate AEs, *i.e.,* $\boldsymbol{\eta} = 0$. Tramèr et al. (2017) apply FGSM with a normal initialization in the half perturbation to generate AEs for training, *i.e.,* $\boldsymbol{\eta} = \frac{\epsilon}{2} \cdot \mathrm{Normal}(0, 1)$, called FGSM-NR. Wong et al. (2020) propose to implement FGSM with a uniform initialization in the whole perturbation to generate AEs for training, *i.e.,* $\boldsymbol{\eta} = \epsilon \cdot \mathrm{Uniform}(-1, 1)$, called FGSM-RS. Moreover, in the recent research (Sriramanan et al., 2020; 2021), they adopt a Bernoulli initialization in the half perturbation to perform FAT, *i.e.,* $\boldsymbol{\eta} = \frac{\epsilon}{2} \cdot \mathrm{Bernoulli}(-1, 1)$, called FGSM-BR.

Prior-based Initialization Several works (Zheng et al., 2020; Jia et al., 2022a) also propose to use the previous adversarial perturbations as the sample initialization, which can be called prior-based initialization, *i.e.,* $\boldsymbol{\eta} \sim \mathrm{Prior}(\epsilon, \mathbf{x})$, where $\mathrm{Prior}(\epsilon, \cdot)$ represents the function related to the maximum perturbation strength ϵ and the sample $\mathbf{x}$, *i.e.,* sample-dependent initialization. Zheng et al. (2020) propose to adopt the previous adversarial perturbation from the last epoch to initialize the clean sample, *i.e.,* $\boldsymbol{\eta} = \boldsymbol{\delta}_E$, called ATTA. And Jia et al. (2022a) propose to use some prior-guided adversarial perturbations as the adversarial initialization which is prior from the previous epoch and the momentum of all previous epochs to perform FGSM , *i.e.,* $\boldsymbol{\eta} = \boldsymbol{\delta}_P$, called FGSM-PGI.

Learning-based Initialization Jia et al. (2022c) propose to adopt a sample-dependent learnable initialization to conduct AT. In detail, they adopt a generative model to generate the sample initialization for training, called FGSM-SDI, *i.e.,* $\boldsymbol{\eta} = \alpha \cdot g\left(\mathbf{x}, \nabla_x \mathcal{L}(\mathbf{x}, \mathbf{y}); \mathbf{w}\right)$, where α is set to ϵ for training, and $g\left(\cdot, \cdot; \mathbf{w}\right)$ represents the generative model with the weight parameters $\mathbf{w}$.

Discussion Previous works adopt different sample initialization for AT to prevent CO and improve the model robustness. As shown in Fig. 1, using a zero initialization, FGSM with a large step size obtains a worse solution to the inner maximization problem. But, using sample initialization, FGSM can achieve a better solution to the inner maximization problem. The quality of the solutions to the inner maximization problem is related to the CO. As shown in Fig. 2, FGSM-AT with a small step size (e.g, $\alpha = 4/255$) does not meet CO and the attack success rate and model robustness are directly related in FAT. It indicates that when the quality of the inner maximization problem becomes worse, CO could happen. Using a zero initialization, FGSM with a larger step size more

Table 1: Test robustness (%) on the CIFAR-10 database using ResNet18 with different sample initialization techniques. Number in bold indicates the best.

Method	Type	Clean Best/Last	PGD-50 Best/Last	C&W Best/Last	AA Best/Last	Time
FGSM-NR	Random-based	84.87/86.41	45.21/43.99	45.88/45.4	41.9/41.29	1 ×
FGSM-RS	Random-based	84.91/86.65	45.43/44.12	45.66/45.65	42.77/42.2	1 ×
FGSM-BR	Random-based	85.38/**86.82**	45.57/44.23	46.46/45.89	42.99/42.37	1 ×
ATTA	Prior-based	85.97/86.30	45.97/44.97	44.6/43.8	43.39/42.84	1 ×
FGSM-PGI	Prior-based	**86.33**/86.61	46.71/45.69	45.5/44.8	43.99/43.26	1 ×
FGSM-SDI	Learning-based	84.76/85.25	**51.79/51.79**	**51.0/50.29**	**48.50/47.91**	1.6 ×

easily meets CO. Using sample initialization can prevent CO when FGSM with a large step size. We summarize the existing initialization methods and make some meaningful findings based on the quality of the solutions to the inner maximization problem. First, sample-dependent initialization methods (prior-based and learning-based initialization) can achieve better robustness performance than sample-independent initialization methods (random-based initialization). The learning-based initialization achieves the best robustness performance. Second, random-based initialization methods achieve approximately the same robustness improvement performance, which is better than zero initialization. We adopt ResNet18 for AT with different initialization to conduct experiments on CIFAR10. The result is shown in Table 1, which demonstrates the correctness of our findings. Compared with the previous random initialization, the advanced ATTA and FGSM-PGI need more memory to store the historical adversarial perturbations, which could require more memory costs. And FGSM-SDI needs more time consumption to train an extra generator. We use random initialization based on the Bernoulli initialization for FAT since it is effective and comparable robustness.

3.3 Regularization

Regularization used in AT can be divided into two categories: a) regularization worked in the inner maximization and b) regularization worked in the outer minimization. The former is related to the inner maximization to improve the quality of the AEs. The latter is related to the outer minimization to improve the model robustness. Existing FAT variants adopt the regularization in min-max optimization or only in minimization optimization. It can be defined as:

$$\boldsymbol{\delta}^* = \text{argmax}_{\boldsymbol{\delta}\in\Omega}[\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}),\mathbf{y}) + \gamma \cdot R(\mathbf{x},\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta})] \tag{2}$$

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \left[\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}^*;\boldsymbol{\theta}),\mathbf{y}) + \mu \cdot R(\mathbf{x},\mathbf{x}+\boldsymbol{\delta}^*;\boldsymbol{\theta})\right], \tag{3}$$

where $\gamma, \mu \in [0, 1]$ control the use of regularization and $R(\mathbf{x}, \mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta})$ represents the regularization.

Regularization only in minimization optimization Andriushchenko & Flammarion (2020) propose a gradient alignment regularization to measure the local linearity. They adopt the regularization only in the minimization optimization, *i.e.,* $\gamma = 0$ and $\mu = 1$. It can be defined as:

$$R\left(\mathbf{x},\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}\right) = 1 - \cos\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x};\boldsymbol{\theta}),\mathbf{y}), \nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\eta};\boldsymbol{\theta}),\mathbf{y})\right), \tag{4}$$

where $\boldsymbol{\eta}$ represents the sample initialization. Though the FGSM-GA achieves model robustness improvement, it requires extra training time to compute the alignment regularization on the gradient.

Regularization in min-max optimization Sriramanan et al. (2020) propose a more effective guided regularization for improved optimization and function smoothing. They adopt regularization in min-max optimization. It works in the inner maximization to generate AEs and the outer minimization to improve the model robustness,*i.e.,* $\gamma = 1$ and $\mu = 1$. It can be defined as:

$$R(\mathbf{x},\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) = \lambda \cdot \|f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) - f(\mathbf{x};\boldsymbol{\theta})\|_2^2, \tag{5}$$

where λ represents the a hyper-parameter that determines the smoothness of the loss surface. Sriramanan et al. (2021) propose a Nuclear-Norm regularization for enforcing function smoothing in min-max optimization, *i.e.,* $\gamma = 1$ and $\mu = 1$. It can be defined as:

$$R(\mathbf{x},\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) = \lambda \cdot \|f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) - f(\mathbf{x};\boldsymbol{\theta})\|_*, \tag{6}$$

where $\|\cdot\|_*$ represents the Nuclear Norm which is the sum of the singular values.

Table 2: Test robustness (%) on the CIFAR-10 database using ResNet18 with different regularization methods. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time
Baseline	86.82	46.62	44.76	44.23	45.89	42.37	1 ×
Guided Regularization(min-max)	83.33	52.04	51.03	50.96	48.64	46.56	2×
Guided Regularization(min)	84.20	52.38	51.54	51.26	47.87	45.74	1.4×
Nuclear-Norm Regularization(min-max)	84.9	51.21	50.09	49.79	49.14	47.36	2×
Nuclear-Norm Regularization(min)	**84.94**	51.94	51.0	50.78	48.83	47.06	1.4×
Ours(min)	82.97	**55.49**	**54.58**	**54.29**	**50.41**	**48.05**	1.4×

Discussion We conclude that the regularization in min-max optimization achieves better robustness. It greatly improves model robustness. In inner maximization, the regularization loss is used to find the least smooth domain of the loss surface. In outer minimization, the regularization is used to improve the local smoothness of the loss surface. Although the regularization is effective to improve the model robustness, it also requires a computation overhead. On the other hand, AEs almost exist in the least smooth domain of the loss surface (Rebuffi et al., 2021; Wu et al., 2020) which has the same effect as inner maximization in regularization. Based on this, we argue that it is not necessary to use regularization in the inner maximization. To verify this opinion, we conduct experiments on CIFAR-10 with ResNet18 comparing regularization in the min-max optimization and regularization solely in the minimization. We adopt FGSM with BR initialization as the baseline. The results are shown in Table 2. Compared with FAT with the regularization in min-max optimization, FAT with the regularization only in the minimization achieves better performance under PGD attack scenarios and comparable robustness performance under C&W and AA attack scenarios. The regularization only in the minimization requires less training time.

3.4 The Proposed Regularization

Previous works (Weng et al., 2018; Wu et al., 2021) have demonstrated that the robustness is related to the local Lipschitzness, *i.e.,* smaller local Lipschitzness leads to stronger robustness. We propose a novel regularization approach motivated by the Lipschitz constraint. It can be defined as:

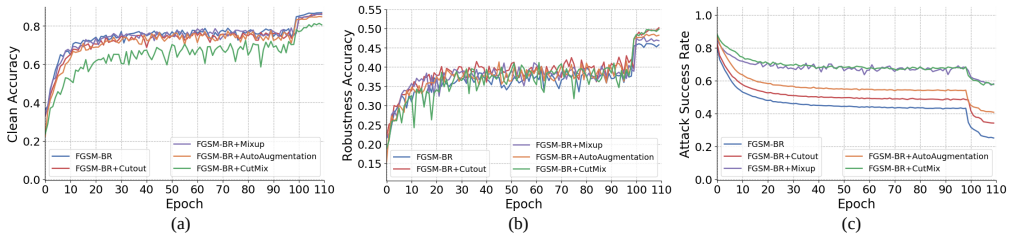
$$\frac{\lambda \cdot (\|f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) - f(\mathbf{x}+\boldsymbol{\eta};\boldsymbol{\theta})\|_2^2 + \|f_F(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) - f_F(\mathbf{x}+\boldsymbol{\eta};\boldsymbol{\theta})\|_2^2)}{\|\boldsymbol{\delta}-\boldsymbol{\eta}\|_2^2}, \tag{7}$$

where $f_F(\cdot;\boldsymbol{\theta})$ represents the feature output of $f(\cdot;\boldsymbol{\theta})$ and λ is the hyper-parameter. We also compared the proposed regularization with the previous regularization methods, the results are shown in Table 2. The proposed regularization achieves the best performance under all attack scenarios. We provide proof to prove the effectiveness of the proposed regularization in Appendix C.

4 Techniques to Improve Model Robustness

4.1 Data Augmentation

Several works (Gowal et al., 2020; Rebuffi et al., 2021) have explored the impact of more sophisticated data augmentation techniques on the robustness of PGD-based AT. However, the impact of the data augmentation techniques on model robustness of FGSM-based AT has not been studied before. In this paper, we introduce the sophisticated data augmentation techniques (Cutout (DeVries & Taylor, 2017), Mixup (Zhang et al., 2017), CutMix (Yun et al., 2019), and AutoAugment (Cubuk et al., 2018)) into existing FAT techniques and explore the impact of them on model robustness of FAT methods. In detail, we use FGSM-BR as the baseline and compare it with FGSM-BR with data augmentation techniques. The results are shown in Fig. 3. More results are presented in **Appendix D**. It can be observed that combined with data augmentation, FAT can achieve better robustness performance in all the attack scenarios. Particularly, under the PGD-10 attack, Cutout improves the performance by about 1.5% without additional training costs. Compared with the vanilla FGSM-BR, using data augmentation methods can improve the quality of the solution to the inner maximization problem, *i.e.,* the higher attack success rate. Unlike PGD-based AT (Rebuffi et al., 2021), using data augmentation techniques can improve the robustness of FGSM-based AT.



(a) (b) (c)

Figure 3: The performance of FAT methods with different data augmentation methods on CIFAR10 during the training phase. (a): The clean accuracy on the clean images. (b): The robustness accuracy under PGD-10. (c): The attack success rate on the AEs used for training.

Table 3: Test robustness (%) on the CIFAR-10 database with different model weight averaging techniques. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time
Baseline	86.82	46.62	44.76	44.23	45.89	42.37	1 ×
Vanilla EMA Rebuffi et al. (2021)	**86.91**	47.1	45.64	45.1	46.53	42.8	1 ×
SE-EMA Wang & Wang (2022)	86.82	47.42	45.71	45.026	46.68	42.92	1 ×
Auto-EMA(Ours)	83.62	**51.26**	**50.21**	**49.9**	**49.42**	**46.53**	1 ×

4.2 Model Weight Averaging

As a technique to improve the model generalization, model weight averaging (WA) (Izmailov et al., 2018) obtain a WA model by computing exponentially weighted moving averages of model parameters of each training iteration. Formally, given a decay rate τ and the model parameters $\boldsymbol{\theta}$ of an iteration, the WA model is updated with $\tilde{\boldsymbol{\theta}} \leftarrow \tau \cdot \tilde{\boldsymbol{\theta}} + (1-\tau) \cdot \boldsymbol{\theta}$. The obtained WA model $\tilde{\boldsymbol{\theta}}$ at the training end often generalizes better to the unseen test data. Recent work (Chen et al., 2020; Wu et al., 2020) has indicated that the WA model shows higher robustness by achieving a flatter adversarial loss landscape. In particular, Wang & Wang (2022) show that the WA model obtained in SAT also shows higher adversarial robustness. However, the impact of WA on the FAT has not been explored in previous works. In this work, we make the first exploration by combining the FGSM-BR with the vanilla EM (Rebuffi et al., 2021) and SE-EMA (Wang & Wang, 2022). As shown in the Table 3, simply applying the existing WA techniques to FAT does not improve the robustness.

4.3 The Proposed Model Weight Averaging

In this work, we attribute the failure of WA on FAT to the weak quality of AEs generated by FGSM. The existing model weight averaging (WA) will accumulate model parameters of all iterations regardless of the quality of AEs. However, as discussed in Sec. 3.1, the gradient directions can be undesired when the quality of AEs in FAT is low. The model parameter updates with the inaccurate gradients can be accumulated in WA model.

To overcome the shortcoming of the existing WA methods, we propose to exclude some training iterations from the WA model update trajectory. Namely, the WA model will not update in an iteration as long as the quality of AEs in the iteration is low. Specifically, we propose a simple yet effective metric to automatically evaluate the quality of AEs. The metric will be used to determine whether the WA model will be updated in the current training iteration. Concretely, the metric is defined as the ratio of the robust accuracy on AEs to the corresponding clean accuracy to evaluate the AE quality. Formally, it can be formulated as:

$$\Delta = \frac{Acc(f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}),\mathbf{y})}{Acc(f(\mathbf{x};\boldsymbol{\theta}),\mathbf{y})}, \tag{8}$$

where $Acc(\cdot)$ is the accuracy of trained model on training samples. When the proposed metric Δ is above a certain threshold T, the trained parameters $\boldsymbol{\theta}$ is added to the average model parameters $\tilde{\boldsymbol{\theta}}$, *i.e,* $\tilde{\boldsymbol{\theta}} \leftarrow \tau \cdot \tilde{\boldsymbol{\theta}} + (1-\tau) \cdot \kappa \cdot \boldsymbol{\theta}$, where $\kappa = \mathbf{1}(\Delta <= T)$. In summary, our proposed method improves EMA by automatically selecting iterations to update WA model, called Auto-EMA. We conduct FGSM-BR with our Auto-EMA and compare it with Vanilla EMA and SE-EMA. The result is shown in Table 3. The proposed Auto-EMA achieves the best robustness under all attack scenarios.

Table 4: Test robustness (%) on the CIFAR-10 database. Number in bold indicates the best.

Method	Clean Best/Last	PGD-10 Best/Last	PGD-20 Best/Last	PGD-50 Best/Last	C&W Best/Last	AA Best/Last	Time
PGD-AT	82.32/82.65	53.76/53.39	52.83/52.52	52.6/52.27	51.08/51.28	48.68/48.93	5.2 ×
FGSM-RS	73.81/83.82	42.31/0.09	41.55/0.04	41.26/0.02	39.84/0.00	37.07/0.00	1 ×
FGSM-PGI	81.72/81.72	55.18/55.18	54.36/54.36	54.17/54.17	50.75/50.75	49.0/49.0	1.4 ×
FGSM-CKPT	**90.29/90.29**	41.96/41.96	39.84/39.84	39.15/39.15	41.13/41.13	37.15/37.15	1.5 ×
FGSM-SDI	84.86/85.25	53.73/53.18	52.54/52.05	52.18/51.79	51.0/50.29	48.50/47.91	1.6 ×
NuAT	81.58/81.38	53.96/53.52	52.9/52.65	52.61/52.48	51/3/50.63	49.09/48.70	2.0 ×
GAT	79.79/80.41	54.18/53.29	53.55/52.06	53.42/51.76	49.04/49.07	47.53/46.56	2.0 ×
FGSM-GA	83.96/84.43	49.23/48.67	47.57/46.66	46.89/46.08	47.46/46.75	43.45/42.63	3.5 ×
Free-AT (m=8)	80.38/80.75	47.1/45.82	45.85/44.82	45.62/44.48	44.42/43.73	42.17/41.17	4.2 ×
FGSM-LAW (ours)	80.46/80.46	**57.33/57.33**	**56.83/56.83**	**56.72/56.72**	**51.76/51.76**	**49.36/49.36**	1.4 ×

5 EXPERIMENTS

To evaluate the proposed FGSM-LAW, We adopt several datasets, *i.e,* CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100 (Krizhevsky et al., 2009), Tiny ImageNet (Deng et al., 2009) and ImageNet (Deng et al., 2009), which are widely used to evaluate the model robustness. The experimental results on Tiny ImageNet and ImageNet are presented in Appendix H.

5.1 DEFAULT TRAINING SETTING

Following the default settings of AT (Pang et al., 2020; Jia et al., 2022c), we apply the same training hyper-parameters to conduct experiments. All models with ReLU activation function are trained for 110 epochs with the batch size 128. We use SGD momentum optimizer with the weight decay of 5×10^{-4}. The learning rate initialized to 0.1 decays with a factor of 0.1 at 100 and 105 epochs. All experiments are conducted under the ℓ_∞ distance metric with the maximal perturbation of $8/255$. We adopt PGD attack of 10-steps (**PGD-10**), 20-steps (**PGD-20**), 50-steps (**PGD-50**), C&W attack of 20 steps (**C&W**) and AutoAttack (**AA**) to evaluate the trained models. The comparative experiments with a cyclic learning rate strategy Smith (2017) are presented in Appendix E. The detailed hyper-parameter settings of FGSM-LAW are presented in Appendix F.

5.2 COMPARISONS WITH STATE-OF-THE-ART METHODS

To evaluate the proposed method, we compare the proposed method with several state-of-the-art FAT methods *i.e,* Free-AT(Shafahi et al., 2019), FGSM-RS (Wong et al., 2020), FGSM-GA (Andriushchenko & Flammarion, 2020), GAT (Sriramanan et al., 2020), FGSM-CKPT (Kim et al., 2021), NuAT (Sriramanan et al., 2021), FGSM-SDI (Jia et al., 2022c), FGSM-PGI (Jia et al., 2022a) and an advanced SAT method (*i.e,* PGD-AT (Rice et al., 2020)). Note that we adopt the training settings reported in their original works to train these AT models.

Results on CIFAR-10 As for CIFAR-10, we use ResNet18 as the backbone. Note that the experiment results of WideResNet34-10 are presented in Appendix G. The results are shown in Table 4. It can be observed that compared with the advanced PGD-AT, the proposed method achieves better model robustness under all attack scenarios and requires much less training time. Compared with other advanced FAT methods, as for the best and last checkpoints, the proposed method achieves the best model robustness under all attack scenarios. Specifically, under the PGD-50 attack, the previous most effective FAT methods achieve an accuracy of about 54%, while the proposed method achieves an accuracy of about 56%. In terms of the training cost, the proposed method requires the minimum training time except for FGSM-RS which meets CO. The training time of the proposed method is the same as FGSM-PGI, which is about 1.1 times faster than FGSM-CKPT, 1.2 times faster than FGSM-SDI, 1.4 times faster than GAT and NuAT, 2.5 times faster than FGSM-GA and 3 times faster than Free-AT. FGSM-PGI requires more memory costs to store the historical adversarial perturbations. But the proposed method requires the same memory cost as the other FAT methods. Different from previous FAT variants, the proposed method not only prevents CO but also improves the model robustness without extra training time and memory cost.

Table 5: Test robustness (%) on the CIFAR-100 database. Number in bold indicates the best.

Method	Clean Best/Last	PGD-10 Best/Last	PGD-20 Best/Last	PGD-50 Best/Last	C&W Best/Last	AA Best/Last	Time
PGD-AT	57.52/57.5	29.6/29.54	28.99/29.0	28.87/28.9	28.85/27.6	25.48/25.58	5.2 ×
FGSM-RS	49.85/60.55	22.47/0.45	22.01/0.25	21.82/0.19	20.55/0.25	18.29/0.00	1 ×
FGSM-PGI	58.78/58.81	31.78/31.6	31.26/31.06	31.14/30.88	28.06/27.72	25.67/25.42	1.4 ×
FGSM-CKPT	**60.93/60.93**	16.58/16.69	15.47/15.61	15.19/15.24	16.4/16.6	14.17/14.34	1.5 ×
FGSM-SDI	60.67/60.82	31.5/30.87	30.89/30.34	30.6/30.08	27.15/27.3	25.23/25.19	1.6 ×
NuAT	59.71/59.62	27.54/27.07	23.02/22.72	20.18/20.09	22.07/21.59	11.32/11.55	2.0 ×
GAT	57.01/56.07	24.55/23.92	23.8/23.18	23.55/23.0	22.02/21.93	19.60/19.51	2.0 ×
FGSM-GA	54.35/55.1	22.93/20.04	22.36/19.13	22.2/18.84	21.2/18.96	18.88/16.45	3.5 ×
Free-AT (m=8)	52.49/52.63	24.07/22.86	23.52/22.32	23.36/22.16	21.66/20.68	19.47/18.57	4.2 ×
FGSM-LAW (ours)	58.79/58.79	**31.8/31.8**	**31.35/31.35**	**31.33/31.33**	**28.47/28.47**	**25.78/25.78**	1.4 ×

Results on CIFAR-100 As for CIFAR-100, ResNet18 is used as the backbone network. The result is shown in Table 5. Compared with PGD-AT, the proposed method achieves better model robustness under most attack scenarios and costs less training time. Even under AA attack, PGD-AT obtains an accuracy of about 25.48%, but our method obtains an accuracy of about 25.78%. Our method achieves the best model robustness under all attack scenarios among the FAT methods. In terms of the training cost, similar results are observed on CIFAR-10.

Table 6: Ablation study on CIFAR-10. Number in bold indicates the best.

L-Regular (Ours)	Cutout	A-WA (Ours)	Clean	PGD-50	C&W	AA	Time
			86.82	44.23	45.89	42.37	1 ×
✓			82.97	54.29	50.41	48.05	1.4 ×
✓	✓		82.55	55.08	50.69	48.63	1.4 ×
✓		✓	80.38	56.33	51.48	49.27	1.4 ×
✓	✓	✓	80.46	**56.72**	**51.76**	**49.36**	1.4 ×

5.3 Ablation Study

In the proposed method, we propose a Lipschitz regularization (L-regular) term and an auto weight averaging (A-WA) with Cutout augmentation to conduct FAT. To validate the effectiveness of each element, we conduct an ablation study on CIFAR-10 by using ResNet18. The result is shown in Table 6. We use the clean accuracy on clean images and the robust accuracy on different adversarial attack methods as the evaluation metric on the last checkpoint. Analyses are summarized as follows. First, only using the proposed regularization, the performance of the model robustness under all attack scenarios significantly improves. Incorporating our regularization with cutout augmentation, the performance of the model robustness further improves. Incorporating our regularization with the proposed weight averaging, the proposed method can better model robustness. Second, using all terms can obtain the best robust performance under all attack scenarios, which indicates that the three elements are compatible, and combining them can achieve the best performance.

6 Conclusion

This work conducts a thorough analysis of existing works in the field of fast adversarial training in terms of adversarial robustness and training cost. Thus, we revisit the effectiveness and efficiency of the fast adversarial training techniques in preventing catastrophic overfitting. We explore efficient and effective FAT-related techniques and propose a novel regularization method motivated by the Lipschitz constraint. Moreover, we explore the effect of data augmentation and weight averaging in FAT and propose an effective auto weight averaging method for FAT to improve robustness. By assembling these techniques, we conclude our fast adversarial training method equipped with the proposed Lipschitz regularization and the proposed auto weight averaging, called FGSM-LAW. Extensive experimental evaluations demonstrate that the proposed method outperforms state-of-the-art FAT methods with less training costs.

ETHICS STATEMENT

In this paper, we propose a novel fast adversarial training method FGSM-LAW to improve the model robustness against adversarial examples. Our goal is to find efficient and effective fast adversarial training techniques for further research. In detail, we revisit the effectiveness and efficiency of fast adversarial training techniques in preventing Catastrophic Overfitting from the perspective of model local nonlinearity. By assembling these techniques, we conclude an efficient and effective fast adversarial training method, *i.e,* FGSM-LAW. We did not use crowdsourcing and did not conduct research with human subjects in our experiments. We cited the creators when using existing assets (e.g., code, data, models).

REPRODUCIBILITY STATEMENT

We show proof of the effectiveness of the proposed method in Appendix C. Our FGSM-LAW is an efficient and effective fast adversarial training method. We conduct an ablation study to evaluate the effectiveness of the proposed FGSM-LAW in Sec. 5.3. We specify the settings of hyper-parameters and how they were chosen in Appendix F. We repeat experiments 5 times. We plan to open the source code to reproduce the main experimental results later.

REFERENCES

Maksym Andriushchenko and Nicolas Flammarion. Understanding and improving fast adversarial training. *Advances in Neural Information Processing Systems*, 33:16048–16059, 2020.

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.

Tianlong Chen, Zhenyu Zhang, Sijia Liu, Shiyu Chang, and Zhangyang Wang. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations*, 2020.

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.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15721–15730, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

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

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020.

Jindong Gu, Hengshuang Zhao, Volker Tresp, and Philip Torr. Adversarial examples on segmentation models can be easy to transfer. *arXiv preprint arXiv:2111.11368*, 2021.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Xiaojun Jia, Yong Zhang, Xingxing Wei, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. Prior-guided adversarial initialization for fast adversarial training. *arXiv preprint arXiv:2207.08859*, 2022a.

Xiaojun Jia, Yong Zhang, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. Las-at: Adversarial training with learnable attack strategy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13398–13408, 2022b.

Xiaojun Jia, Yong Zhang, Baoyuan Wu, Jue Wang, and Xiaochun Cao. Boosting fast adversarial training with learnable adversarial initialization. *IEEE Transactions on Image Processing*, 31: 4417–4430, 2022c.

Hoki Kim, Woojin Lee, and Jaewook Lee. Understanding catastrophic overfitting in single-step adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8119–8127, 2021.

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

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.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. *arXiv preprint arXiv:2010.00467*, 2020.

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021.

Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pp. 8093–8104. PMLR, 2020.

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *Advances in Neural Information Processing Systems*, 32, 2019.

Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 464–472. IEEE, 2017.

Gaurang Sriramanan, Sravanti Addepalli, Arya Baburaj, et al. Guided adversarial attack for evaluating and enhancing adversarial defenses. *Advances in Neural Information Processing Systems*, 33:20297–20308, 2020.

Gaurang Sriramanan, Sravanti Addepalli, Arya Baburaj, et al. Towards efficient and effective adversarial training. *Advances in Neural Information Processing Systems*, 34:11821–11833, 2021.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

Hongjun Wang and Yisen Wang. Self-ensemble adversarial training for improved robustness. *arXiv preprint arXiv:2203.09678*, 2022.

Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1924–1933, 2021.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2019.

Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel. Evaluating the robustness of neural networks: An extreme value theory approach. *arXiv preprint arXiv:1801.10578*, 2018.

Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

Boxi Wu, Jinghui Chen, Deng Cai, Xiaofei He, and Quanquan Gu. Do wider neural networks really help adversarial robustness? *Advances in Neural Information Processing Systems*, 34:7054–7067, 2021.

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

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 *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

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.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *International conference on machine learning*, pp. 11278–11287. PMLR, 2020.

Haizhong Zheng, Ziqi Zhang, Juncheng Gu, Honglak Lee, and Atul Prakash. Efficient adversarial training with transferable adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1181–1190, 2020.

This appendix contains the following contents:

- Summary of the existing FAT in Sec. 1 of the manuscript. (see Appendix A).
- Detail description of the sample initialization used in the existing FAT in Sec. 3.2 of the manuscript. (see Appendix B).
- More results of FGSM-BR with different data augmentation methods in Sec. 4.1 of the manuscript (see Appendix D).
- Proof to the effectiveness of the proposed regularization in Sec. 3.4 of the manuscript. (see Appendix C).
- Results of the comparative experiments with a cyclic learning rate strategy in Sec. 5.1 of the manuscript. (see Appendix E).
- Detailed hyper-parameter settings in Sec. 5.1 of the manuscript. (see Appendix F).
- Experiments with WideResNet34-10 as the backbone in Sec. 5.2 of the manuscript. (see Appendix G).
- Experiments on Tiny ImageNet and ImageNet in Sec. 5 of the manuscript. (see Appendix H).

A SUMMARY OF THE EXISTING FAST ADVERSARIAL TRAINING METHODS

We investigate the used techniques of more than tens of papers working on the FAT methods. The summary is shown in Table 7. It can be observed that different FAT methods require different training costs, *i.e,* training time and memory costs.

B DETAIL DESCRIPTION OF THE SAMPLE INITIALIZATION USED IN THE EXISTING FAT

The primary FGSM-AT proposed by Goodfellow et al. (2014) conduct FGSM with a zero initialization to generate adversarial examples, *i.e.,* $\boldsymbol{\eta} = 0$. Tramèr et al. (2017) apply FGSM with a normal initialization in the half perturbation to generate adversarial examples for training, called **FGSM-NR**, It can be defined as:

$$\boldsymbol{\delta} = \Pi_{[-\epsilon,\epsilon]}\left[\frac{\epsilon}{2}\cdot \mathrm{Normal}(0,1) + \alpha\cdot \mathrm{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\frac{\epsilon}{2}\cdot\mathrm{Normal}(0,1);\boldsymbol{\theta}),\mathbf{y})\right)\right], \tag{9}$$

where α is set to $\epsilon/2$ for training. After that, Andriushchenko & Flammarion (2020) propose to implement FGSM with a uniform initialization in the whole perturbation to generate adversarial examples for training, called **FGSM-RS**. It can be defined as:

$$\boldsymbol{\delta} = \Pi_{[-\epsilon,\epsilon]}\left[\epsilon\cdot\mathrm{Uniform}(-1,1) + \alpha\cdot\mathrm{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\epsilon\cdot\mathrm{Uniform}(-1,1);\boldsymbol{\theta}),\mathbf{y})\right)\right], \tag{10}$$

where α is set to 1.25ϵ for training. Moreover, in the recent research (Sriramanan et al. (2020; 2021)), they adopt a Bernoulli initialization in the half perturbation to perform fast adversarial training, called **FGSM-BR**. It can be defined as:

$$\boldsymbol{\delta} = \Pi_{[-\epsilon,\epsilon]}\left[\frac{\epsilon}{2}\cdot \mathrm{Bernoulli}(-1,1) + \alpha\cdot \mathrm{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\frac{\epsilon}{2}\cdot\mathrm{Bernoulli}(-1,1);\boldsymbol{\theta}),\mathbf{y})\right)\right], \tag{11}$$

where α is set to ϵ for training. Also, Zheng et al. (2020) propose to adopt the previous adversarial perturbation from the last epoch to initialize the clean sample, called **ATTA**. It can be defined as:

$$\boldsymbol{\delta}_{E_{t+1}} = \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\delta}_{E_t,\mathbf{x}} + \alpha\cdot\mathrm{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}_{E_t,\mathbf{x}};\boldsymbol{\theta}),\mathbf{y})\right)\right], \tag{12}$$

where α is set to ϵ for training, $\boldsymbol{\delta}_{E_t,\mathbf{x}}$ is the generated adversarial perturbation by FGSM attack at the t-th epoch on the input sample $\mathbf{x}$. Besides, Jia et al. (2022a) use series of prior-guided adversarial perturbations as the adversarial initialization which are prior from the previous batch, the previous epoch and the momentum of all previous epochs to perform FGSM for training, called **FGSM-PGI**. It can be defined as:

$$\mathbf{g}_c = \mathrm{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\eta}_{E_{t-1}};\boldsymbol{\theta}),\mathbf{y})\right), \tag{13}$$

$$\mathbf{g}_{E_t} = \mu\cdot\mathbf{g}_{E_{t-1}} + \mathbf{g}_c, \tag{14}$$

$$\boldsymbol{\delta}_{E_t} = \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\eta}_{E_{t-1}} + \alpha\cdot\mathbf{g}_c\right], \tag{15}$$

$$\boldsymbol{\eta}_{E_t} = \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\eta}_{E_{t-1}} + \alpha\cdot\mathrm{sign}(\mathbf{g}_{E_t})\right]. \tag{16}$$

Table 7: The techniques used in different fast adversarial training methods on CIFAR-10.

Method	Initialization	Step Size	Regularization	Training Time	Data Memory	Early Stop	CO w/o Early Stop	>45% under AA
FGSM-AT	$\eta = 0$	ϵ	✘	1 ×	One Batch	✔	✔	✘
Free-AT	$\eta = 0$	ϵ	✘	4.2 ×	One Batch	✘	✘	✘
FGSM-RS	$\eta \in \mathbf{U}(-\epsilon, \epsilon)$	1.25ϵ	✘	1 ×	One Batch	✔	✔	✘
FGSM-GA	$\eta \in \mathbf{U}(-\epsilon, \epsilon)$	1.25ϵ	✔	3.5 ×	One Batch	✔	✘	✘
FGSM-CKPT	$\eta \in \mathbf{U}(-\epsilon, \epsilon)$	1.25ϵ	✘	1.5 ×	One Batch	✘	✘	✘
ATTA	$\eta_{t+1} = \boldsymbol{\delta}_E$	ϵ	✘	1 ×	Whole Dataset	✘	✘	✘
NuAT	$\eta \in \mathbf{B}(-\epsilon/2, \epsilon/2)$	ϵ	✔	2 ×	One Batch	✘	✘	✔
GAT	$\eta \in \mathbf{B}(-\epsilon/2, \epsilon/2)$	ϵ	✔	2 ×	One Batch	✘	✘	✔
SLAT	$\eta \in \mathbf{U}(-\epsilon, \epsilon)$	ϵ	✔	1.1 ×	One Batch	✘	✘	✘
FGSM-SDI	$\eta = \epsilon \cdot g\left(, \theta\right)$	ϵ	✘	1.6 ×	One Batch	✘	✘	✔
FGSM-PGI	$\eta_{t+1} = \boldsymbol{\delta}_P$	ϵ	✔	1.4 ×	Whole Dataset	✘	✘	✔

Table 8: Test robustness (%) on the CIFAR-10 database using ResNet18 with different data augmentation techniques. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time
Baseline	86.82	46.62	44.76	44.23	45.89	42.37	1 ×
FGSM-BR+Mixup	**86.94**	46.78	44.93	44.41	45.52	42.34	1 ×
FGSM-BR+Cutout	84.93	**48.31**	**46.9**	**46.43**	**46.91**	**43.79**	1 ×
FGSM-BR+AutoAugment	85.79	48.97	47.24	46.57	46.06	42.25	1 ×
FGSM-BR+CutMix	81.12	50.45	49.28	48.95	45.36	42.87	1 ×

where α is set to ϵ for training, $\mathbf{g}_c$ is regarded as the signed gradient and $\mathbf{g}_{E_t}$ is the signed gradient momentum in the t-th epoch. Moreover, Jia et al. (2022c) propose to adopt a generative model to generate the sample initialization for training, called **FGSM-SDI**. It can be defined as:

$$\boldsymbol{\delta} = \Pi_{[-\epsilon,\epsilon]}\left[\epsilon \cdot g\left(\mathbf{x}, \mathbf{x}_g; \mathbf{w}\right) + \alpha \cdot \operatorname{sign}\left(\nabla_{\mathbf{x}} \mathcal{L}(f(\mathbf{x} + \epsilon \cdot g\left(\mathbf{x}, \mathbf{x}_g; \mathbf{w}\right); \boldsymbol{\theta}), \mathbf{y})\right)\right], \tag{17}$$

where α is set to ϵ for training, $\mathbf{x}_g$ represents gradient information of $\mathbf{x}$ on the trained model and $g\left(\cdot, \cdot; \mathbf{w}\right)$ represents the generative model with the weight parameters $\mathbf{w}$.

C Proof to the effectiveness of the proposed regularization

Actually the proposed regularization equation 7 corresponds to a new sample-based local linearity metric of loss function ℓ corresponding to $(\mathbf{x}, \mathbf{y})$ and $\boldsymbol{\delta}$:

$$\mathbb{E}_{\boldsymbol{\eta}}\left[\frac{|f(\mathbf{x} + \boldsymbol{\delta}; \boldsymbol{\theta}) - f(\mathbf{x} + \boldsymbol{\eta}; \boldsymbol{\theta})|}{\|\boldsymbol{\delta} - \boldsymbol{\eta}\|_2}\right]. \tag{18}$$

Table 9: Test robustness (%) on the CIFAR-10 database using ResNet18 with different adversarial training methods by using a cyclic learning rate strategy. Number in bold indicates the best.

Method	Clean Best/Last	PGD-10 Best/Last	PGD-20 Best/Last	PGD-50 Best/Last	AA Best/Last	Time
PGD-AT	80.12/80.12	51.59/51.59	50.83/50.83	50.7/50.7	46.83/46.83	5.2 ×
FGSM-RS	83.75/83.75	48.05/48.05	46.47/46.47	46.11/46.11	42.92/42.92	1 ×
FGSM-PGI	80.68/80.68	52.48/53.48	51.69/51.69	51.5/51.5	46.65/46.65	1.4 ×
FGSM-CKPT	**89.08/89.08**	40.47/40.47	38.2/38.2	37.69/37.69	35.66/35.66	1.5 ×
FGSM-SDI	82.08/82.08	51.63/51.63	50.65/50.65	50.33/50.33	46.21/46.21	1.6 ×
NuAT	76.23/76.23	51.52/51.52	50.81/50.81	50.64/50.64	46.33/46.33	2.0 ×
GAT	81.91/81.91	50.43/50.43	49.82/49.82	49.62/49.62	45.24/45.24	2.0 ×
FGSM-GA	80.83/80.83	48.76/48.76	47.83/47.83	47.54/47.54	43.06/43.06	3.5 ×
Free-AT (m=8)	75.22/75.22	44.67/44.67	43.97/43.97	43.72/43.72	40.30/40.30	4.2 ×
FGSM-LAW (ours)	80.44/80.44	**53.54/53.54**	**52.91/52.91**	**52.73/52.73**	**46.93/46.93**	1.4 ×

Table 10: Test robustness (%) on the CIFAR-10 database using WideResNet34-10 with different adversarial training methods. Number in bold indicates the best.

CIFAR-10	Clean	PGD-10	PGD-20	PGD-50	AA	Time
PGD-AT	85.17	56.1	55.07	54.87	51.67	5.2 ×
FGSM-RS	74.3	42.3	41.2	40.9	38.4	1 ×
FGSM-PGI	85.09	57.72	56.86	56.4	50.11	1.4 ×
FGSM-CKPT	**91.84**	44.7	42.72	42.25	40.46	1.5 ×
FGSM-SDI	86.4	55.89	54.95	54.6	51.17	1.6 ×
NuAT	85.30	55.8	54.68	53.75	50.06	2.0 ×
GAT	85.17	56.3	55.23	54.97	50.01	2.0 ×
FGSM-GA	82.1	48.9	47.1	46.9	45.7	3.5 ×
Free-AT	80.1	47.9	46.7	46.3	43.9	4.2 ×
Ours	84.78	**59.88**	**59.26**	**59.12**	**52.29**	1.4 ×

This metric is bounded with the gradient of f at $(\mathbf{x}+\boldsymbol{\delta}, \mathbf{y})$ as

$$\begin{aligned} & \mathbb{E}_{\boldsymbol{\eta}}\left[\frac{|f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta}) - f(\mathbf{x}+\boldsymbol{\eta};\boldsymbol{\theta})|}{\|\boldsymbol{\delta}-\boldsymbol{\eta}\|_2}\right] \\ \leq\ & \mathbb{E}_{\boldsymbol{\eta}}\left[\frac{|\nabla_{\mathbf{x}}^{\top} f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta})(\boldsymbol{\delta}-\boldsymbol{\eta})|}{\|\boldsymbol{\delta}-\boldsymbol{\eta}\|_2}\right] \\ \leq\ & \mathbb{E}_{\boldsymbol{\eta}}\left[\frac{\|\nabla_{\mathbf{x}} f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta})\|_2 \|\boldsymbol{\delta}-\boldsymbol{\eta}\|_2}{\|\boldsymbol{\delta}-\boldsymbol{\eta}\|_2}\right] \\ \leq\ & \|\nabla_{\mathbf{x}} f(\mathbf{x}+\boldsymbol{\delta};\boldsymbol{\theta})\|_2 \leq \max_{\mathbf{x}} \|\nabla_{\mathbf{x}} f(\mathbf{x};\boldsymbol{\theta})\|_2. \end{aligned} \tag{19}$$

The first inequality holds as the score function $f(\cdot,\cdot)$ is a convex function *w.r.t* $\mathbf{x}$ and the second one holds by the Hölder inequality. It means that the proposed metric is bounded by the gradient of the adversarial perturbation $(\mathbf{x}+\boldsymbol{\delta}, \mathbf{y})$. Then equation 7 could generate the parsimonious effect on the local Lipschitzness for any $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$. As a consequence, the proposed regularization could produce a more robust model than the existing adversarial training methods.

Table 11: Test robustness (%) on the CIFAR-10 database using ResNet18 with equipped with Lipschitz regularization with different hyper-parameter λ . Number in bold indicates the best.

Hyper-Parameter	Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time
Baseline	**86.82**	46.62	44.76	44.23	45.89	42.37	1 ×
$\lambda = 8$	84.34	54.09	53.21	52.88	49.28	46.57	1.4 ×
$\lambda = 10$	84.04	84.04	54.31	53.03	49.83	47.08	1.4 ×
$\lambda = 12$	83.35	53.94	54.03	53.66	50.05	47.38	1.4 ×
$\lambda = 14$	82.97	**55.49**	**54.58**	**54.29**	**50.41**	**48.05**	1.4 ×
$\lambda = 16$	83.49	54.86	53.99	53.62	49.53	47.11	1.4 ×
$\lambda = 18$	82.74	55.2	54.35	54.27	50.17	47.58	1.4 ×

Table 12: Test robustness (%) on the CIFAR-10 database using ResNet18 equipped with auto weight averaging with diffrerent hyper-parameters Δ. Number in bold indicates the best.

Hyper-Parameter	Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time
Baseline	**86.82**	46.62	44.76	44.23	45.89	42.37	1 ×
$\Delta = 0.6$	79.98	51.47	**50.7**	**50.53**	48.53	46.08	1 ×
$\Delta = 0.65$	80.92	**51.69**	50.9	49.73	49.15	46.18	1 ×
$\Delta = 0.7$	81.82	50.94	50.11	49.85	49.21	46.22	1 ×
$\Delta = 0.75$	83.62	51.26	50.21	49.9	**49.42**	**46.53**	1 ×
$\Delta = 0.8$	85.76	48.9	47.35	46.89	47.59	44.33	1 ×
$\Delta = 0.85$	86.7	47.83	46.04	45.68	46.84	43.47	1 ×

D More Results of Different Data Augmentation Methods

We adopt the PGD-10, PGD-20, PGD-50, C&W and AA to evaluate the FGSM-BR with different data augmentation methods. The result is shown in Table 8. It ii clear that using Cutout, Mixup, CutMix and AutoAugment can improve model robustness. FGSM-BR combined with Cutout achieves the best robustness performance in all attack scenarios. Specifically, under the strong AA attack, Cutout improves the performance by about 1.5%. Moreover, Cutout also achieves comparable clean accuracy on the clean images to the vanilla FAT (FGSM-BR).

E Experiments with a cyclic learning rate strategy

All experiments are implemented by using a multi-step learning rate strategy in the manuscript. The comparative experiments are also implemented by using a cyclic learning rate strategy on CIFAR-10. Following the training settings (Wong et al. (2020); Jia et al. (2022a), the maximum learning rate of FGSM-GA Andriushchenko & Flammarion (2020) and FGSM-CKPT Kim et al. (2021) is set to 0.3. The maximum learning rate of other FAT methods is set to 0.2. The results are shown in Table 9. It can be observed that compared with PGD-AT, the proposed FGSM-LAW achieves better model robustness performance under all adversarial attack scenarios and require less training cost. And compared with previous FAT methods, the proposed FGSM-LAW can achieve the best adversarial robustness under all adversarial attack scenarios on the best and last checkpoint. In terms of the training cost, we observe similar phenomenons as the models trained with the multi-step learning rate strategy.

F Detailed hyper-parameter settings

There are two hyper-parameters in the proposed FGSM-LAW, *i.e,* the hyper-parameter of the Lipschitz regularization λ and the hyper-parameter of the auto weight averaging Δ. Using ResNet18

Table 13: Test robustness (%) on the Tiny ImageNet database using PreActResNet18 with different adversarial training methods. Number in bold indicates the best.

Method	Clean Best/Last	PGD-10 Best/Last	PGD-20 Best/Last	PGD-50 Best/Last	C&W Best/Last	AA Best/Last	Time
PGD-AT	43.6/45.28	20.2/16.12	19.9/15.6	19.86/15.4	17.5/14.28	16.0/12.84	5.2×
FGSM-RS	44.98/45.18	17.72/0.00	17.46/0.00	17.36/0.00	15.84/0.00	14.08/0.00	1 ×
FGSM-PGI	43.32/45.88	23.8/22.02	23.4/21.7	23.38/21.6	19.28/17.44	**17.56**/15.50	1.4 ×
FGSM-CKPT	**49.98/49.98**	9.20/9.20	9.20/9.20	8.68/8.68	9.24/9.24	8.10/8.10	1.5 ×
FGSM-SDI	46.46/47.64	23.22/19.84	22.84/19.36	22.76/19.16	18.54/16.02	17.0/14.10	1.6 ×
NuAT	42.9/42.42	15.12/13.78	14.6/13.34	14.44/13.2	12.02/11.32	10.28/9.56	2.0 ×
GAT	42.16/41.84	15.02/14.44	14.5/13.98	14.44/13.8	11.78/11.48	10.26/9.74	2.0 ×
FGSM-GA	43.44/43.44	18.86/18.86	18.44/18.44	18.36/18.36	16.2/16.2	14.28/14.28	3.5 ×
Free-AT (m=8)	38.9/40.06	11.62/8.84	11.24/8.32	11.02/8.2	11.00/8.08	9.28/7.34	4.2 ×
FGSM-LAW (ours)	47.74/47.74	**24.45/24.45**	**24.09/24.09**	**24.06/24.06**	**18.65/18.65**	16.43/**16.43**	1.4 ×

Table 14: Test robustness (%) on the ImageNet database using ResNet50 with different adversarial training methods. Number in bold indicates the best.

Method	Clean	AA	Time
PGD-AT Madry et al. (2017)	64.02	34.96	5.2 ×
Free-AT Shafahi et al. (2019)	59.96	28.58	2.8 ×
FGSM-RS Wong et al. (2020)	55.62	26.24	1 ×
FGSM-LAW (Ours)	**66.14**	**30.12**	1.4 ×

on CIFAR-10, we adopt the FGSM-BR combined with Lipschitz regularization and auto weight averaging to determine the optimal hyper-parameters, respectively. As for the hyper-parameter λ, the results are shown in Table 11. It can be observed that when $\lambda = 14$, the proposed Lipschitz regularization can achieve the best model robustness under all attack scenarios. In detail, under AA attack, the proposed Lipschitz regularization achieves an accuracy of about 48%. Hence, the hyper-parameter λ of the proposed Lipschitz regularization is set to 14. As for the hyper-parameter Δ, the results are shown in Table 12. It can be observed that when $\Delta = 0.75$, the proposed auto weight averaging can achieve the best model robustness under C&W and AA attack scenarios. Considering the clean accuracy on clean samples, the hyper-parameter Δ of the proposed auto weight averaging is set to 0.75.

G Experiments with WideResNet34-10 as the backbone

We adopt a larger architecture (WideResNet34-10) as the backbone to conduct comparative experiments on CIFAR-10. The results are shown in Table 10. It can be observed that compared with PGD-AT, the proposed FGSM-LAW achieves better model robustness. Particularly, under AA attack, PGD-AT achieves an accuracy of about 51%, while the proposed FGSM-LAW achieves an accuracy of about 52%. Compared with previous FAT methods, the proposed method achieves the best robustness performance under all attack scenarios. In terms of the training cost, we observe similar phenomenons as ResNet18 used as the backbone.

H Experiments on Tiny ImageNet and ImageNet

Results on Tiny ImageNet Compared with CIFAR-10 and CIFAR-100, Tiny ImageNet is a larger database that covers more classes. It is hard to obtain robustness on Tiny ImageNet As for Tiny ImageNet, following Jia et al. (2022c;a), we adopt PreActResNet18 as the backbone to conduct experiments. The results are shown in Table 13. It can be observed that compared with PGD-AT,

the proposed FGSM-LAW achieves better model robustness and requires less training time. And compared with previous FAT methods, the proposed method achieves the best model robustness under most attack scenarios and comparable robustness to the advanced FAT method (FGSM-PGI) under AA attack. However, compared with FGSM-PDI, the proposed FGSM-LAW achieves better model robustness under all attack scenarios without extra data memory. In terms of the training cost, we can observe similar phenomenons as the model trained on CIFAR-10 and CIFAR-100.

Results on ImageNet Following the training settings (Shafahi et al., 2019; Wong et al., 2020), we adopt ResNet50 as the backbone to conduct comparative experiments. The maximum perturbation strength ϵ is set to 8/255. We compare the proposed FGSM-LAW with PGD-AT (Madry et al., 2017), Free-AT (Shafahi et al., 2019) and FGSM-RS (Wong et al., 2020). The results are shown in Table 14. Compared with PGD-AT, our FGSM-LAW achieves higher clean accuracy and comparable robustness under AA attack. Compared with Free-AT and FGSM-RS, the proposed FGSM-LAW achieves the best clean and robust accuracy.