PIAT: Parameter Interpolation based Adversarial Training for Image Classification

Kun He [*1] **Xin Liu** [*1] **Yichen Yang** [*1] **Zhou Qin** [2] **Weigao Wen** [2] **Hui Xue** [2] **John E. Hopcroft** [3]

Abstract

Adversarial training has been demonstrated to be the most effective approach to defend against adversarial attacks. However, existing adversarial training methods show apparent oscillations and overfitting issues in the training process, degrading the defense efficacy. In this work, we propose a novel framework, termed Parameter Interpolation based Adversarial Training (PIAT), that makes full use of the historical information during training. Specifically, at the end of each epoch, PIAT tunes the model parameters as the interpolation of the parameters of the previous and current epochs. Besides, we suggest to use the Normalized Mean Square Error (NMSE) to further improve the robustness by aligning the relative magnitude of logits between clean and adversarial examples, rather than the absolute magnitude. Extensive experiments on several benchmark datasets and various networks show that our framework could prominently improve the model robustness and reduce the generalization error.

1. Introduction

Deep Neural Networks (DNNs) have been widely used in various tasks of computer vision (He et al., 2016) and natural language processing (Devlin et al., 2019). However, even the model performance surpasses humans in some tasks, they are known to be vulnerable to adversarial examples by injecting malicious and imperceptible perturbations to clean inputs that can cause the model to misclassify inputs with high confidence (Szegedy et al., 2014; Goodfellow et al., 2015; Madry et al., 2018; Croce & Hein, 2020; Athalye et al., 2018; Hendrycks et al., 2019; Xie et al., 2017; Meng & Chen, 2017). Since DNNs are applied in many safety systems, it is crucial to make them more reliable and robust.

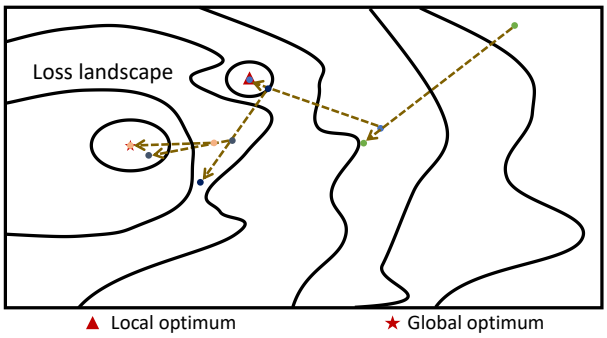


Figure 1. Illustration of the parameter update for PIAT framework. The start and end points of each epoch are in colored dots.

To improve the model's robustness against adversarial attacks, Adversarial Training (AT) is known to be the most effective approach to defend against adversarial attacks, which generates adversarial examples during the training and incorporates them into the training. However, the training process of AT has apparent oscillations in the early stage and overfitting issues in the later stage. In the early stage, the model parameters are updated rapidly with a large learning rate, thus the adversarial examples generated at each epoch are pretty different, leading to the oscillations on robust accuracy. In the later stage, many works (Yu et al., 2022; Rice et al., 2020; Dong et al., 2022) and our experiments have shown that the overfitting issue occurs, that is, the training accuracy continues to increase but the robust accuracy on the testing data begins to decline.

Therefore, we propose a Parameter Interpolation based Adversarial Training (PIAT) framework to solve the oscillation and overfitting issues. As illustrated in Figure 1, at the end of each epoch of the training, we tune the model parameters as the interpolation of the model parameters of the previous and current epochs. In the early stage of AT, the adversarial examples generated at each epoch is significantly different, and the model's decision boundary changes dramatically. In contrast, PIAT tunes the model parameters with the previous epoch and makes the change of the decision boundary more

[*]Equal contribution [1]School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China [2]Alibaba Group, China [3]Department of Computer Science, Cornell University, Ithaca, NY, USA. Correspondence to: Kun He <brooklet60@hust.edu.cn>.

2^{nd} AdvML Frontiers workshop at 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

moderate, helping to eliminate the oscillation. In the later stage, PIAT considers the previous boundary, preventing the decision boundary from becoming too complex and alleviating the overfitting issue. As the training continues and the model parameters become more valuable, PIAT gradually increases the weight of the previous parameters when tuning the current parameters.

There have been many works proposed to encourage similarity between the output of clean and adversarial examples. We observe that ALP (Kannan et al., 2018b) uses the Square Error (SE) loss to align absolute magnitude of logits between clean and adversarial examples. However, the data distribution of clean and adversarial examples is quite different, and simply forcing the output to be close is too demanding. We suggest that AT should pay more attention to aligning the relative magnitude, rather than the absolute magnitude of logits. Therefore, we propose a new metric called Normalized Mean Square Error (NMSE) to better align the clean and adversarial examples.

We incorporate the Normalized Mean Square Error (NMSE) as the regularization term into the proposed PIAT framework. Extensive experiments on CIFAR10, CIFAR100, and SVHN datasets show that our method performs better and effectively improves the adversarial robustness of the model against white-box and black-box attacks. In addition, our PIAT framework is general and other adversarial training methods can be incorporated into our framework to achieve better performance.

Our main contributions are summarized as follows: (1) We propose the PIAT framework that interpolates the model parameters of the previous and current epochs to consider the historical information during training. (2) We propose to use the NMSE loss as a new regularization term to better align the clean and adversarial examples by the relative magnitude of logits. (3) Extensive experiments on three standard datasets and two networks show that PIAT combined with NMSE offers excellent robustness without incurring additional cost. Besides, the framework is general and various AT methods can be integrated with it to further boost their robustness.

2. Related Work

Adversarial Training (AT) (Madry et al., 2018) has been demonstrated to be one of the most effective defensive methods against adversarial attacks, which generates a locally most adversarial perturbed point for each clean example and trains the model to classify them correctly.

Given an image classification task, the training dataset $\mathbf{D} = f(\mathbf{x}_i; y_i)g_{i=1}^n$ consists of n clean examples with c classes, where $\mathbf{x}_i \in \mathbb{R}^d$ represents a clean example with the ground-truth label $y_i \in f1; 2; :::; cg$. The adversarial training optimization problem can be formulated as the following min-max problem:

$$\min_{} \sum_{i} \max_{\mathbf{x}_i' \in \mathbf{S}(\mathbf{x}_i;\ )} L(f\ (\mathbf{x}_i'); y_i); \tag{1}$$

where $f\ (\ ) : \mathbb{R}^d\ \ !\ \ \mathbb{R}^c$ is the DNN classifier with parameter $\ .\ L(\ ;\ )$ represents the cross entropy loss and $\mathbf{S}(\mathbf{x}_i;\ ) = fjj\mathbf{x}_i' - \mathbf{x}_i jj_p \quad g$ represents an -ball of a benign data point $\mathbf{x}_i$. $\mathbf{x}_i'$ denotes the adversarial example generated from $\mathbf{x}_i$.

For the inner maximization problem, the adversarial example $\mathbf{x}_i'$ is often crafted by the Projected Gradient Decent (PGD) attack (Madry et al., 2018), formulated by:

$$\mathbf{x}_i^{t+1} = \prod_{\mathbf{S}(\mathbf{x}_i;\ )} (\mathbf{x}_i^t + \quad \text{sign}(r_{\mathbf{x}_i^t} L(f\ (\mathbf{x}_i^t); y_i)); \tag{2}$$

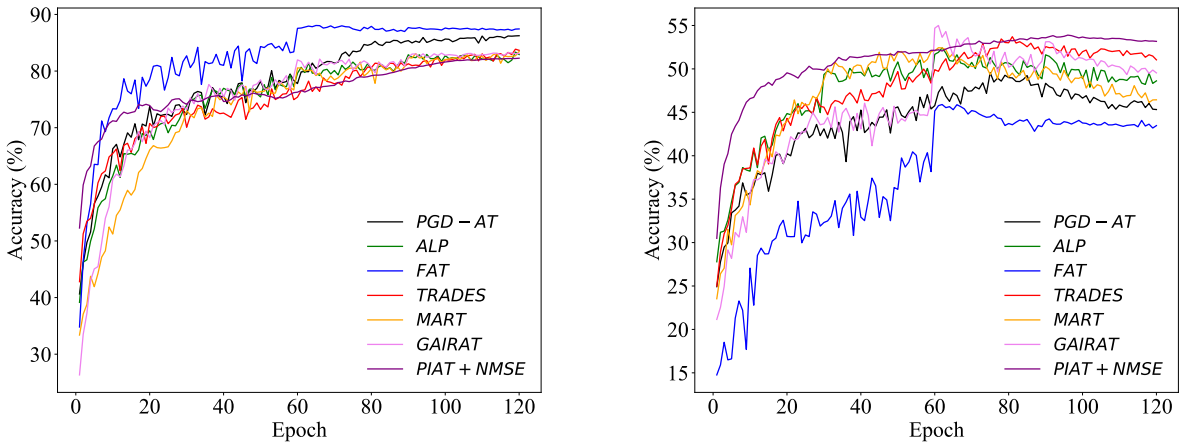
where $\mathbf{x}_i^t$ denotes the adversarial example at the t^{th} step, $\prod(\ )$ is the projection operator, and is the step size.

Subsequent efforts have been devoted to achieve better performance of AT. TRADES (Zhang et al., 2019) optimizes to classify the clean examples and align the logits between clean examples and corresponding adversarial examples to achieve a better tradeoff between accuracy and robustness. MART (Wang et al., 2019) explicitly differentiates the misclassified and correctly classified examples during the training. STAT (Li et al., 2023) takes both adversarial examples and collaborative examples into account for regularizing the loss landscape. FreeAT (Shafahi et al., 2019) considers the adversarial examples of each epoch of PGD. FAT (Zhang et al., 2020a) searches for the least adversarial data for training, rather than employing the most adversarial data that maximizes the loss. LAS-AT (Jia et al., 2022) learns to automatically produce attack strategies to generate adversarial examples for training. SCORE (Pang et al., 2022) facilitates the reconciliation between robustness and accuracy, while still handles worst-case uncertainty via robust optimization.

The works most related to ours are SEAT (Wang & Wang, 2022) and ALP (Kannan et al., 2018b). SEAT trains the model with standard AT and gets the ensemble model by averaging weights of the historical model at each minibatch. In contrast, our PIAT framework interpolates the last and current model parameters to achieve a more moderate change in the decision boundary at each epoch and continues to train the model using the interpolated parameters. ALP calculates the regularization loss using the absolute magnitude of logits with the SE loss. In contrast, our NMSE regularization focuses on aligning the relative magnitude of logits, rather than the absolute magnitude.

3. Motivation

This section further analyzes the oscillations in the early stage and the overfitting issues in the later stage of the



(a) Test accuracy on clean examples

(b) Test accuracy on adversarial examples

Figure 2. Test accuracy on clean and adversarial examples of ResNet18 trained by various AT methods on CIFAR10.

AT process. Based on our observations, we propose to fully utilize the historical training information by parameter interpolation at the end of each epoch.

Figure 2 illustrates the test accuracy on clean and adversarial examples of several popular AT methods during training, including PGD-AT (Madry et al., 2018), ALP (Kannan et al., 2018a), FAT (Zhang et al., 2020a), TRADES (Zhang et al., 2019), MART (Wang et al., 2019) and GAIRAT (Zhang et al., 2020b). As illustrated in Figure 2 (b), over all these AT methods except our method (PIAT + NMSE), we can observe that in the early stage, there are apparent oscillations in terms of robust accuracy, and in the later stage, the adversarial robustness declines as the model overfits the adversarial examples.

In the early stage of AT, a large learning rate leads to rapid changes on the decision boundary, causing the adversarial examples generated by the PGD attack to vary significantly at each epoch. This makes it hard for the model to learn features and have good generalization, resulting in unstable robust accuracy. To address this issue, the change on the decision boundary needs to be more moderate, which can be achieved by reducing the learning rate. However, a low learning rate will slow down the convergence of AT and lead to overfitting.

In the later stage, as the learning rate of most AT methods is reduced significantly, the decision boundary will become overly complex when the model struggles to well fit the adversarial examples crafted during training. As a result, the learned model is not robust enough to well handle unseen adversarial examples, leading to the overfitting issue.

We observe that existing AT methods ignore the historical information in the training process, which would be useful to stabilize the training and alleviate overfitting. To this end, we propose to utilize the model parameters of previous epoch to tune the current parameters at the end of each epoch. Such an approach allows us to leverage historical information and improve the the robust generalization capability. In the early stage, considering parameters of the previous epoch will result in a more moderate update on the model parameters, helping to ensure a smoother and more stable training. In the later stage, mixing parameters of the previous epoch could smooth the decision boundary and prevent the model from overfitting.

4. Methodology

In this section, we introduce the realization of the PIAT framework, and also describe how to combine with our proposed Normalized Mean Square Error (NMSE).

4.1. The PIAT Framework

To fully utilize the historical information during training, at the end of each epoch, PIAT tunes the model parameters as the interpolation of parameters of the previous and current epochs, which can be formalized as:

$$\theta_t^0 = \theta_{t-1}^0 + (1 - \lambda)\theta_t; \quad 0 \leq \lambda \leq 1; \tag{3}$$

where θ_{t-1}^0 is the model parameters of the previous epoch after interpolation, and θ_t is current parameters before interpolation at the end of the training epoch. Before starting the next epoch, we tune the parameters to θ_t^0. The hyper-parameter λ controls the tradeoff between previous and current parameters.

Algorithm 1 The PIAT Framework

Input: Initial model parameters $_0$, perturbation size , number of adversarial attack steps K, number of epochs N, weight function $g(\ )$
Output: ^0_N
Initialize $^0_0$ $_0$
for $i = 1$ **to** N **do**
  $_i$ ^0_i $_1$
  for *minibatch* $\mathbf{x}$ $\mathbf{X}$ **do**
    $\mathbf{x}_{adv}$ $\mathbf{x}$
    for $k = 1$ **to** K **do**
      $\mathbf{x}_{adv}$ $\mathbf{x}_{adv}$ + $sign(\ _{\mathbf{x}} L_{CE}(\mathbf{x}_{adv}; y))$
      $\mathbf{x}_{adv}$ $clip(\mathbf{x}_{adv}; \mathbf{x}$ $; \mathbf{x}$ + $)$
    end for
    $loss = L(\mathbf{x}_{adv}; y)$
    update $_i$
  end for
  $g(i)$
  ^0_i $^0_{i\ 1}$ + (1) $_i$
end for
return ^0_N

The value of is critical to PIAT. In the early stage of AT, the model has not yet fit the training data well enough. Thus, the model is not robust enough against adversarial attacks, and its parameters are not very informative. If is set too large, the model mainly relies on previous parameters and learns the training data in small steps, leading to slow convergence. Therefore, should be small in the early stage.

As the training continues, the model starts to learn enough information from the adversarial examples and attains adversarial robustness. In the later stage, the model has learned enough information and gained good robustness. should be close to 1, otherwise the model tends to discard useful information learned over the training process and overfit the current adversarial examples.

According to the above analysis, should change over the course of training, instead of using a fixed value. The value of should be small in the early stage of training and gradually increase along with the training, which not only ensures the convergence speed but also alleviates the overfitting issue in AT. In this paper, we set as follows:

$$= g(n) = \frac{an + b}{cn + d}; \quad c \quad a; \quad d \quad b; \tag{4}$$

where n denotes the current number of training epochs. a, b, c and d are hyper-parameters and we set $a = b = c = 1$, $d = 10$ in this work. We verify that a dynamic has better robustness against a fixed value in Appendix A.

Algorithm 1 concludes the overall framework. Since PIAT does not restrict the type of loss function in the framework, it is flexible and can be combined with various adversarial training methods such as TRADES (Zhang et al., 2019), MART (Wang et al., 2019) and GAIRAT (Zhang et al., 2020b).

4.2. The NMSE Regularization

According to the discussion in Section 3, instead of aligning the clean and adversarial examples by classification probabilities, we utilize the output logits normalized with l_2-norm.

We align the clean and adversarial examples by minimizing the mean square error between their normalized output logits. Besides, we set $(1 \quad p_{clean})$ as the weight for different adversarial examples so that the model will pay more attention to the clean examples which are vulnerable. We formulate the Normalized Mean Square Error (NMSE) regularization as follows:

$$L_{NMSE} = (1 \quad p_{clean}) \quad \frac{f\ (\mathbf{x})}{jjf\ (\mathbf{x})jj_2} \quad \frac{f\ (\mathbf{x}^0)}{jjf\ (\mathbf{x}^0)jj_2}\ ^2_2; \tag{5}$$

where $\mathbf{x}^0$ is the adversarial example, $f\ (\mathbf{x})$ is the output logits of the model, and $jj \quad jj_2$ denotes l_2-norm.

In summary, the overall loss function in PIAT framework with NMSE is as follows:

$$L = L_{CE} + \quad L_{NMSE}; \tag{6}$$

where is a hyper-parameter to trade off the cross-entropy loss L_{CE} on adversarial examples and the NMSE regularization term L_{NMSE}. Moreover, we could replace the loss function in PIAT framework to combine with various AT methods.

5. Experiments

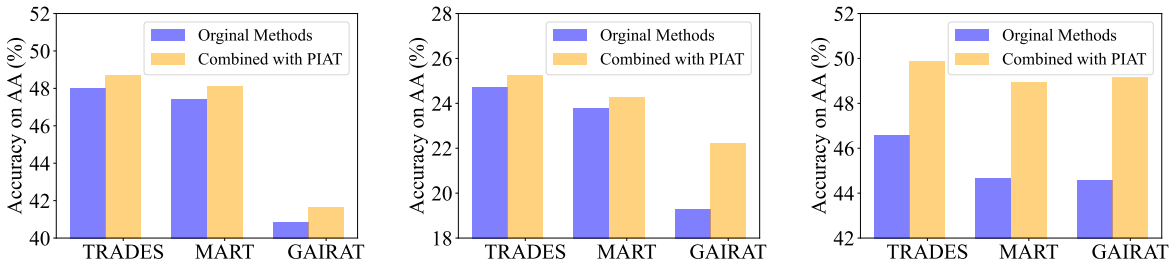
5.1. Experimental Setup

Datasets and Models We conduct experiments on three benchmark datasets including CIFAR10 (Krizhevsky et al., 2009), CIFAR100 (Krizhevsky et al., 2009), and SVHN (Netzer et al., 2011). All images are normalized into $[0; 1]$. We do evaluation on two models, ResNet18 (He et al., 2016) and WRN-32-10 (Zagoruyko & Komodakis, 2016), to verify the efficacy of our method.

Evaluation Details We compare the PIAT combined with NMSE regularization with the following AT baselines: TRADES (Zhang et al., 2019), MART (Wang et al., 2019), and GAIRAT (Zhang et al., 2020b). To thoroughly evaluate the defense efficacy of our method and the baselines, we adopt various adversarial attacks including PGD (Madry et al., 2018), MIM (Dong et al., 2018), CW (Carlini & Wagner, 2017), and AA (Croce & Hein, 2020). For crafting

Table 1. The accuracy (%) of our method and AT baselines under various adversarial attacks on CIFAR10, CIFAR100 and SVHN datasets with ResNet18 model.

Dataset	Method	Clean	PGD^{20}	PGD^{100}	MIM	CW	AA
CIFAR10	PGD-AT	**84.28**	50.29	50.12	51.21	49.31	46.33
	TRADES	82.39	53.60	53.65	54.55	50.90	48.04
	MART	81.91	53.70	53.70	54.95	49.35	47.45
	GAIRAT	81.69	**55.84**	**55.90**	**56.62**	45.50	40.85
	PIAT	79.08	51.81	51.74	52.63	49.32	47.08
	PIAT +NMSE	80.76	53.54	53.59	54.51	**51.72**	**48.80**
CIFAR100	PGD-AT	58.48	28.36	28.33	29.30	27.06	23.85
	TRADES	57.98	29.90	29.88	29.55	26.14	24.72
	MART	55.26	30.10	30.16	30.51	26.00	23.77
	GAIRAT	50.26	23.33	23.35	23.90	21.55	19.26
	PIAT	**58.84**	29.11	29.14	29.97	27.89	24.15
	PIAT +NMSE	54.34	**31.11**	**30.99**	**31.42**	**28.45**	**25.79**
SVHN	PGD-AT	**93.85**	59.01	58.92	59.93	48.66	43.02
	TRADES	90.88	59.50	59.43	60.52	52.76	46.59
	MART	88.73	59.45	59.42	59.83	60.19	44.65
	GAIRAT	90.50	54.14	54.09	55.88	50.71	44.57
	PIAT	92.15	59.53	59.61	61.32	55.54	50.93
	PIAT +NMSE	91.70	**61.21**	**61.43**	**61.97**	**55.88**	**51.29**



(a) CIFAR10 (b) CIFAR100 (c) SVHN

Figure 3. The accuracy of the PIAT framework combined with various AT methods under the AA attack on CIFAR10, CIFAR100, and SVHN datasets with ResNet18 model.

adversarial examples, the maximum perturbation of each pixel is $= \frac{8}{255}$ with the PGD step size $= \frac{2}{255}$ and step number of 10. We will provide more details in Appendix A.

5.2. Evaluation on Defense Efficacy

We compare the defense efficacy of our method with four AT baselines including PGD-AT, TRADES, MART and GAIRAT. Table 1 reports the accuracy of ResNet18 model trained with our method or the defense baselines under various adversarial attacks on three datasets.

As shown in Table 1, our method (PIAT + NMSE) exhibits the best performance except for the PGD and MIM attack on CIFAR10 dataset. Under the AA attack, our method achieves 48.80%, 25.79% and 51.29% accuracy on CIFAR10, CIFAR100 and SVHN datasets. Compared to the best results of defense baselines, we gain an improvement of 0.76%, 1.07% and 4.70% on the three datasets, respectively, indicating the great superiority of our method. We do further evaluation on the WRN-32-10 model. The results are reported in Table 2, and our method achieves the dominant robustness under the AA attack with a clear margin as well.

5.3. Ablation Study

PIAT Framework Since PIAT is a general framework, we incorporate other AT methods into PIAT to demonstrate its defense efficacy. Figure 3 illustrates the accuracy of PIAT framework combined with TRADES, MART, and GAIRAT, respectively, under the AA attack on the three datasets. As shown in Figure 3, on ResNet18 model, PIAT boosts the robustness of various AT methods against the AA attack over all the three datasets. The results demonstrate that we can easily incorporate other AT methods into our PIAT

Table 2. The accuracy (%) of our method and AT baselines under adversarial attacks on CIFAR10 and CIFAR100 datasets with WRN-32-10 model.

DATASET	METHOD	CLEAN	PGD^{20}	AA
CIFAR10	PGD-AT	**86.87**	48.77	47.78
	TRADES	82.13	55.14	50.38
	MART	81.57	56.44	49.58
	GAIRAT	82.97	59.16	40.28
	PIAT	85.56	52.80	48.35
	PIAT +TRADES	82.08	58.93	53.73
	PIAT +MART	79.88	59.51	52.84
	PIAT +GAIRAT	84.66	**62.44**	44.21
	PIAT +NMSE	85.04	58.04	**53.83**
CIFAR100	PGD-AT	59.30	28.13	23.99
	TRADES	57.99	31.97	26.76
	MART	55.19	31.16	26.46
	GAIRAT	53.90	26.09	21.69
	PIAT	60.09	34.46	29.47
	PIAT +TRADES	59.78	34.52	29.25
	PIAT +MART	54.32	34.87	28.79
	PIAT +GAIRAT	56.86	32.82	26.14
	PIAT +NMSE	**61.04**	**35.15**	**30.07**

Table 3. The accuracy (%) of AT methods with or without NMSE regularization under adversarial attacks on CIFAR10 and CIFAR100 dataset with ResNet18 model.

DATASET	METHOD	CLEAN	PGD^{20}	AA
CIFAR10	PGD-AT	84.28	50.29	46.33
	PGD-AT+NMSE	**84.77**	51.56	46.60
	PIAT	79.08	51.81	47.08
	PIAT +NMSE	80.76	**53.55**	**48.70**
CIFAF100	PGD-AT	58.48	28.36	23.85
	PGD-AT+NMSE	**58.88**	29.55	24.82
	PIAT	58.84	29.11	24.15
	PIAT +NMSE	54.34	**31.11**	**25.79**

framework without incurring any additional cost to achieve better performance.

We also do ablation study on the WRN-32-10 model and report the results in Table 2. Specifically, our PIAT framework significantly boosts the robust accuracy of TRADES, gaining the improvement of 3.35% and 2.49% on CIFAR10 and CIFAR100 under the AA attack, respectively. Moreover, our PIAT framework significantly enhances the robust accuracy of both MART and GAIRAT, leading to a noteworthy improvement margin. Our framework leads to higher robust accuracy when combined with other AT methods on two models, indicating that PIAT has good flexibility and generalization.

NMSE Regularization To evaluate the effectiveness of our proposed NMSE regularization, we observe the performance of PGD-AT and PIAT with or without NMSE regularization. Table 3 reports the accuracy of ResNet18 models against PGD and AA attack on the CIFAR10 and CIFAR100 datasets. As shown in Table 3, both PGD-AT and PIAT achieve better robustness with NMSE regularization than that without NMSE. For instance, the accuracy of PGD-AT against AA attack gains absolutely 0.27% and 0.97% on CIFAR10 and CIFAR100, respectively. It indicates that the NMSE regularization greatly helps the model learn the features of adversarial examples.

Table 4. The accuracy (%) of NMSE and ALP under adversarial attacks on CIFAR10 and CIFAR100 datasets with ResNet18 model.

DATASET	METHOD	CLEAN	PGD^{20}	AA
CIFAF10	ALP	79.74	**52.37**	46.13
	NMSE	**84.77**	51.56	**46.60**
CIFAF100	ALP	57.29	28.12	23.57
	NMSE	**58.88**	**29.55**	**24.82**

Moreover, we verify the effectiveness of NMSE regularization by comparing with ALP in Table 4. Specifically, the NMSE regularization obtains an absolute improvement of 0.47% and 1.25% on CIFAR10 and CIFAR100, respectively. The results show that compared to ALP, the NMSE regularization achieves better performance in both clean and robust accuracy. More results are provided in Appendix B.3.

The hyper-parameter in Eq. 6 is used to trade off the cross-entropy loss on adversarial examples and the NMSE regularization term. We conduct further hyper-parameter analysis in Appendix B.1.

6. Conclusion

In this work, we propose a novel AT framework called PIAT to eliminate the oscillation phenomenon in the early stage and alleviate the overfitting issue in the later stage of AT by considering the parameters of previous training epochs. We further suggest to use Normalized Mean Square Error (NMSE) as regularization to align clean and adversarial examples, that focuses more on the relative magnitude of the output logits rather than the absolute magnitude. Extensive experiments verify that our framework can eliminate the oscillation phenomenon and alleviate the overfitting issue of AT. Furthermore, combining PIAT with the NMSE loss improves the model robustness without extra computational cost. In addition, PIAT is flexible and general, and various adversarial training methods can be combined into our framework to further boost their performance. Our work shows that the historical information of adversarial training process is very useful.

Acknowledgments

This work is supported by National Natural Science Foundation of China (U22B2017,62076105).

References

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 274–283, 2018.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pp. 39–57, 2017.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 2206–2216, 2020.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193, 2018.

Dong, Y., Xu, K., Yang, X., Pang, T., Deng, Z., Su, H., and Zhu, J. Exploring memorization in adversarial training. In *International Conference on Learning Representations*, 2022.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Hendrycks, D., Lee, K., and Mazeika, M. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning*, pp. 2712–2721, 2019.

Jia, X., Zhang, Y., Wu, B., Ma, K., Wang, J., and Cao, X. Las-at: Adversarial training with learnable attack strategy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13398–13408, 2022.

Kannan, H., Kurakin, A., and Goodfellow, I. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018a.

Kannan, H., Kurakin, A., and Goodfellow, I. J. Adversarial logit pairing. *CoRR*, abs/1803.06373, 2018b.

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

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, pp. 6391–6401, 2018.

Li, Q., Guo, Y., Zuo, W., and Chen, H. Squeeze training for adversarial robustness. In *International Conference on Learning Representations*, 2023.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Meng, D. and Chen, H. Magnet: A two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 135–147, 2017.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Pang, T., Lin, M., Yang, X., Zhu, J., and Yan, S. Robustness and accuracy could be reconcilable by (proper) definition. In *International Conference on Machine Learning*, pp. 17258–17277, 2022.

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

Shafahi, A., Najibi, M., Ghiasi, M. A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. Adversarial training for free! *Advances in Neural Information Processing Systems*, 32, 2019.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Wang, H. and Wang, Y. Self-ensemble adversarial training for improved robustness. In *The Tenth International Conference on Learning Representations, ICLR*, 2022.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2019.

Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., and Yuille, A. Adversarial examples for semantic segmentation and object detection. In *IEEE International Conference on Computer Vision*, pp. 1369–1378, 2017.

Yu, C., Han, B., Shen, L., Yu, J., Gong, C., Gong, M., and Liu, T. Understanding robust overfitting of adversarial training and beyond. In *International Conference on Machine Learning*, pp. 25595–25610, 2022.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In *Proceedings of the British Machine Vision Conference*, 2016.

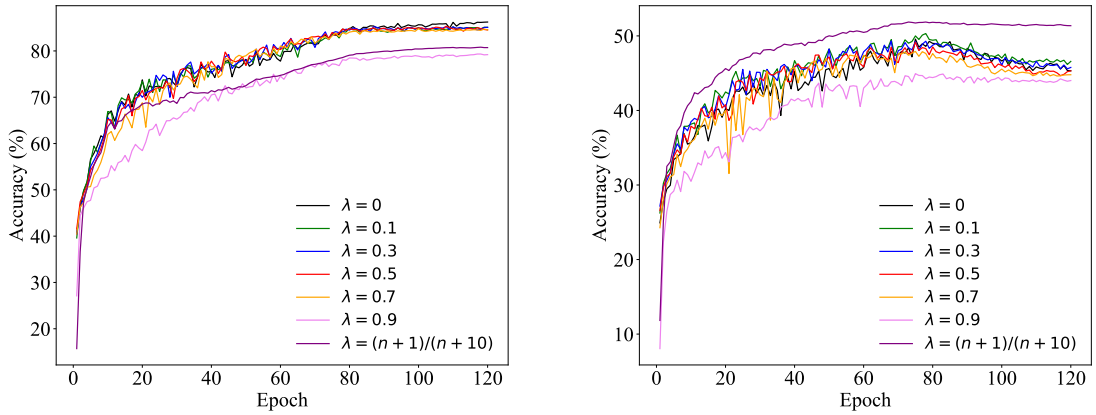
Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 7472–7482, 2019.

Zhang, J., Xu, X., Han, B., Niu, G., Cui, L., Sugiyama, M., and Kankanhalli, M. Attacks which do not kill training make adversarial learning stronger. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 11278–11287, 2020a.

Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*, 2020b.

A. Training Details

For all the experiments, we train ResNet18 (WRN-32-20) using SGD with 0.9 momentum for 120 (180) epochs. The weight decay is 3:5 10 3 for ResNet18 and 7 10 4 for WRN-32-10 on the three datasets. The initial learning rate for ResNet18 (WRN-32-10) is 0.01 (0.1) till epoch 60 (90) and then linearly decays to 0.001 (0.01), 0.0001 (0.001) at epoch 90 (135) and 120 (180). To address the cold boot problem of training, we perform standard training on clean data for the first 10 epochs, and then perform adversarial training. For crafting adversarial examples, the maximum perturbation of each pixel is $= \frac{8}{255}$ with the PGD step size $= \frac{2}{255}$ and step number of 10. For the baseline of TRADES, we adopt = 6 for the best robustness.



(a) Test accuracy on clean examples

(b) Test accuracy on adversarial examples

Figure 4. The accuracy on clean and adversarial examples of ResNet18 models trained by PIAT with different λ on the CIFAR10 dataset. n denotes the current number of training epochs.

B. Further Experiments

B.1. Hyper-parameter Analysis

The hyper-parameter in Eq. 3 is used to control the trade-off between previous and current parameters for PIAT. In Section 3, we intuitively suggest that should change over the course of training, instead of using a fixed value. To verify this point, we compare the accuracy of PIAT combined with NMSE using fixed = 0; 0:1; 0:3; 0:5; 0:7; 0:9 and our variable as in Eq. 4. Figure 4 illustrates the results on the CIFAR10 dataset. It is obvious that the variable strategy of is vital to alleviate the oscillations in the early stage and the overfitting issue in the later stage of the AT process.

The hyper-parameter in Eq. 6 is used to trade off the cross-entropy loss on adversarial examples and the NMSE regularization term. We study the accuracy of PIAT combined with NMSE using different values of . Figure 5 illustrates the results on the CIFAR10 dataset when we take = 3; 4; 5; 6. It indicates that the defense efficacy of our method is not sensitive to . Similar observations can be obtained on the CIFAR-100 dataset. Therefore, we set = 5 in our experiments for an appropriate trade-off between the accuracy on clean and adversarial examples.

B.2. Loss Landscape

To comprehensively evaluate the efficacy of our framework, we refer to the method proposed by Li et al. (2018) and compare the model obtained by PIAT framework and PGD-AT in three dimension (3D). Let $\mathbf{u}$ and $\mathbf{v}$ be two random direction vectors sampled from the Gaussian distribution. We plot the loss landscape around of the following equation when inputting the same data, where $m_1; m_2$ 2 [1; 1]:

$$L(\ ; \mathbf{u}; \mathbf{v}) = L \quad + m_1 \frac{\mathbf{u}}{jj\mathbf{u}jj} + m_2 \frac{\mathbf{v}}{jj\mathbf{v}jj} \quad : \tag{7}$$

Figure 6 illustrates the shape of the 3D landscape map. We observe that compared with PGD-AT, the model trained using the

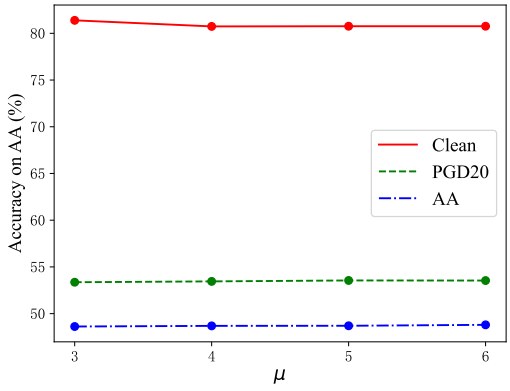
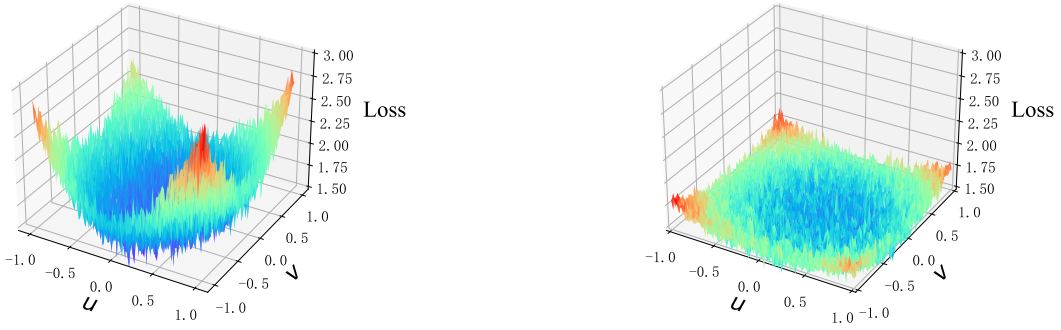


Figure 5. The clean and robust accuracy of different hyper-parameters of NMSE loss combined with PIAT framework on CIFAR10.



(a) Loss landscape of PGD-AT

(b) Loss landscape of PIAT

Figure 6. Illustration of the loss landscape in three dimensions. Our PIAT framework has a flatter loss landscape than PGD-AT.

PIAT framework exhibits less fluctuation in the loss landscape under the same perturbation. Compared with PGD-AT, the landscape obtained using PIAT framework indicates that the model converges to a flatter area and has better robust accuracy.

B.3. Comparison between ALP and NMSE

In order to demonstrate the effectiveness of the NMSE regularization, we conduct experiments on CIFAR10 and CIFAR100 using the ResNet18 model. The results show that compared to ALP, the NMSE loss regularization performs better on clean examples and achieves higher accuracy against AA attacks. Specifically, the NMSE regularization obtains an absolute improvement of 0.47% and 1.25% on CIFAR10 and CIFAR100, respectively. These results suggest that the relative magnitude of logits is a more reasonable metric than the absolute magnitude of logits.

Table 5. The accuracy (%) of NMSE and ALP under adversarial attacks on CIFAR10 and CIFAR100 datasets with ResNet18 model.

DATASET	METHOD	CLEAN	PGD^{20}	PGD^{100}	MIM	CW	AA
CIFAF10	ALP	79.74	**52.37**	**52.30**	**53.29**	49.60	46.13
	NMSE	**84.77**	51.56	51.61	52.98	**50.94**	**46.60**
CIFAF100	ALP	57.29	28.12	28.21	28.96	26.84	23.57
	NMSE	**58.88**	**29.55**	**29.50**	**30.47**	**28.18**	**24.82**