RELIABLY FAST ADVERSARIAL TRAINING VIA LATENT ADVERSARIAL PERTURBATION

Geon Yeong Park, Sang Wan Lee
Department of Bio and Brain Engineering
KAIST
{pky3436,sangwan}@kaist.ac.kr

ABSTRACT

While multi-step adversarial training is widely popular as an effective defense method against strong adversarial attacks, its computational cost is notoriously expensive, compared to standard training. Several single-step adversarial training methods have been proposed to mitigate the above-mentioned overhead cost; however, their performance is not sufficiently reliable depending on the optimization setting. To overcome such limitations, we deviate from the existing input-space-based adversarial training regime and propose a single-step latent adversarial training method (**SLAT**), which leverages the gradients of latent representation as the latent adversarial perturbation. We demonstrate that the ℓ_1 norm of feature gradients is implicitly regularized through the adopted latent perturbation, thereby recovering local linearity and ensuring reliable performance, compared to the existing single-step adversarial training methods. Because latent perturbation is based on the gradients of the latent representations which can be obtained for free in the process of input gradients computation, the proposed method costs roughly the same time as the fast gradient sign method. Experiment results demonstrate that the proposed method, despite its structural simplicity, outperforms state-of-the-art accelerated adversarial training methods.

1 INTRODUCTION

Although several studies have suggested the use of deep learning methods to solve challenging tasks, adversarial vulnerability (Szegedy et al. (2013)) is one of the remaining major challenges while employing deep learning to safety-critical applications. Adversarial training (AT) approaches aim to mitigate the problem by training the model on generated adversarial examples. Although PGD AT (Madry et al. (2017)) is one of the most effective training methods, it consumes a considerable training time because it relies on multiple projected gradient descent steps to generate the adversaries. The AT based on Fast Gradient Sign Method (FGSM; Goodfellow et al. (2014)) reduces the training time; however, recent works (Madry et al. (2017); Tramèr et al. (2017b;a)) have identified the FGSM's vulnerability to the sophisticated adversaries.

The trade-off between adversarial robustness and computational cost has facilitated the development of accelerated and trustworthy AT methods. Shafahi et al. (2019b) significantly reduced the computational burden by presenting a *free* AT method that updates both model parameters and adversarial perturbation through a single shared backward propagation. Wong et al. (2020) proposed a *fast* adversarial training based on the discovery that a slight modification in the FGSM training method such as random initialization allows it to achieve an adversarial robustness on par with PGD AT. They also discovered the *catastrophic overfitting* problem of FGSM AT, wherein the model suddenly loses its robustness during training within an epoch.

Although substantial technical advances have been made with regard to the above-mentioned methods, recent works have reported that such approaches are not sufficiently reliable. Andriushchenko & Flammarion (2020) demonstrated that fast adversarial training still suffers from the catastrophic overfitting, owing to the deteriorated local linearity of neural networks. Kim et al. (2020) found that the fast adversarial training suddenly loses its robustness and eventually collapses when a simple

multi-step learning rate schedule is used. Li et al. (2020) reported that although fast adversarial training may recover quickly, it still temporally exhibits catastrophic overfitting.

This remaining problem in AT motivates us to explore novel ways to improve the reliability of single-step AT, without bearing a considerable training time. In this study, we demonstrate that the single-step latent adversarial training (**SLAT**) with the latent adversarial perturbation operates more effectively and reliably compared to the other single-step adversarial training variants. While many of existing adversarial training methods require multiple gradient computations which is inevitably time-consuming, we exploit the gradients of latent representations from multiple layers in *parallel* for the synergistic generation of adversary. Note that the gradients of latent representations can be obtained for *free* in the process of computation of input gradients.

The contribution of this study is summarized as follows. First, we propose that the local linearity of neural networks can be regularized without any cost of training time by adopting latent adversarial perturbation, unlike the gradient alignment (GA) regularization (Andriushchenko & Flammarion (2020)), which is three times slower compared to FGSM training. In particular, we demonstrates that the ℓ_1 norm of feature gradients is implicitly regularized by introducing latent adversarial perturbation, which closes the gap between the loss function of neural networks and its first-order approximation. As the latent adversarial perturbation is adopted across multiple latent layers, the synergistic regularization effect can be expected. Second, we demonstrate that **SLAT** outperforms the state-of-the-art accelerated adversarial training methods, while achieving performance comparable to PGD AT.

2 LATENT ADVERSARIAL PERTURBATION

We begin with formulating the generalized min-max adversarial training objective in terms of latent adversarial perturbation. Let $(x \in \mathcal{X}, y \in \mathcal{Y}) \sim D$ be the pair of sample and label instance generated from the distribution D, given sample space $\mathcal{X}$ and label space $\mathcal{Y}$. We represent $f_l(\cdot)$ as the function defined by the l-th layer, and $h_l(x)$ as the latent-representation vector given sample x, where $h_0(x) \triangleq x$. Precisely, $h_l(x) = f_l(f_{l-1}(\dots(f_1(x)))) = f_l(h_{l-1}(x))$, where $f_l(h_{l-1}(x)) = \phi(W_l h_{l-1}(x) + b_l)$, given the l-th weight matrix W_l, bias vector b_l, and the activation function $\phi(\cdot)$. Note that the latent adversarial perturbation is not considered yet. We denote the L-layer neural networks as a function $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, parameterized by $\boldsymbol{\theta} = \{W_1, \dots, W_L, b_1, \dots, b_L\}$: $f_{\boldsymbol{\theta}}(x) = f_L(f_{L-1}(\dots(f_1(x))))$. For simplicity, let $f_{m:n}(x) = f_n(f_{n-1}(\dots(f_m(h_{m-1}(x)))))$, for any $1 \leq m < n \leq L$.

In this study, we investigate the benefits of latent adversarial perturbation, which is built based on the gradients of latent representations. Let $K \subseteq \{0, \dots, L-1\}$ be the subset of layer indexes injected with adversarial perturbation, where the 0th layer represents the input layer. We denote the adversarial perturbation given x for the k-th layer as $\delta_k(x)$, where $k \in K$, and the set of adversarial perturbations is $\boldsymbol{\delta} = \{\delta_k(x)\}_{\forall k \in K}$.

To examine the marginalized effect of latent adversarial perturbations, we define the *accumulated* perturbation in layer $L-1$ as $\hat{\delta}_{\ell-1}(x)$, where $\hat{\delta}_{\ell-1}(x)$ originates from the forward propagation of $\boldsymbol{\delta}$. Note that we virtually introduced the accumulated perturbation for the sake of the analysis. In practice, each latent adversarial perturbation is applied layer-wise. Thus, $h_{l+1}(x) = f_l(h_l(x) + \delta_l(x)), \forall l \in K$. The detailed pseudo-code is provided in the supplementary material.

The optimal set of parameters $\boldsymbol{\theta}^*$ and adversarial perturbations $\boldsymbol{\delta}^*$ can be obtained by solving the following min-max problem:

$$\boldsymbol{\theta^*}, \boldsymbol{\delta^*} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{(x,y)\sim D} \max_{\boldsymbol{\delta}} \left[L\Big(f_L\big(h_{L-1}(x) + \hat{\delta}_{\ell-1}(x)\big), y \Big) \right]. \tag{1}$$

While it is difficult to obtain $\hat{\delta}_{\ell-1}(x)$ in a closed form given highly nonlinear function $f_{\boldsymbol{\theta}}(\cdot)$, we can approximate it by making a reasonable assumption that $\delta_k(x)$ is sufficiently small. The approximation is based on (Camuto et al. (2020)) which examined the effect of Gaussian Noise Injection (GNI) into multiple latent layers. The adversarial perturbation accumulated on the layer $L-1$ can be expressed as follows:

Proposition 1. *Consider an L layer neural network, with the latent adversarial perturbations $\delta_k(x)$ being applied at each layer $k \in K$. Assuming the Hessians, of the form $\nabla^2 h_l(x)|_{h_m(x)}$ where l, m*

are the index over layers, are finite. Then the perturbation accumulated at the layer $L-1$, $\hat{\delta}_{\ell-1}(x)$, is approximated by:

$$\hat{\delta}_{\ell-1}(x) = \sum_{k \in K} \mathbf{J}_k(x)\delta_k(x) + O(\gamma), \tag{2}$$

where $\mathbf{J}_k(x) \in \mathbb{R}^{N_{L-1} \times N_k}$ represents each layer's Jacobian; $\mathbf{J}_k(x)_{i,j} = \frac{\partial h_{L-1}(x)_i}{\partial h_k(x)_j}$, given the number of neurons in layer $L-1$ and k as N_{L-1} and N_k, respectively. $O(\gamma)$ represents higher order terms in $\boldsymbol{\delta}$ that tend to zero in the limit of small perturbation.

The detailed proof is provided in the supplementary material. Based on the framework (1) and Proposition 1, we provide the details of the proposed latent adversarial perturbation. By neglecting the higher order terms in Proposition 1, the linear approximation of the loss function $L\Big(f_L\big(h_{L-1}(x) + \hat{\delta}_{\ell-1}(x)\big), y\Big)$ is as follows:

$$\begin{aligned} & L\Big(f_L\big(h_{L-1}(x) + \hat{\delta}_{\ell-1}(x)\big), y\Big) \\ & \approx L\Big(f_L\big(h_{L-1}(x), y\big)\Big) + \nabla_{h_{L-1}(x)} L\Big(f_L\big(h_{L-1}(x)\big), y\Big)^T \hat{\delta}_{\ell-1}(x) \\ & = L\Big(f_L\big(h_{L-1}(x), y\big)\Big) + \nabla_{h_{L-1}(x)} L\Big(f_L\big(h_{L-1}(x)\big), y\Big)^T \sum_{k \in K} \mathbf{J}_k(x)\delta_k(x). \end{aligned} \tag{3}$$

Then, we approximate the solution of the inner maximization problem in (1), as similarly done in FGSM:

$$\begin{aligned} \delta_k(x) &= \eta_k \cdot sign\Big(\mathbf{J}_k(x)^T \nabla_{h_{L-1}(x)} L\big(f_L(h_L(x)), y\big)\Big) \\ &= \eta_k \cdot sign\Big(\nabla_{h_k(x)} L\big(f_{\boldsymbol{\theta}}(x), y\big)\Big), \forall k \in K, \end{aligned} \tag{4}$$

where η_k is the step size for the k-th layer. Then, the explicit regularizer of latent adversarial perturbation in (3) is derived as follows:

$$\begin{aligned} L_{latent} &= \sum_{k \in K} \eta_k \cdot sign\Big(\nabla_{h_k(x)} L(f_{\boldsymbol{\theta}}(x), y)\Big) \circ \nabla_{h_k(x)} L(f_{\boldsymbol{\theta}}(x), y) \\ &= \sum_{k \in K} \eta_k \cdot ||\nabla_{h_k(x)} L(f_{\boldsymbol{\theta}}(x), y)||_1, \end{aligned} \tag{5}$$

where $\circ$ represents dot product.

It affords us the theoretical insights that the latent adversarial perturbation leads to the implicit regularization of ℓ_1 norm of the feature gradients. Although the uses of input gradient regularization in adversarial defense (Ross & Doshi-Velez (2018)) or explainable machine learning (Ross et al. (2017); Smilkov et al. (2017)) have been widely recognized, the effects of feature gradient regularization on adversarial robustness have been poorly understood. To address this, we establish a connection between the feature gradient regularization and local linearity. For a better linear approximation, the linear approximation error $R_k(x)$ should be constrained:

$$\Big|L\Big(f_{k:L}\big(h_{k-1}(x) + \epsilon\big), y\Big) - L\Big(f_{k:L}\big(h_{k-1}(x)\big), y\Big) - \Big\langle \nabla_{h_{k-1}(x)} L\Big(f_{k:L}\big(h_{k-1}(x)\big), y\Big), \epsilon \Big\rangle\Big|, \tag{6}$$

where ϵ is a perturbation with sufficiently small size, and $R_k(x)$ is defined with the function $f_{k:L}(\cdot)$ for an arbitrary $k \in K$. Note that $R_k(x)$ includes the second-order term $\big\langle \epsilon, H_k(x)\epsilon \big\rangle$ where $H_k(x) = \nabla^2_{h_{k-1}(x)} L\big(f_{k:L}(h_{k-1}(x)), y\big)$. Let ∇_{k-1} be a shorthand for $\nabla_{h_{k-1}(x)} L\big(f_{k:L}(h_{k-1}(x)), y\big)$. By the low-rank approximation of the Hessian matrix (Martens et al. (2012)), let $H_k(x) \approx \nabla_{k-1} \nabla_{k-1}^T$. Then, the upper bound of the second-order term is as follows:

$$\begin{aligned} \big\langle \epsilon, H_k(x)\epsilon \big\rangle &\approx |\langle \epsilon, \nabla_{k-1} \rangle|^2 \\ &\leq ||\epsilon||_2^2 ||\nabla_{k-1}||_2^2 \\ &\leq ||\epsilon||_2^2 ||\nabla_{k-1}||_1^2, \end{aligned} \tag{7}$$

since $||x||_p \geq ||x||_q$ for $0 < p < q, \forall x \in \mathbb{R}^n$. Thus the regularization of ℓ_1 norm of feature gradients may result in a better linear approximation of the loss function. This eventually contributes to improving the reliability of FGSM which relies heavily on linear approximation of the loss function.

We further foster a close collaboration between feature gradient regularization and the minimization of adversarial loss. Inspired from (Simon-Gabriel et al. (2019)), the small variation in the loss $\triangle L_k$ caused by the latent adversarial perturbation $\delta_k(x)$ is as follows:

$$\begin{aligned}\triangle L_k &= \max_{\delta_k(x):||\delta_k(x)||\leq\eta_k} \left| L\Big(f_{k+1:L}\big(h_k(x)+\delta_k(x)\big), y\Big) - L\Big(f_{k+1:L}\big(h_k(x)\big), y\Big)\right| \\ &\approx \max_{\delta_k(x):||\delta_k(x)||\leq\eta_k} \left| \Big\langle \nabla_{h_k(x)} L\Big(f_{k+1:L}\big(h_k(x)\big), y\Big), \delta_k(x)\Big\rangle\right| \\ &= \eta_k \Big|\Big| \nabla_{h_k(x)} L\Big(f_{k+1:L}\big(h_k(x)\big), y\Big)\Big|\Big|_*,\end{aligned} \tag{8}$$

where η_k is the allowed step size for the perturbation $\delta_k(x)$. The last equality comes from the definition of the dual norm $||\cdot||_*$ of $||\cdot||$. Thus the regularization of ℓ_1 norm of feature gradients is closely related to minimizing the adversarial loss stems from $\delta_k(x)$ with limited ℓ_∞ norm.

3 Experiments

To investigate the effect of latent adversarial perturbation on adversarial robustness, we compare the adversarial robustness of several models on CIFAR-10. ℓ_∞-perturbation is used with radius $\eta_0 = 8/255$. The details regarding the simulation settings and additional experimental results are presented in the supplementary material.

Table 1: Standard and robust accuracies (%) on CIFAR-10 dataset.

Method	Standard	PGD-50-10	AutoAttack	Training time (min)
PGD-7	84.86±0.16	51.63±0.13	48.65±0.08	383.2
FGSM-GA	82.88±0.01	48.90±0.37	46.22±0.30	297.9
YOPO-5-3	82.35±1.78	34.23±3.61	32.79±3.65	62.5
Free-AT ($m = 8$)	76.57±0.19	44.15±0.30	41.02±0.20	119.4
FGSM	87.42±1.08	0.01±0.01	0.00±0.00	100.5
FGSM-RS	90.76±6.36	3.90±4.06	0.44±0.50	99.7
FGSM-CKPT ($c = 3$)	89.32±0.10	40.83±0.36	39.38±0.24	121.4
SLAT	85.91±0.31	47.06±0.03	44.62±0.11	104.6

The clean and robust accuracies for the CIFAR-10 dataset are summarized in Table 1. Note that FGSM-RS (Wong et al. (2020)) experienced a catastrophic overfitting during the training process. We found that **SLAT** reliably outperforms most of the accelerated adversarial training methods, including YOPO (Zhang et al. (2019a)), Free-AT (Shafahi et al. (2019b)), FGSM-RS, and FGSM-CKPT (Kim et al. (2020)), with respect to robust accuracy against PGD attack and AutoAttack (Croce & Hein (2020b)). Although FGSM-GA (Andriushchenko & Flammarion (2020)) and PGD-7 demonstrate superior performance than **SLAT**, both methods are much slower than the other adversarial training methods. Moreover, **SLAT** achieves higher clean accuracy than both methods. We believe that expanding the proposed framework beyond single-step AT will be an interesting future work.

4 Conclusion

In this study, we demonstrate that the latent adversarial perturbation may provide a novel breakthrough for the efficient AT. The proposed framework allows us to compensate for the local linearity without sacrificing training time. Further, we establish a bridge between latent adversarial perturbation and adversarial loss minimization. It enables us to learn adversarially robust model in a more reliable manner, compared to the fast adversarial training which lacks any form of regularization. The proposed method is fully-architecture agnostic, has only a few free parameters to tune, and is potentially compatible with many other AT methods.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (NRF-2019M3E5D2A01066267), Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No.2019-0-01371, Development of brain-inspired AI with human-like intelligence), and Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-TC1603-06.

REFERENCES

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

Alexander Camuto, Matthew Willetts, Umut Şimşekli, Stephen Roberts, and Chris Holmes. Explicit regularisation in gaussian noise injections. *arXiv preprint arXiv:2007.07368*, 2020.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pp. 1310–1320. PMLR, 2019.

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.

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

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pp. 125–136, 2019.

Daniel Jakubovitz and Raja Giryes. Improving dnn robustness to adversarial attacks using jacobian regularization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 514–529, 2018.

Hoki Kim, Woojin Lee, and Jaewook Lee. Understanding catastrophic overfitting in single-step adversarial training. *arXiv preprint arXiv:2010.01799*, 2020.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world, 2016.

Bai Li, Shiqi Wang, Suman Jana, and Lawrence Carin. Towards understanding fast adversarial training. *arXiv preprint arXiv:2006.03089*, 2020.

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.

James Martens, Ilya Sutskever, and Kevin Swersky. Estimating the hessian by back-propagating curvature. *arXiv preprint arXiv:1206.6464*, 2012.

Andrew Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Andrew Ross, Isaac Lage, and Finale Doshi-Velez. The neural lasso: Local linear sparsity for interpretable explanations. In *Workshop on Transparent and Interpretable Machine Learning in Safety Critical Environments, 31st Conference on Neural Information Processing Systems*, 2017.

Swami Sankaranarayanan, Arpit Jain, Rama Chellappa, and Ser Nam Lim. Regularizing deep networks using efficient layerwise adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Ali Shafahi, Amin Ghiasi, Furong Huang, and Tom Goldstein. Label smoothing and logit squeezing: a replacement for adversarial training? *arXiv preprint arXiv:1910.11585*, 2019a.

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! In *Advances in Neural Information Processing Systems*, pp. 3358–3369, 2019b.

Carl-Johann Simon-Gabriel, Yann Ollivier, Leon Bottou, Bernhard Schölkopf, and David Lopez-Paz. First-order adversarial vulnerability of neural networks and input dimension. In *International Conference on Machine Learning*, pp. 5809–5817. PMLR, 2019.

Mayank Singh, Abhishek Sinha, Nupur Kumari, Harshitha Machiraju, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Harnessing the vulnerability of latent layers in adversarially trained models. *arXiv preprint arXiv:1905.05186*, 2019.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, pp. 1100612. International Society for Optics and Photonics, 2019.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

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

Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017b.

BS Vivek and R Venkatesh Babu. Single-step adversarial training with dropout scheduling. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 947–956. IEEE, 2020.

Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8684–8694, 2020.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5286–5295. PMLR, 2018.

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

Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 501–509, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. *arXiv preprint arXiv:1905.00877*, 2019a.

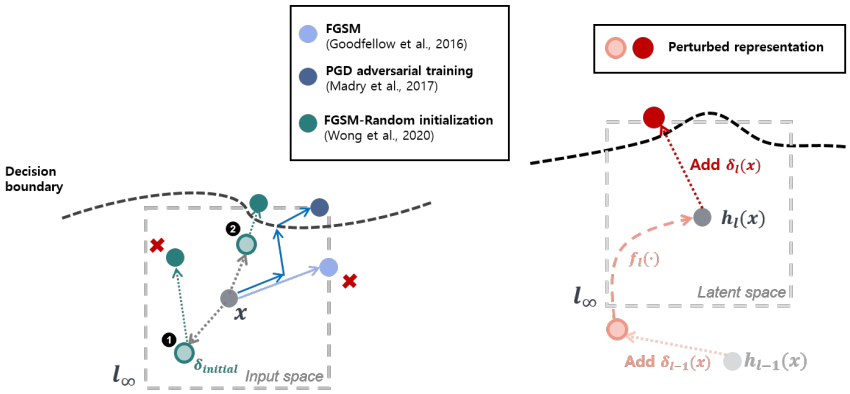
Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural networks. *arXiv preprint arXiv:1906.06316*, 2019b.

Appendix

This supplementary material is organized as follows. We begin with visually illustrating the conceptual difference between the existing and proposed method for better understanding. In section B, we present the proof for Proposition 1, which is obtained by modifying (Camuto et al. (2020)), for completeness. In section C, we provide the pseudo-code for **SLAT**. Discussion on related works is provided in section D. Network architecture, optimization setting and hyperparameter configuration is presented in section E. Extended experiment results are provided in section F.

A VISUAL ILLUSTRATION



(a) Existing approaches

(b) **SLAT**

Figure 1: Visual illustration of the conceptual difference between existing and proposed approach. **(a)** FGSM may fail to generate the appropriate adversary because it approximates the solution of inner maximization problem with a single gradient step. While PGD-based AT may generate relatively more desirable adversary, it takes multiple iterations per sample to solve the inner maximization which is computationally expensive. Uniform random initialization (Wong et al. (2020)) contributes to improving the performance of FGSM; however, the success of such initialization is not mathematically justified. **(b)** Our proposed method mitigates the suggested problems by introducing latent adversarial perturbations in parallel.

B PROOF

Proposition 1. *Consider an L layer neural network, with the latent adversarial perturbations $\delta_k(x)$ being applied at each layer $k \in K$. Assuming the Hessians, of the form $\nabla^2 h_l(x)|_{h_m(x)}$ where l, m are the index over layers, are finite. Then the perturbation accumulated at the layer $L-1$, $\hat{\delta}_{\ell-1}(x)$, is approximated by:*

$$\hat{\delta}_{\ell-1}(x) = \sum_{k \in K} \mathbf{J}_k(x)\delta_k(x) + O(\gamma), \tag{9}$$

where $\mathbf{J}_k(x) \in \mathbb{R}^{N_{L-1} \times N_k}$ represents each layer's Jacobian; $\mathbf{J}_k(x)_{i,j} = \frac{\partial h_{L-1}(x)_i}{\partial h_k(x)_j}$, given the number of neurons in layer $L-1$ and k as N_{L-1} and N_k, respectively. $O(\gamma)$ represents higher order terms in $\boldsymbol{\delta}$ that tend to zero in the limit of small perturbation.

Proof. Starting with layer 0 as the input layer, the accumulated perturbation on a layer $L-1$ can be approximated through recursion. Following the conventional adversarial training, suppose that the first layer index 0 is included in the set K. At layer 0, we apply Taylor's theorem on $h_1(x + \delta_0(x))$ around the original input x. If we assume that all values in Hessian of $h_1(x)$ is finite, i.e., $|\partial^2 h_1(x)_i / \partial x_j x_k| < \infty, \forall i, j, k$, the following approximation holds:

$$h_1(x + \delta_0(x)) = h_1(x) + \frac{\partial h_1(x)}{\partial x}\delta_0(x) + O(\kappa_0), \tag{10}$$

where $O(\kappa_0)$ represents asymptotically dominated higher order terms given the small perturbation. By accommodating $L = 2$ as a special case, we obtain the accumulated noise $\delta_{\ell-1} = \frac{\partial h_1(x)}{\partial x}\delta_0(x) + O(\kappa_0)$. Note that (10) can be generalized with an arbitrary layer index $k+1$ and perturbation $\delta_k(x)$.

Repeating this process for each layer $k \in K$ recursively, and assuming that all Hessians of the form $\nabla^2 h_l(x)|_{h_m(x)}, \forall m < l$ are finite, we obtain the accumulated perturbation for a layer $L-1$ as follows:

$$\hat{\delta}_{\ell-1}(x) = \sum_{k \in K} \frac{\partial h_{L-1}(x)}{\partial h_k(x)} \delta_k(x) + O(\gamma), \tag{11}$$

where $O(\gamma)$ represents asymptotically dominated higher order terms as the perturbation $\delta_k(x), \forall k \in K$ is sufficiently small. Denoting $\frac{\partial h_{L-1}(x)}{\partial h_k(x)}$ as the Jacobian $\mathbf{J}_k(x) \in \mathbb{R}^{N_{L-1} \times N_k}$ completes the proof. □

C PSEUDO CODE

Due to limited space, we provide the algorithm of **SLAT** in this section.

Algorithm 1: Single-step Latent Adversarial Training method (**SLAT**)

Input: Training iteration T, Number of samples N, Number of layers L, Training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, Subset of layer indexes K, Layer-wise step size η_k
Output: Adversarially robust network $f_{\boldsymbol{\theta}}$

```
for t ← 1 to T do
    for i ← 1 to N do
        for k ∈ K do
            // Compute latent adversarial perturbations
            δ_k(x_i) = η_k · sign( ∇_{h_k(x_i)} L(f_θ(x_i), y_i) )
        for l ∈ {0, ..., L − 2} do
            if l ∈ K then
                // Propagate adversarial perturbations forward
                h_{l+1}(x_i) = f_{l+1}(h_l(x_i) + δ_l(x_i))
            else
                h_{l+1}(x_i) = f_{l+1}(h_l(x_i))
        Optimize θ by the objective L(f_L(h_{L−1}(x_i)), y_i) using gradient descent.
```

D RELATED WORK

Adversarial training has been improved with the help of adversarial attack algorithms. Goodfellow et al. (2014) proposed FGSM to enable rapid generation of adversarial examples through a single-step gradient update. Subsequently, R+FGSM (Tramèr et al. (2017a)) was proposed as a means to escape from the nonlinear local vicinity through application of randomized perturbation to the sample. Kurakin et al. (2016) proposed the Basic Iterative Method whereby much finer perturbations are generated through multiple gradient steps. Based on these developments, Madry et al. (2017) generated a stronger adversary through projected gradient descent (PGD). PGD-based AT has been recognized as a more effective defense method than others, such as provable defenses (Wong & Kolter (2018); Zhang et al. (2019b); Cohen et al. (2019)), label smoothing (Shafahi et al. (2019a)), mix-up (Zhang et al. (2017)), and Jacobian regularization (Jakubovitz & Giryes (2018)). However, the complexity overhead of generating the adversary significantly limits the scalability of the PGD-based AT method.

As introduced in Section 1, several methods (Shafahi et al. (2019b); Wong et al. (2020)) have been proposed to improve the efficiency of AT. Zhang et al. (2019a) analyzed AT from the perspective of a differential game and merged the inner loop of the PGD attack and the gradient update of model

parameters. Vivek & Babu (2020) experimentally found that the adversarial robustness of FGSM AT is improved via application of dropouts to all non-linear layers. Kim et al. (2020) demonstrated that the characteristic of FGSM AT, which uses only adversary with the maximum perturbation, leads to the decision boundary distortion and therefore proposed an ad-hoc method to determine an appropriate step size. While most of the proposed methods heavily rely on the input perturbation, we explore the possibility for efficient AT using latent adversarial perturbations.

Several existing works have attempted to ascribe adversarial vulnerability to potentially problematic traits of models developed during training. Xie et al. (2019) determined that the adversarial perturbations on input samples lead to severe noise in the latent representations. Wang et al. (2020) suggested that convolutional neural networks, unlike humans, can recognize high-frequency components of images, resulting in unexpected problems such as a trade-off between robustness and accuracy. Although the aforementioned works spark interest regarding the importance of latent representations in adversarial robustness, only a few works have directly leveraged the latent adversarial perturbations for AT. Sankaranarayanan et al. (2018) used the gradients of latent representations computed from the preceding mini-batch to approximate the solution of inner-maximization. Although the proposed method has yielded modest improvements in terms of the adversarial robustness with respect to FGSM-based AT, they do not necessarily mean that a latent adversarial perturbation from a gradient of the preceding mini-batch is optimal. While Singh et al. (2019) suggested latent adversarial training for further fine-tuning of the adversarially trained model, it is contingent on performance of multi-step PGD AT. On the contrary, our approach is focused on the reduction of computational costs of AT via latent adversarial perturbations.

E EXPERIMENTAL SETUP

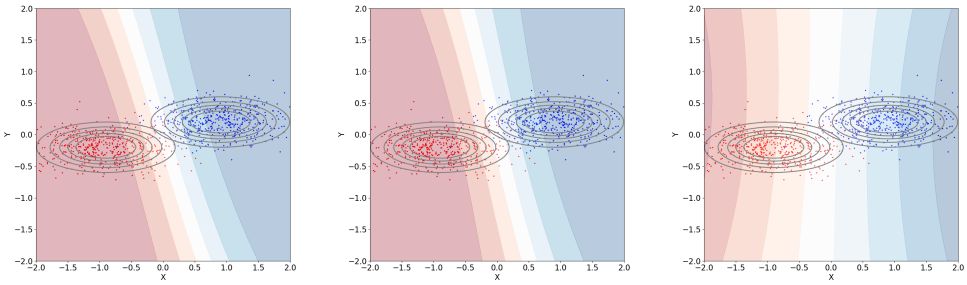
In this section, we describe network architecture, optimization setting and hyperparameter configuration. For a fair comparison, we reproduced all the other baseline results using the same back-bone architecture and the optimization settings. We validate the single-step latent adversarial training method (**SLAT**) on CIFAR-10 using Wide ResNet 28-10 (Zagoruyko & Komodakis (2016)). The latent adversarial perturbation is injected into three layers (K), including input layer and last layers in each group $conv1, conv2$ (Zagoruyko & Komodakis (2016)). We use $\eta_k = 8/255$ for every $k \in K$. Every method is trained for 30 epochs except Free-AT (Shafahi et al. (2019b)) which is trained for 72 epochs to get results comparable to the other methods. Following the setup of Andriushchenko & Flammarion (2020), we use cyclic learning rates (Smith & Topin (2019)) with the SGD optimizer. Specifically, the learning rate increases linearly from 0 to 0.2 in first 12 epochs, and then decreases linearly to 0 in left 18 epochs. Every experiment is repeated four times and the average accuracy is reported. Every experiments are run on a single GeForce Titan X.

We evaluate the adversarial robustness of several models using PGD-50-10 attack (Madry et al. (2017)), i.e., with 50 iterations and 10 restarts, and AutoAttack (Croce & Hein (2020b)) which is the ensemble of two extended PGD attacks, a white-box FAB-attack (Croce & Hein (2020a)), and the black-box Square Attack (Andriushchenko et al. (2020)). We succeeded in reproducing the robust accuracy of FGSM-RS against PGD-50-10 attack using the experimental setup reported in Wong et al. (2020), but found that catastrophic overfitting occurs when the epoch was increased to 30 (Table 1).

F EXTENDED EXPERIMENT RESULTS

Experiments on Toy Dataset. To conceptually clarify the effects of latent adversarial perturbation, we observed the behavior of the classifier on a simple binary classification problem. We generate samples (x, y) from 2D gaussian distributions, $\mathcal{N}_1(\mu, \Sigma)$, and $\mathcal{N}_2(-\mu, \Sigma)$. Inspired by Ilyas et al. (2019), we intentionally compose the input features as robust (on X-axis) and non-robust feature (on Y-axis, Figure 2). A simple neural network ($L = 2$) is implemented where the adversarial perturbation is injected to each input layer and latent layer, i.e., $K = \{0, 1\}$. Then we compare the decision boundary of binary classifiers obtained through standard training, FGSM AT, and **SLAT** ($\eta_0 = 0.1$ for both FGSM AT and **SLAT**).

Under the limited adversarial budgets, while FGSM AT still heavily relies on the predictive but non-robust feature (Figure 2b), **SLAT** learns to select a more robust feature (Figure 2c). It implies



(a) Standard (b) FGSM-AT (c) **SLAT**

Figure 2: Decision boundary of binary classifiers. Best viewed in color.

that the latent adversarial perturbation collaborates with the input adversarial perturbation, so that the effect of adversarial training can be amplified.

Evaluating adversarial robustness with different ℓ_∞ radius. To investigate the contribution of latent adversarial perturbation on reliability, we compare the robustness of FGSM-based adversarial training methods with different ℓ_∞-radius (Figure 3). The results illustrate that the latent adversarial perturbation prevents the model from losing its robustness quickly when the ℓ_∞-radius increases. Note that the FGSM-based single-step adversarial training methods (FGSM, FGSM-RS, FGSM-ckpt) are relatively sensitive to the ℓ_∞-radius, potentially due to the lack of regularization for a better linear approximation.

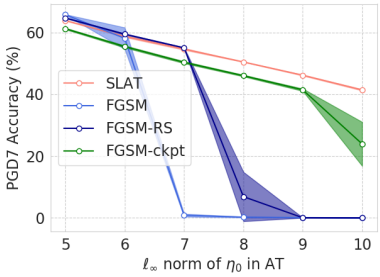
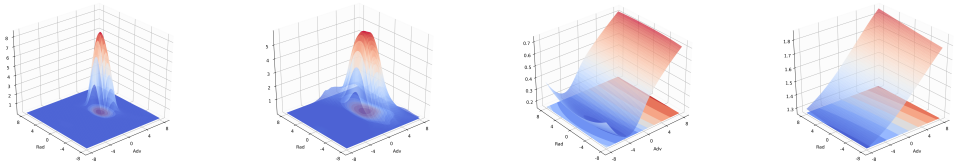


Figure 3: Robust accuracy (%) of various AT methods with different ℓ_∞ radius on CIFAR-10. The results are averaged on 4 different random seeds.

Visualizing loss landscape. To fully justify the association between latent hidden perturbation and local linearity, we visualize the loss landscape of several models. Figure 4d shows that the loss in ℓ_∞ ball is almost perfectly in linear with the adversarial direction, thus qualitatively proving that the local linearity is recovered by adopting latent perturbation. Furthermore, Figure 4a, 4b shows that the loss landscape of FGSM and FGSM-RS are highly distorted. Although FGSM-ckpt (Kim et al. (2020)) performs better than FGSM-RS, the obtained loss landscape is not perfectly linear as **SLAT** in adversarial direction.



(a) FGSM (b) FGSM-RS (c) FGSM-ckpt (d) **SLAT**

Figure 4: Visualization of loss landscape on CIFAR-10 for various models. We plot the softmax cross entropy loss projected on adversarial direction and random (Rademacher) direction with $\eta_0 = 8/255$ radius.

Measuring ℓ_1 norm of feature gradients. With respect to the analysis in the main paper, we compared the ℓ_1 norm of the stochastic gradients computed with different methods. After training, we computed the ℓ_1 norm of gradients for every representation $h_l(x)$ for all $l \in K$. Figure 5 shows that **SLAT** yields exponentially lower norm of the gradients compared to the other methods, especially on

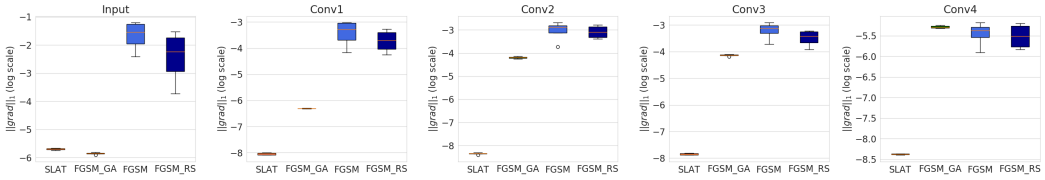
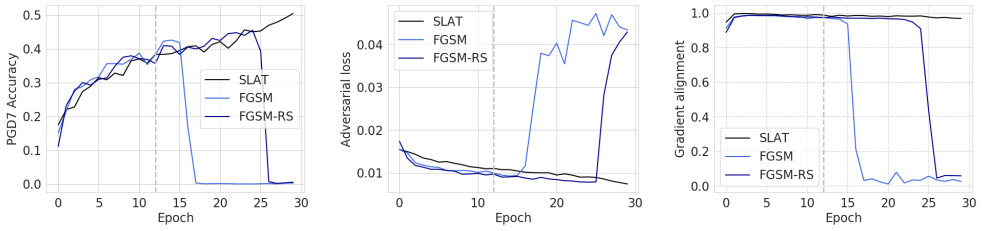


Figure 5: ℓ_1 norm of gradients on CIFAR-10 after training. The results are averaged on 4 different random seeds.

the latent representations. It is potentially due to the additional regularization by latent adversarial perturbation.

Measuring local linearity. To specify the connection between catastrophic over-fitting and local linearity, we measure the adversarial robustness and gradient alignment as done in Andriushchenko et al. (2020). We figured out that the gradient alignment suddenly drops as the model loses its adversarial robustness (Figure 6). While FGSM or FGSM-RS may rely on the early-stopping technique (Wong et al. (2020)) to empirically prevent catastrophic overfitting, it is not sufficient to outperform **SLAT** (Figure 6a).



(a) Robust accuracy (b) Adversarial loss (c) Gradient alignment

Figure 6: Demonstrating the relationship between catastrophic overfitting and local linearity. Gray dotted line indicates the epoch at maximum learning rate.

Ablation study on latent adversarial perturbation. To quantify the extent of performance improvement achieved by the latent adversarial perturbation, we compared the standard and robust accuracy of the different versions of **SLAT** (Table 2). In the case where latent adversarial perturbation is excluded, the proposed method reduces to the conventional FGSM AT. In order to take into account the effect of step size reduction (Andriushchenko & Flammarion (2020)), the version of FGSM AT with reduced step size was also compared. Table 2 shows that the vanilla FGSM AT is not sufficient to prevent the catastrophic overfitting problem regardless of step size. Moreover, we experimentally found that latent adversarial perturbations prevent the catastrophic overfitting of FGSM-RS (Wong et al. (2020)). This implies that the latent adversarial perturbation itself plays a major role in recovering the local linearity. Although the latent adversarial perturbations improve the adversarial robustness of FGSM-RS to some extent, it does not outperform **SLAT**. It is because latent adversarial perturbation was derived from the original sample rather than the perturbed sample as in (4).

Table 2: Ablation study on latent adversarial perturbation with $\eta_0 = 8/255$ (CIFAR-10).

Method	Standard	PGD-50-10
FGSM (step size$= 8/255$)	87.42	0.01
FGSM (step size$= 0.9 * 8/255$)	90.87	3.09
FGSM-RS w/ latent adversarial perturbation	82.87	44.95
SLAT	85.91	47.06

Moreover, to understand the effect of layer depth on latent adversarial training, we compared the performance of **SLAT** with different subsets of layer indexes K. Following Zagoruyko & Komodakis (2016), latent adversarial perturbation was added to the last layer of some selected blocks among

conv1, conv2, conv3, conv4 of Wide ResNet 28-10. The input layer was included equally in K for all experiments. We experimentally found that the robustness and standard accuracy decreased significantly when the latent adversarial perturbations were injected into deeper layers (Table 3). This implies that simply injecting latent adversarial perturbation to arbitrarily many hidden layers is not necessarily be effective and may over-regularize the networks. These findings provide us an interesting conjecture that the recovery of local linearity should be primarily confined to early sub-networks that recursively include other deeper sub-networks. The theoretical analysis is left for future research.

Table 3: Analysis of adversarial robustness and standard accuracy based on layer depth change.

Layers	Standard	PGD-50-10	AutoAttack
conv1, conv2 (**SLAT**)	85.91	47.06	44.62
conv3, conv4	84.60	0.00	0.00
conv1, conv2, conv3, conv4	81.41	47.3	43.58

We additionally conducted the analysis on hyperparameter sensitivity with the adversarial step size η_k (Figure 7). Standard and adversarial accuracy are measured on CIFAR-10, where the step size η_k varies from $0.6 * 8/255$ to $1.0 * 8/255$. The robust accuracy is high when $\eta_k = 8/255$. Thus, we fix η_k as $8/255$ for every dataset and layer $k \in K$. It may work better to fine-tune η_k differently depending on k. From our preliminary analyses, the unified η_k for the latent representations worked sufficiently well that we did not feel the need to fine-tune η_k for every layer. Moreover, while the trade-off between robustness and accuracy is observed, the standard accuracy for $\eta_k = 8/255$ is still superior than that of FGSM-GA or PGD-7 (Table 1).

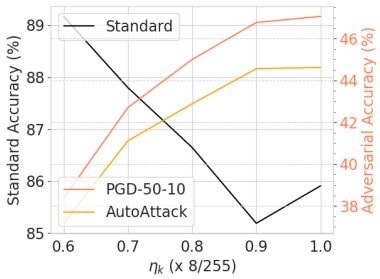


Figure 7: Impact of η_k on robustness.

Adversarial robustness on CIFAR-100. We additionally evaluate the adversarial robustness of several models on CIFAR-100. Experimental settings, such as network architecture, learning rate and the size of adversarial perturbation, are kept same as for the CIFAR-10 dataset. Table 4 clarifies the adversarial robustness gap between **SLAT** and other accelerated adversarial training methods. We leave for future work the experiments on larger scale benchmarks such as ImageNet.

Table 4: Standard and robust accuracies (%) on CIFAR-100 dataset ($\eta_0 = 8/255$).

Method	Standard	PGD-50-10	AutoAttack
PGD-7	59.59±0.17	29.58±0.24	26.00±0.20
FGSM-GA	58.63±0.17	27.53±0.10	24.07±0.15
YOPO-5-3	51.45±7.33	15.23±2.01	13.94±1.82
Free-AT ($m = 8$)	48.02±0.29	22.40±0.19	18.67±0.03
FGSM	61.96±2.17	0.00±0.00	0.00±0.00
FGSM-RS	50.96±4.57	0.00±0.00	0.00±0.00
FGSM-CKPT ($c = 3$)	73.53±0.65	0.66±0.60	0.09±0.09
SLAT	59.56±0.50	26.26±0.47	23.02±0.14