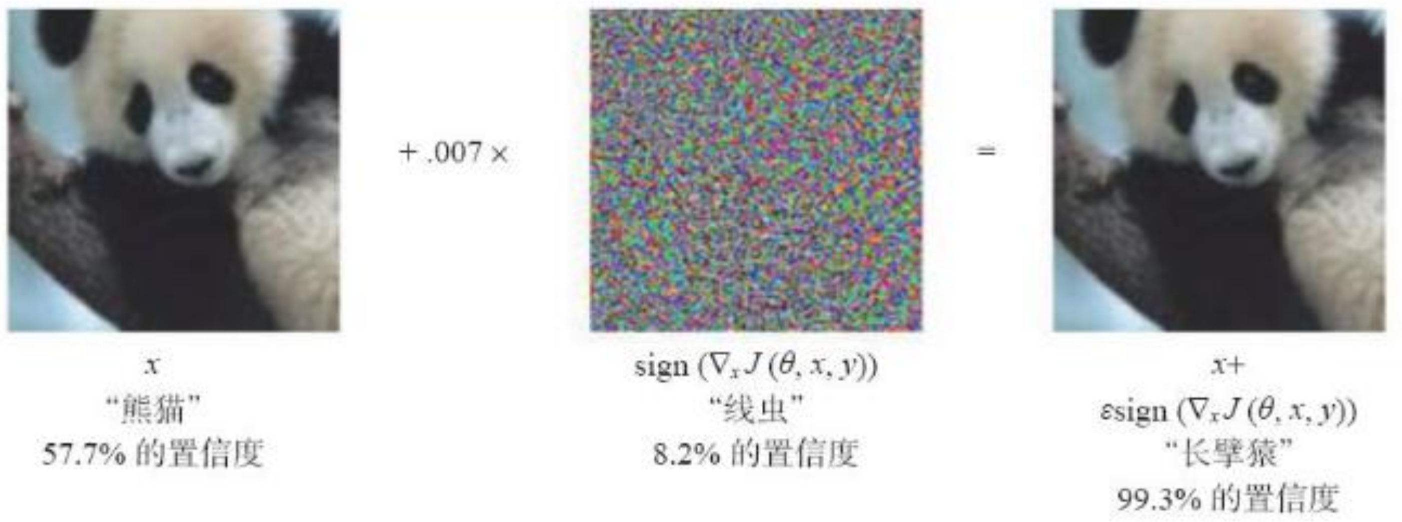
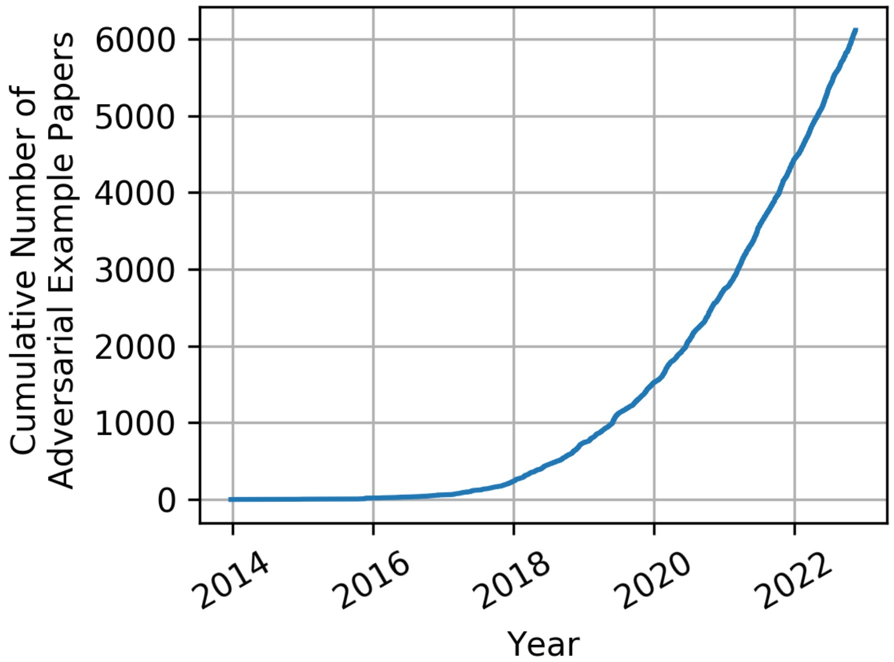
Boosting Adversarial Training from Perspective of Effectiveness and Efficiency

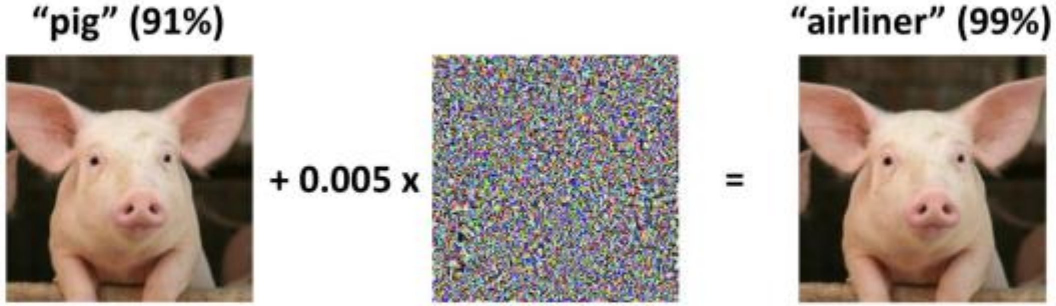
深度学习技术已经在很多领域取得了巨大的成功


+ .007 ×
=
x
“熊猫”
57.7% 的置信度
$\text{sign}(\nabla_x J(\theta, x, y))$
“线虫”
8.2% 的置信度
$x+$
$\varepsilon\text{sign}(\nabla_x J(\theta, x, y))$
“长臂猿”
99.3% 的置信度


Cumulative Number of
Adversarial Example Papers
6000
5000
4000
3000
2000
1000
0
2014
2016
2018
2020
2022
Year


#NEWS
GOOGLE SELF DRIVING CAR
CRASHES INTO A BUS

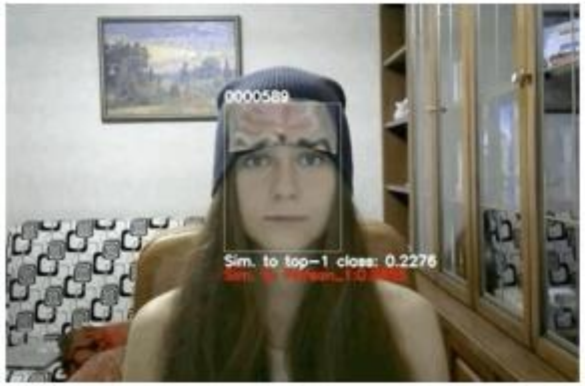

CCTV 13
新闻
CNTV
法治封面
“自动驾驶”：安全，不安全！？
事故前一分钟 司机状态轻松

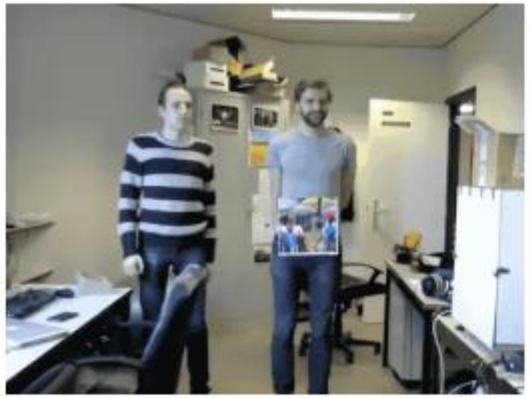


[Szegedy et al. 2014]: Imperceptible noise (adversarial examples) can fool state-of-the-art classifiers

[Sharif et al. 2016]:
Glasses that fool face recognition

Eykholt et al., Robust Physical-World Attacks on Deep Learning Visual Classification, CVPR 2018





Adversarial defense methods

- Adversarial defense
  - Robustness itself
    - Adversarial training
    - Gradient masking
    - …
  - Pre-processing module
    - Image denoise
    - Image compression
    - …

BPDA

Athalye, Anish, Nicholas Carlini, and David Wagner. "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples."ICML, 2018.

Obfuscated Gradients

- Shattered Gradients
- Stochastic Gradients
- Vanishing Gradients

Defense	Dataset	Distance	Accuracy
Buckman et al. (2018)	CIFAR	0.031 (ℓ_∞)	0%*
Ma et al. (2018)	CIFAR	0.031 (ℓ_∞)	5%
Guo et al. (2018)	ImageNet	0.005 (ℓ_2)	0%*
Dhillon et al. (2018)	CIFAR	0.031 (ℓ_∞)	0%
Xie et al. (2018)	ImageNet	0.031 (ℓ_∞)	0%*
Song et al. (2018)	CIFAR	0.031 (ℓ_∞)	9%*
Samangouei et al. (2018)	MNIST	0.005 (ℓ_2)	55%**
Madry et al. (2018)	CIFAR	0.031 (ℓ_∞)	47%
Na et al. (2018)	CIFAR	0.015 (ℓ_∞)	15%

LAS-AT: Adversarial Training with Learnable Attack Strategy(CVPR Oral)

Xiaojun Jia[1,2,†,*], Yong Zhang[3,*], Baoyuan Wu[4,5,‡], Ke Ma[6], Jue Wang[3], Xiaochun Cao[1,2,‡]
1. Institute of Information Engineering, Chinese Academy of Sciences
2. School of Cyberspace Security, University of Chinese Academy of Sciences
3. Tencent, AI Lab 4. School of Data Science, The Chinese University of Hong Kong, Shenzhen
5. Secure Computing Lab of Big Data, Shenzhen Research Institute of Big Data, Shenzhen
6. School of Computer Science and Technology, University of Chinese Academy of Sciences

目录

Content

01

Motivation

Motivation

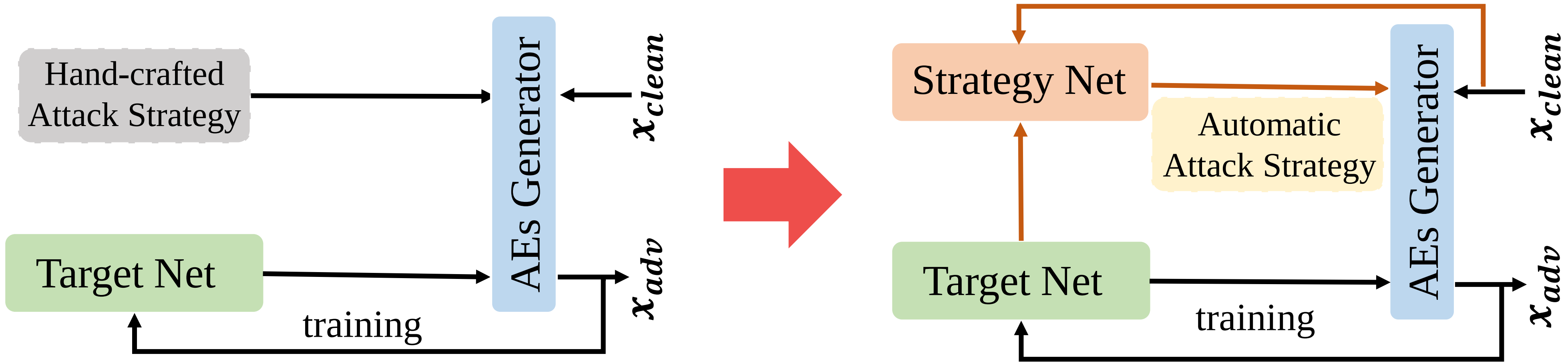
$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\max_{\boldsymbol{\delta}\in\Omega} \mathcal{L}(f_{\mathbf{w}}(\mathbf{x}+\boldsymbol{\delta}), y)]$$

1. The inner maximization problem of standard AT is to generate adversarial examples by maximizing the classification loss.
2. The inner maximization problem of standard AT is to find model parameters by minimizing the classification loss on adversarial examples.
3. The inner maximization problem can be regarded as the attack strategy that guides the creation of AEs, which is the core to improve the model robustness. A training strategy is designed accordingly, which significantly improves the network’s robustness.

Motivation

$$\mathbf{x}_{adv} := \mathbf{x} + \boldsymbol{\delta} \leftarrow g(\mathbf{x}, \mathbf{a}, \mathbf{w})$$

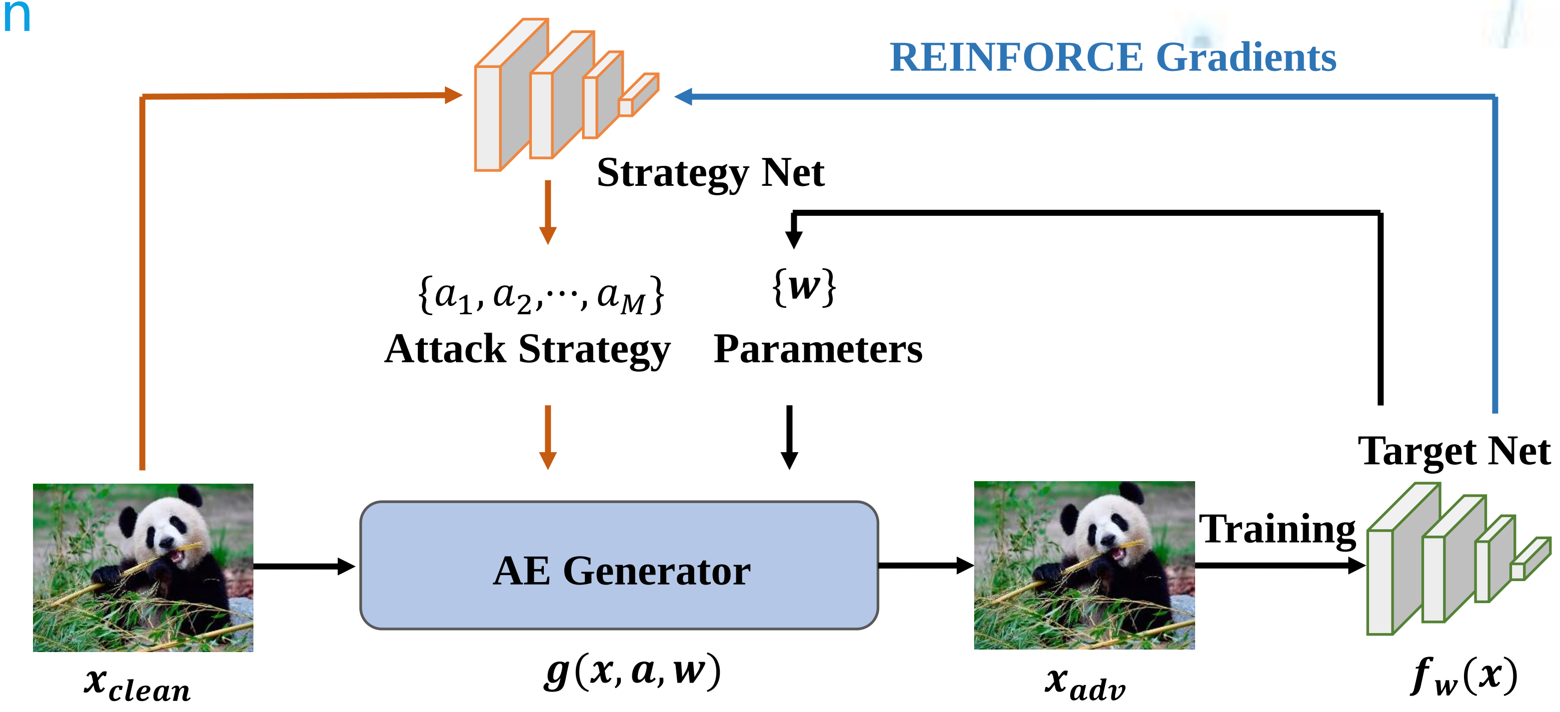
a is an attack strategy, i.e., the configuration of how to perform the adversarial attack. For example, PGD attack has three attack parameters, i.e., the attack step size, the attack iteration, and the maximal perturbation strength.



Conventional AT

LAS-AT

Contribution



Our main contributions are as follows:

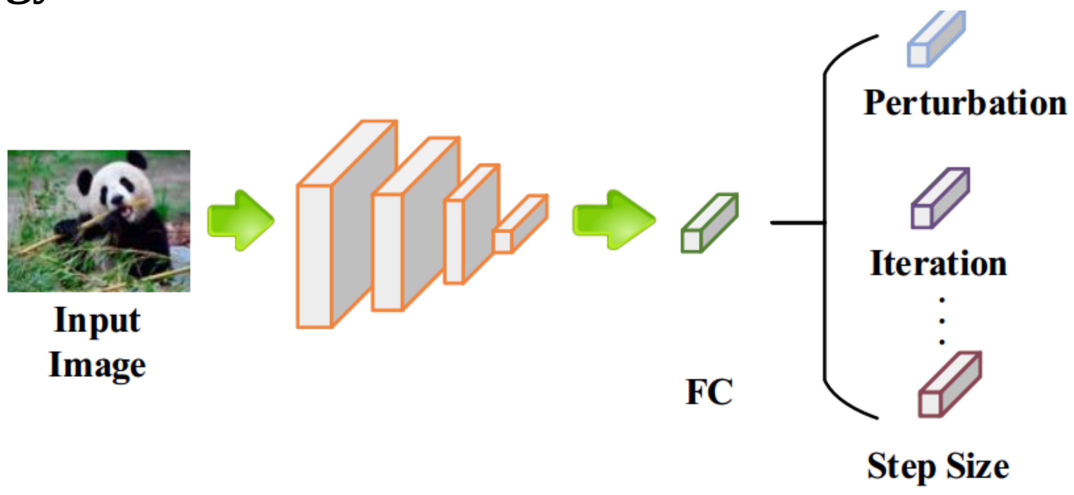
1. We propose a novel adversarial training framework by introducing the concept of "learnable attack strategy", which learns to automatically produce sample-dependent attack strategies to generate AEs. Our framework can be combined with other state-of-the-art methods as a plug-and-play component.
2. We propose two loss terms to guide the learning of the strategy network, which involve explicitly evaluating the robustness of the target model and the accuracy of clean samples.
3. We conduct experiments and analyses on three databases to demonstrate the effectiveness of the proposed method, and the proposed method outperforms state-of-the-art adversarial training methods.

02

Method

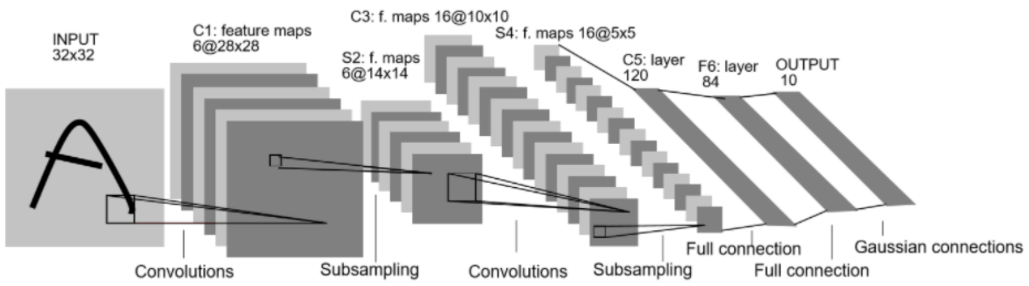
Method

Strategy Net



Given an image, the strategy network outputs an attack strategy, i.e., the configuration of how to perform the adversarial attack. A combination of the selected values for these attack parameters is an attack strategy. The strategy network captures the conditional distribution of a given x and θ.

Target Net



The target network is a convolutional network for image classification.

Adversarial Example Generator

$$\mathbf{x}_{adv} := \mathbf{x} + \boldsymbol{\delta} \leftarrow g(\mathbf{x}, \mathbf{a}, \mathbf{w})$$

g(·) is the PGD attack. The process is equivalent to solving the inner optimization problem, given an attack strategy a, i.e., finding the optimal perturbation to maximize the loss.

Method

Original Formulation of Adversarial Training:

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\, \mathcal{L}(f_{\mathbf{w}}(\mathbf{x}_{adv}), y)$$

Our Formulation of Adversarial Training:

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\max_{\boldsymbol{\theta}}\ \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})}\, \mathcal{L}(f_{\mathbf{w}}(\mathbf{x}_{adv}), y) \right]$$

It can be observed that the two networks compete with each other in minimizing or maximizing the same objective. learns to improve attack strategies according to the given samples to attack the target network. At the beginning of the training phase, the target network is vulnerable, which a weak attack can fool. Hence, the strategy network can easily generate effective attack strategies. The strategies could be diverse because both weak and strong attacks can succeed. As the training process goes on, the target network becomes more robust. The strategy network has to learn to generate attack strategies that create stronger AEs. Therefore, the gaming mechanism could boost the robustness of the target network gradually along with the improvement of the strategy network

Method

Loss of adversarial training:

$$\mathcal{L}_1(\mathbf{w},\boldsymbol{\theta}) := \mathcal{L}(f(\mathbf{x}_{adv},\mathbf{w}),y)$$

Loss of Evaluating Robustness:

$$\mathcal{L}_2(\boldsymbol{\theta}) = -\mathcal{L}(f(\mathbf{x}_{adv}^{\hat{\mathbf{a}}},\hat{\mathbf{w}}),y)$$

Loss of Predicting Clean Samples:

$$\mathcal{L}_3(\boldsymbol{\theta}) = -\mathcal{L}(f(\mathbf{x},\hat{\mathbf{w}}),y)$$

Formal Formulation:

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\max_{\boldsymbol{\theta}}\ \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})}\ [\mathcal{L}_1(\mathbf{w},\boldsymbol{\theta}) + \alpha\mathcal{L}_2(\boldsymbol{\theta}) + \beta\mathcal{L}_3(\boldsymbol{\theta})]\right]$$

Method

Optimization of target network:

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})} [\mathcal{L}_1(\mathbf{w}, \boldsymbol{\theta})].$$

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \eta_1 \frac{1}{N} \sum_{n=1}^{N} \nabla_{\mathbf{w}} \mathcal{L}\left(f(\mathbf{x}_{adv}^n, \mathbf{w}^t), y_n\right)$$

Optimization of strategy network:

$$\max_{\boldsymbol{\theta}} J(\boldsymbol{\theta}),$$

$$\text{where } J(\boldsymbol{\theta}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})} [\mathcal{L}_1 + \alpha\mathcal{L}_2 + \beta\mathcal{L}_3].$$

The biggest challenge of this optimization problem is that the process of AE generation is not differentiable, namely, the gradient can not be backpropagated to the attack strategy through the AEs. Moreover, there are some non-differentiable operations (e.g. choosing the iteration times) related to attack , which sets an obstacle to backpropagate the gradient to the strategy network.

Method

$$\begin{aligned}\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})} [\mathcal{L}_0] \\ &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \int_{\mathbf{a}} \mathcal{L}_0 \cdot \nabla_{\boldsymbol{\theta}} p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta}) d\mathbf{a} \\ &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \int_{\mathbf{a}} \mathcal{L}_0 \cdot p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta}) d\mathbf{a} \\ &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \mathbb{E}_{\mathbf{a}\sim p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})} [\mathcal{L}_0 \cdot \nabla_{\boldsymbol{\theta}} \log p(\mathbf{a}|\mathbf{x};\boldsymbol{\theta})],\end{aligned}$$

$$\Downarrow$$

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \approx \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}_0(\mathbf{x}^n;\boldsymbol{\theta}) \cdot \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{a}^n|\mathbf{x}^n).$$

$$\Downarrow$$

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \eta_2 \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^t),$$

Method

Convergence Analysis

Theorem 1. *Suppose that the objective function $\mathcal{L}_0 = \mathcal{L}_1 + \alpha\mathcal{L}_2 + \beta\mathcal{L}_3$ in (7) satisfied the gradient Lipschitz conditions w.r.t. $\boldsymbol{\theta}$ and $\mathbf{w}$, and $\mathcal{L}_0$ is μ-strongly concave in $\boldsymbol{\Theta}$, the feasible set of $\boldsymbol{\theta}$. If $\hat{\mathbf{x}}_{adv}(\mathbf{x}, \mathbf{w})$ is a σ-approximate solution of the ℓ_∞ ball with radius ϵ constraint, the variance of the stochastic gradient is bounded by a constant $\sigma^2 > 0$, and we set the learning rate of $\mathbf{w}$ as*

$$\eta_1 = \min\left(\frac{1}{L_0}, \sqrt{\frac{\mathcal{L}_0(\mathbf{w}^0) - \min_{\mathbf{w}} \mathcal{L}_0(\mathbf{w})}{\sigma^2 T L_0}}\right), \tag{14}$$

where $L_0 = L_{\mathbf{w}\boldsymbol{\theta}} L_{\boldsymbol{\theta}\mathbf{w}}/\mu + L_{\mathbf{w}\mathbf{w}}$ is the Lipschitz constants of $\mathcal{L}_0$, it holds that

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left[\|\nabla \mathcal{L}_0(\mathbf{w}^t)\|_2^2\right] \leq 4\sigma\sqrt{\frac{\Delta L_0}{T}} + \frac{5\delta L_{\mathbf{w}\boldsymbol{\theta}}^2}{\mu}, \tag{15}$$

03

Experiments

Experiments

Table 1. Test robustness (%) on the CIFAR-10 database using ResNet18. Number in bold indicates the best.

Method	PGD-AT [33]	k=1	k=10	k=20	k=40	k=60
Clean	82.56	**82.88**	82.38	82.00	82.3	82.10
PGD-10	53.15	53.71	53.89	53.53	**54.29**	53.85
Time(min)	261	1378	432	418	365	333

Table 6. Test robustness (%) on the CIFAR-10 database using ResNet18. Number in bold indicates the best.

$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_3$	clean	PGD-10	AA
✓			81.83	53.88	49.06
✓	✓		81.54	53.98	49.34
✓		✓	81.90	53.89	49.20
✓	✓	✓	**82.3**	**54.29**	**49.89**

Table 5. Test robustness (%) on the CIFAR-10 and CIFAR-100 database. Number in bold indicates the best.

Database	Target network	Method	Clean	AA
CIFAR-10	WRN70-16	Gowal *et al* [14]	85.29	57.20
		LAS-AWP(ours)	**85.66**	**57.86**
CIFAR-100	WRN34-20	LBGAT [8]	62.55	30.20
		LAS-AWP(ours)	**67.31**	**31.92**

Table 7. Test robustness (%) on the CIFAR-10 database using WRN34-10. Comparisons with Madry, CAT, DART and FAT. The results are reported in [51]. Number in bold indicates the best.

Method	Clean	FGSM	PGD-20	C&W
Madry-AT [27]	87.3	56.1	45.8	46.8
CAT [40]	77.43	57.17	46.06	42.28
DART [40]	85.03	63.53	48.70	47.27
FAT [51]	**87.97**	65.94	49.86	48.65
LAS-Madry-AT	84.95	**67.16**	**55.61**	**54.31**

Experiments

Table 2. Test robustness (%) on the CIFAR-10 database using WRN34-10. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA
PGD-AT [33]	85.17	56.07	55.08	54.88	53.91	51.69
TRADES [50]	85.72	56.75	56.1	55.9	53.87	53.40
MART [41]	84.17	58.98	58.56	58.06	54.58	51.10
FAT [51]	**87.97**	50.31	49.86	48.79	48.65	47.48
GAIRAT [52]	86.30	60.64	59.54	58.74	45.57	40.30
AWP [45]	85.57	58.92	58.13	57.92	56.03	53.90
LBGAT [8]	88.22	56.25	54.66	54.3	54.29	52.23
LAS-AT(ours)	86.23	57.64	56.49	56.12	55.73	53.58
LAS-TRADES(ours)	85.24	58.01	57.07	56.8	55.45	54.15
LAS-AWP(ours)	87.74	**61.09**	**60.16**	**59.79**	**58.22**	**55.52**

Table 3. Test robustness (%) on the CIFAR-100 database using WRN34-10. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA
PGD-AT [33]	60.89	32.19	31.69	31.45	30.1	27.86
TRADES [50]	58.61	29.20	28.66	28.56	27.05	25.94
SAT [35]	62.82	28.1	27.17	26.76	27.32	24.57
AWP [45]	60.38	34.13	33.86	33.65	31.12	28.86
LBGAT [8]	60.64	35.13	34.75	34.62	30.65	29.33
LAS-AT(ours)	61.80	33.45	32.77	32.54	31.12	29.03
LAS-TRADES(ours)	60.62	32.99	32.53	32.39	29.51	28.12
LAS-AWP(ours)	**64.89**	**37.11**	**36.36**	**36.13**	**33.92**	**30.77**

Table 4. Test robustness (%) on the Tiny Imagenet database using PreActResNet18. Number in bold indicates the best.

Method	Clean	PGD-50	C&W	AA
PGD-AT [33]	43.98	19.98	17.6	13.78
TRADES [50]	39.16	15.74	12.92	12.32
AWP [45]	41.48	22.51	19.02	17.34
LAS-AT(ours)	44.86	22.16	18.54	16.74
LAS-TRADES(ours)	41.38	18.36	14.5	14.08
LAS-AWP(ours)	**45.26**	**23.42**	**19.88**	**18.42**

Method	Clean	PGD-50	C&W	AA
Clean	**98.22**	12.63	13.28	9.77
PGD-AT	90.34	59.02	60.04	57.54
TRADES	87.35	61.95	61.40	59.99
AWP	91.82	64.94	64.69	62.24
LAS-AT(ours)	91.98	64.33	64.06	62.07
LAS-TRADES(ours)	88.67	63.26	62.40	61.09
LAS-AWP(ours)	93.17	**67.03**	**67.77**	**65.21**

Table 1. Results on GTSRB (%).

Experiments

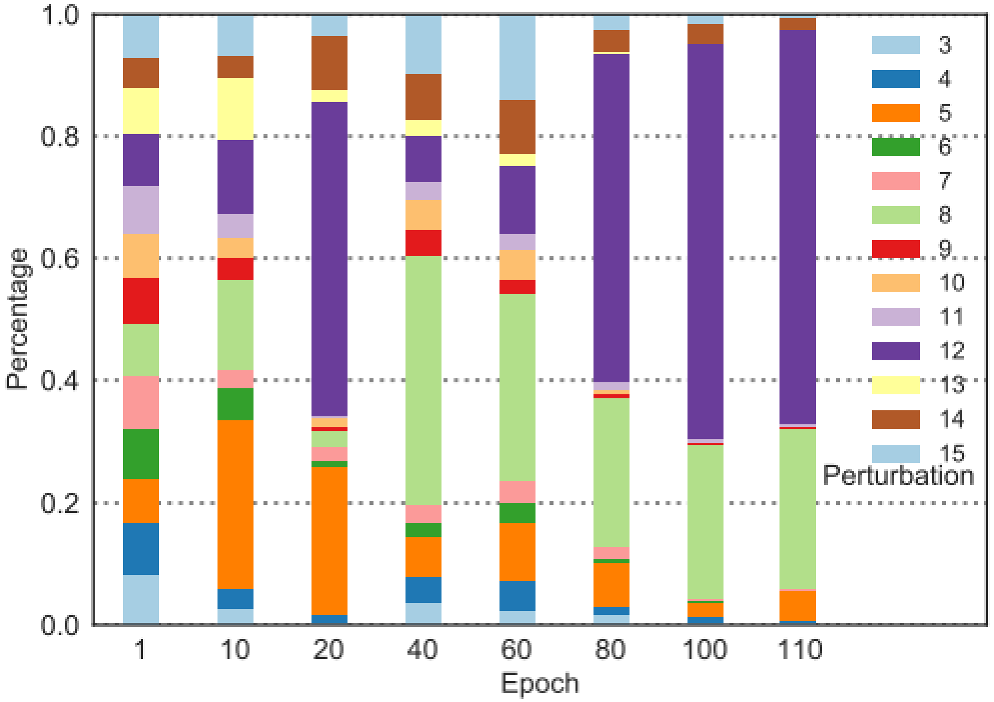


Figure 4. The distribution evolution of the maximal perturbation strength in LAS-PGD-AT during training.

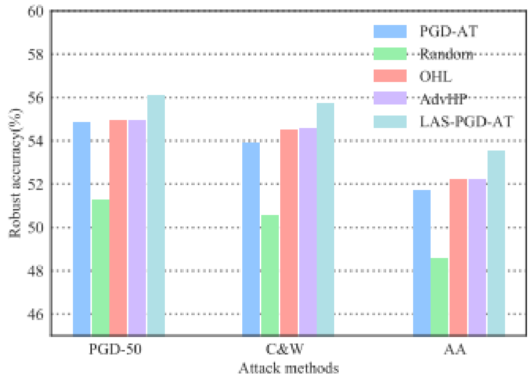


Figure 3. Comparisons with the hyper-parameter search methods using WRN34-10 on the CIFAR-10 database. x-axis represents the attack methods. y-axis represents the robust accuracy.

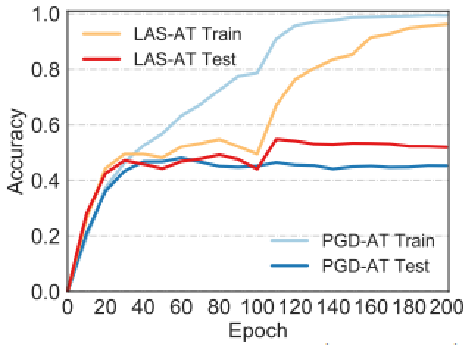
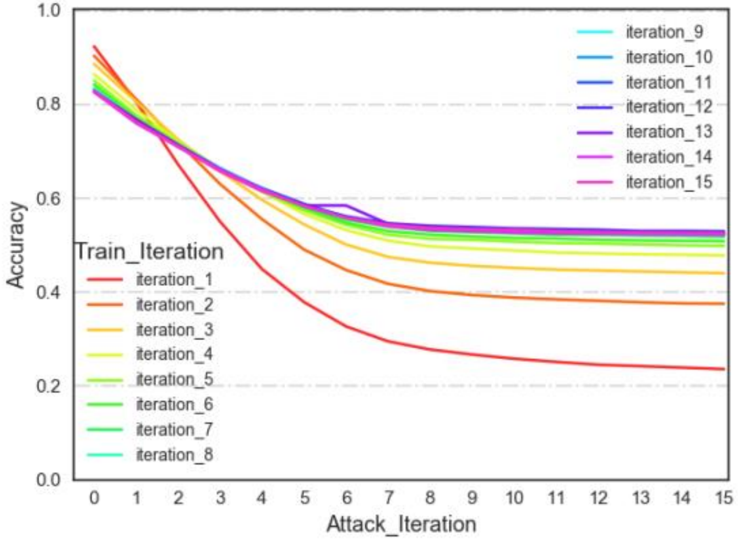
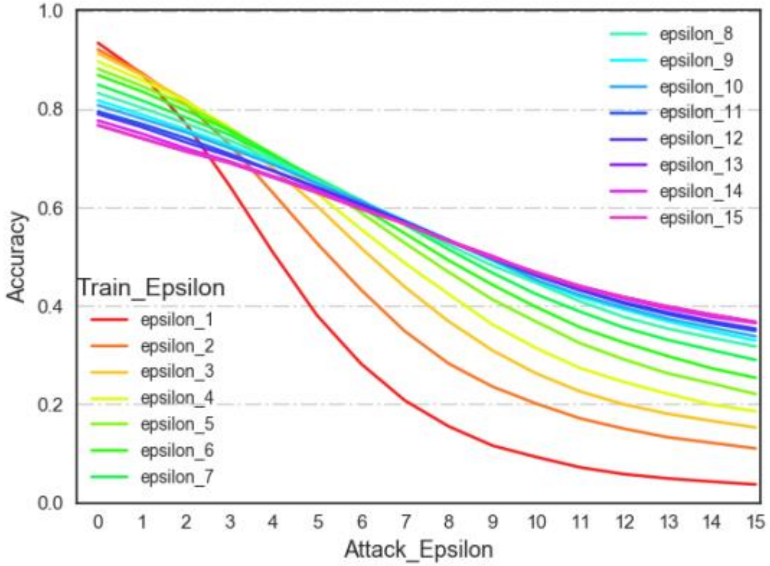


Figure 1. Robustness accuracy curves under PGD-10 attack on the training and test data of CIFAR-10.

Experiments





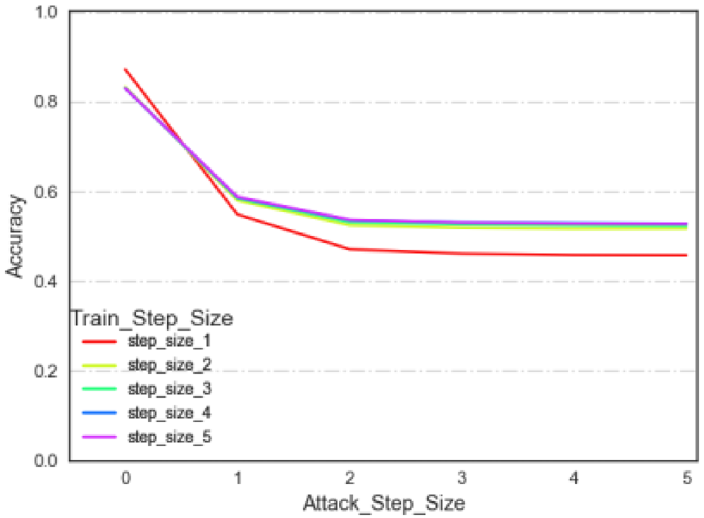


Table 1. Test robustness (%) on the CIFAR-10 database using ResNet18. Number in bold indicates the best.

Method	Clean	PGD-10	PGD-20	PGD-50	C&W	AA
AWP($I_{\text{train}} = 10, \epsilon_{\text{train}} = 8$)	80.72	55.33	54.78	54.28	51.67	49.44
AWP($I_{\text{train}} = 10, \epsilon_{\text{train}} = 15$)	66.73	52.24	52.14	52.06	48.1	47.03
AWP($I_{\text{train}} = 15, \epsilon_{\text{train}} = 8$)	80.13	55.82	55.24	55.13	51.53	49.62
LAS-AWP(ours)	**83.03**	**56.45**	**55.76**	**55.43**	**53.06**	**50.77**

Experiments

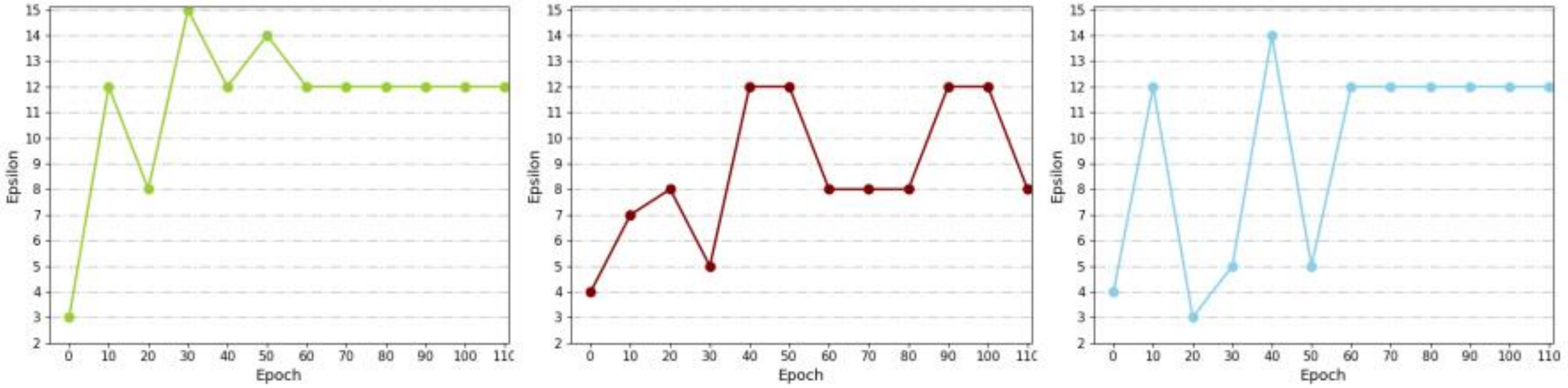


Figure 5. The evolution of the generated perturbation strength of several samples during the whole training process. X-axis represents the training epoch. Y-axis represents the perturbation strength.

Experiments

Rank	Method	Standard accuracy	AutoAttack robust accuracy	Best known robust accuracy	AA eval. potentially unreliable	Extra data	Architecture	Venue
1	Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples	69.15%	36.88%	36.88%	×	☑	WideResNet-70-16	arXiv, Oct 2020
2	Fixing Data Augmentation to Improve Adversarial Robustness *It uses additional 1M synthetic images in training.*	63.56%	34.64%	34.64%	×	×	WideResNet-70-16	arXiv, Mar 2021
3	Robustness and Accuracy Could Be Reconcilable by (Proper) Definition *It uses additional 1M synthetic images in training.*	65.56%	33.05%	33.05%	×	×	WideResNet-70-16	arXiv, Feb 2022
4	Fixing Data Augmentation to Improve Adversarial Robustness *It uses additional 1M synthetic images in training.*	62.41%	32.06%	32.06%	×	×	WideResNet-28-10	arXiv, Mar 2021
5	LAS-AT: Adversarial Training with Learnable Attack Strategy	67.31%	31.91%	31.91%	×	×	WideResNet-34-20	arXiv, Mar 2022
31	HYDRA: Pruning Adversarially Robust Neural Networks *Compressed model*	88.98%	57.14%	57.14%	×	☑	WideResNet-28-10	NeurIPS 2020
32	Helper-based Adversarial Training: Reducing Excessive Margin to Achieve a Better Accuracy vs. Robustness Trade-off *It uses additional 1M synthetic images in training.*	86.86%	57.09%	57.09%	×	×	PreActResNet-18	OpenReview, Jun 2021
33	LTD: Low Temperature Distillation for Robust Adversarial Training	85.21%	56.94%	56.94%	×	×	WideResNet-34-10	arXiv, Nov 2021
34	Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples *56.82% robust accuracy is due to the original evaluation (AutoAttack + MultiTargeted)*	85.64%	56.86%	56.82%	×	×	WideResNet-34-20	arXiv, Oct 2020
35	Fixing Data Augmentation to Improve Adversarial Robustness *It uses additional 1M synthetic images in training.*	83.53%	56.66%	56.66%	×	×	PreActResNet-18	arXiv, Mar 2021
36	Improving Adversarial Robustness Requires Revisiting Misclassified Examples	87.50%	56.29%	56.29%	×	☑	WideResNet-28-10	ICLR 2020
37	LAS-AT: Adversarial Training with Learnable Attack Strategy	84.98%	56.26%	56.26%	×	×	WideResNet-34-10	arXiv, Mar 2022

04

Conclusion

Conclusion

- **Learnable attack strategy:** we propose a novel adversarial training framework by introducing the concept of “learnable attack strategy”.
- **Two loss terms:** we also propose two loss terms that involve evaluating the robustness of the target network and predicting clean samples.
- **Superiority:** extensive experimental evaluations are performed on three benchmark databases to demonstrate the superiority of the proposed method.
- The code is released at *https://github.com/jiaxiaojunQAQ/LAS-AT* .

Boosting Fast Adversarial Training with Learnable Adversarial Initialization (Accepted by TIP)

Xiaojun Jia[1,2,†,*], Yong Zhang[3,*], Baoyuan Wu[4,5,‡], Jue Wang[3], Xiaochun Cao[1,2,‡]
1. Institute of Information Engineering, Chinese Academy of Sciences
2. School of Cyberspace Security, University of Chinese Academy of Sciences
3. Tencent, AI Lab 4. School of Data Science, The Chinese University of Hong Kong, Shenzhen
5. Secure Computing Lab of Big Data, Shenzhen Research Institute of Big Data, Shenzhen

目录

Content

Motivation

Catastrophic Overfitting

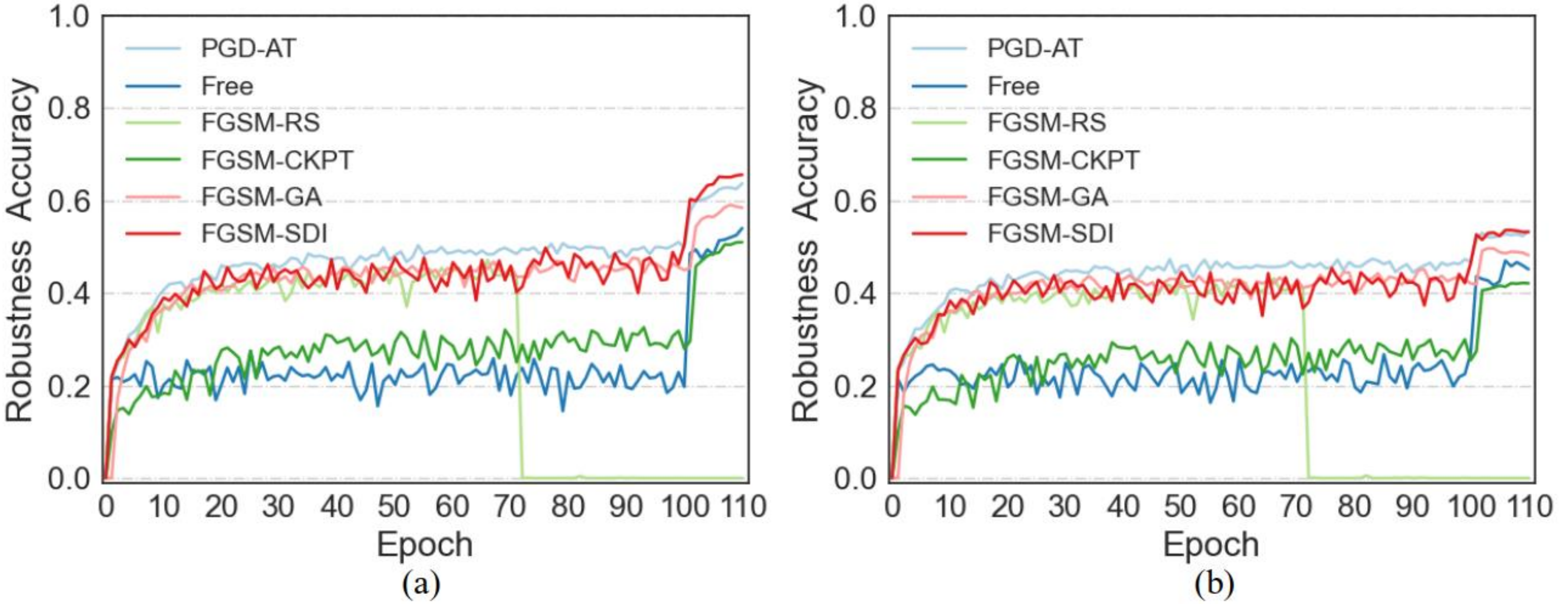


Fig. 5. The PGD-10 accuracy of AT methods on the CIFAR10 database in the training phase. (a) The PGD-10 accuracy on the training dataset. (b) The PGD-10 accuracy on the test dataset.

Motivation

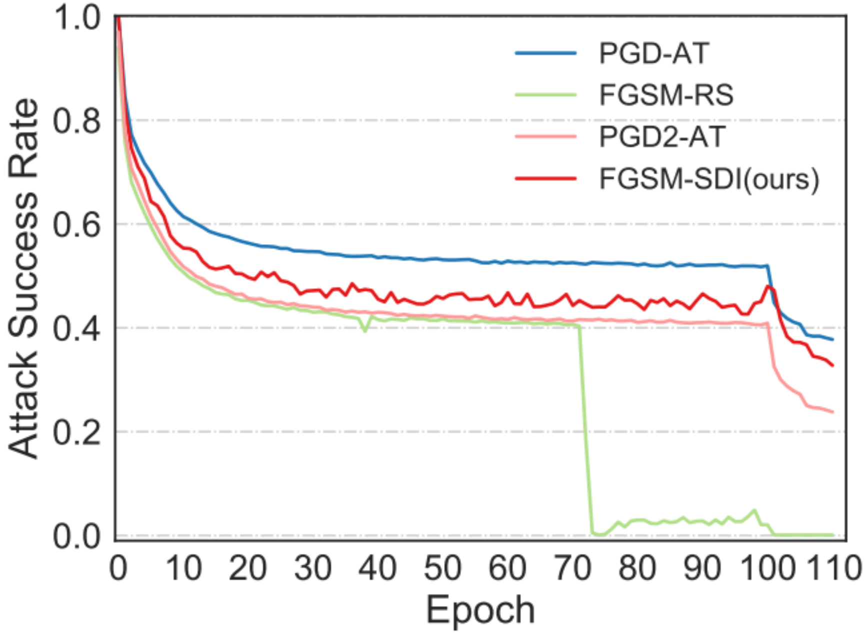


Fig. 6. Attack success rate of FGSM-RS, PGD-AT, PGD2-AT and FGSM-SDI(ours) during the training process.

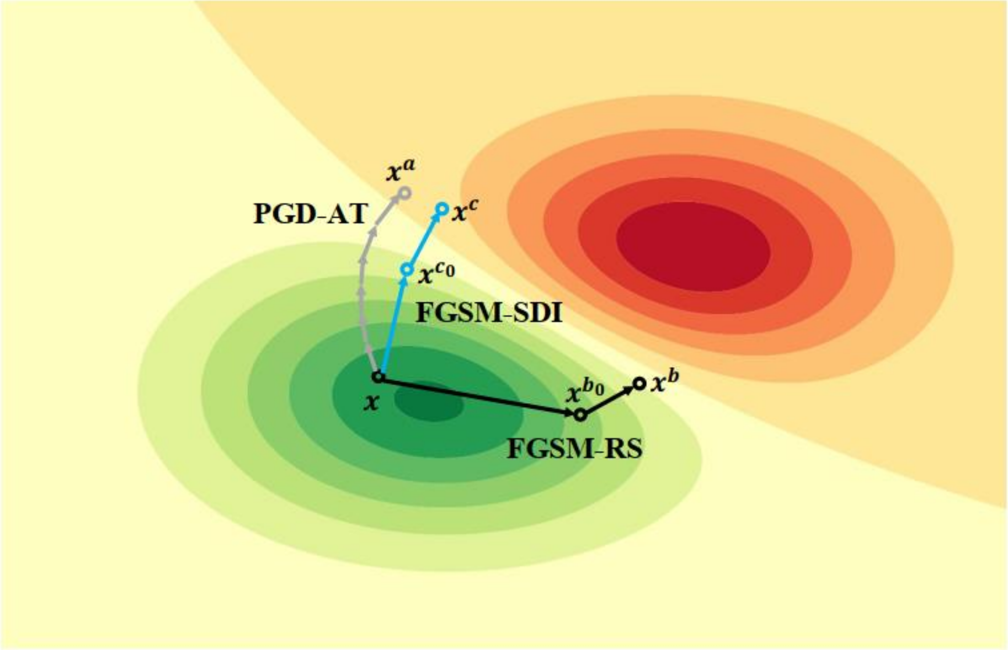


Fig. 1. Adversarial example generation process of PGD-AT [12], FGSM-RS [30], and our FGSM-SDI in the loss landscape of binary classification. Background is the contour of cross entropy. The redder the color, the lower the loss. PGD-AT is a multi-step AT method that computes gradients w.r.t the input at each step. FGSM-RS uses a random sample-agnostic initialization followed by FGSM, requiring the computation of gradient only once. But our FGSM-SDI uses a sample-dependent learnable initialization followed by FGSM.

Motivation

Our main contributions are as follows:

1. We propose a sample-dependent adversarial initialization method for fast AT. The sample-dependent property is achieved by a generative network trained with both benign examples and their gradient information from the target network, which outperforms other sample-agnostic fast AT methods. Our proposed adversarial initialization is dynamic and optimized by the generative network along with the adjusted robustness of the target network in the training phase, which further enhances adversarial robustness.
2. Extensive experiment results demonstrate that our proposed method not only shows a satisfactory training efficiency but also greatly boosts the robustness of fast AT methods. That is, it can achieve superiority over state-ofthe-art fast AT methods, as well as comparable robustness to advanced multi-step AT methods.

02

Method

Method

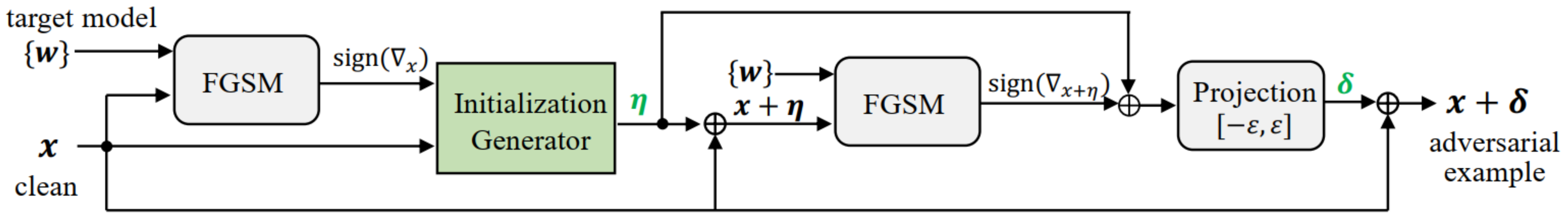


Fig. 2. Adversarial example generation of the proposed FGSM-SDI. The first FGSM is conducted on the clean image for the initialization generator to generate the initialization. The second FGSM is performed on the input image added with the generated initialization to generate adversarial examples. The two FGSM modules keep the same in the FGSM-SDI.

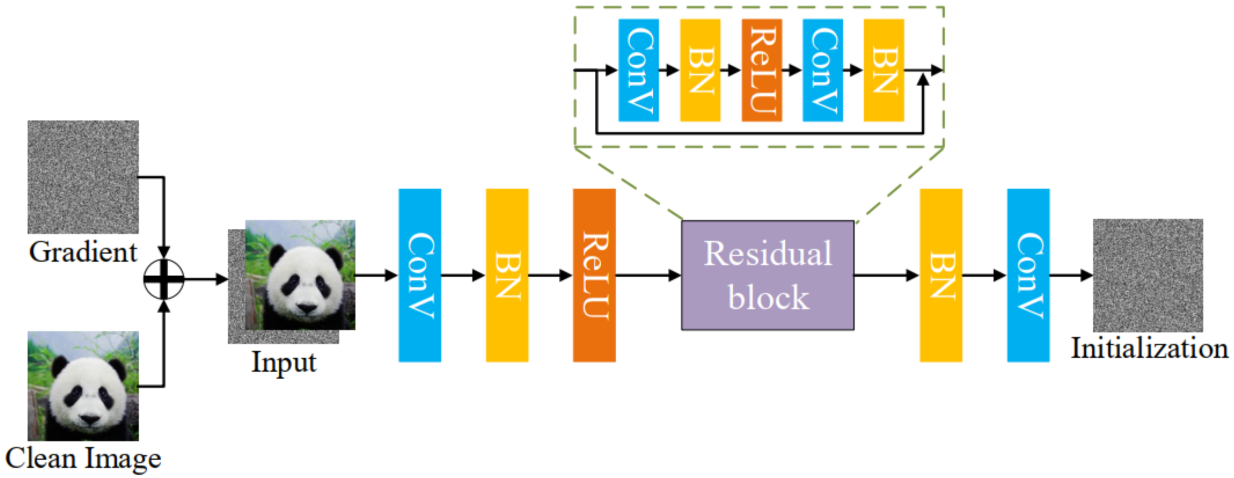


Fig. 3. The architecture of our lightweight generative network. The clean image combined with its gradient information from the target network forms the input of the generative network. The generative network consists of two convolutional layers and one ResBlock, which outputs the adversarial initialization for the clean image.

Method

Formulation of Adversarial Training:

$$\min_{\mathbf{w}} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\triangle} \mathcal{L}(f(x+\delta;\mathbf{w}),y) \right], \qquad (1)$$

Adversarial perturbation for PGD-AT:

$$\delta_{t+1} = \Pi_{[-\epsilon,\epsilon]^d}[\delta_t + \alpha \mathrm{sign}(\nabla_x \mathcal{L}(f(x+\delta_t;\mathbf{w}),y))], \qquad (2)$$

Adversarial perturbation for FGSM-AT:

$$\delta^* = \epsilon \mathrm{sign}(\nabla_x \mathcal{L}(f(x;\mathbf{w}),y)), \qquad (3)$$

Adversarial perturbation for FGSM-RS:

$$\delta^* = \Pi_{[-\epsilon,\epsilon]^d}[\eta + \alpha \mathrm{sign}(\nabla_x \mathcal{L}(f(x+\eta;\mathbf{w}),y))], \qquad (4)$$

Method

The signed gradient can be calculated as:

$$s_x = \mathrm{sign}(\nabla_x \mathcal{L}(f(x; \mathbf{w}), y)), \tag{5}$$

The initialization generation process can be defined as:

$$\eta_g = \epsilon g(x, s_x; \theta), \tag{6}$$

Adversarial perturbation for our proposed method:

$$\delta_g = \delta_g(\theta) = \Pi_{[-\epsilon,\epsilon]^d}[\eta_g + \alpha \mathrm{sign}(\nabla_x \mathcal{L}(f(x + \eta_g; \mathbf{w}), y))], \tag{7}$$

Method

Algorithm 3 FGSM-SDI (Ours)

Require: The epoch N, the maximal perturbation ϵ, the step size α, the dataset $\mathcal{D}$ including the benign sample x and the corresponding label y, the dataset size M, the target network $f(\cdot, \mathbf{w})$ with parameters $\mathbf{w}$, the generative network $g(\cdot, \theta)$ with parameters θ and the interval k.

for $n = 1, ..., N$ **do**
  for $i = 1, ..., M$ **do**
    $s_{x_i} = \text{sign}(\nabla_{x_i} \mathcal{L}(f(x_i; \mathbf{w}), y_i))$
    if $i \bmod \text{k} = 0$ **then**
      $\eta_g = \epsilon g(x_i, s_{x_i}; \theta)$
      $\delta = \Pi_{[-\epsilon,\epsilon]^d}[\eta_g + \alpha \text{sign}(\nabla_x \mathcal{L}(f(x_i + \eta_g; \mathbf{w}), y))]$
      $\theta \leftarrow \theta + \nabla_\theta \mathcal{L}(f(x_i + \delta; \theta), y_i)$
    end if
    $\eta_g = \epsilon g(x_i, s_{x_i}; \theta)$
    $\delta = \Pi_{[-\epsilon,\epsilon]^d}[\eta_g + \alpha \text{sign}(\nabla_x \mathcal{L}(f(x_i + \eta_g; \mathbf{w}), y))]$
    $\mathbf{w} \leftarrow \mathbf{w} - \nabla_{\mathbf{w}} \mathcal{L}(f(x_i + \delta; \mathbf{w}), y_i)$
  end for
end for

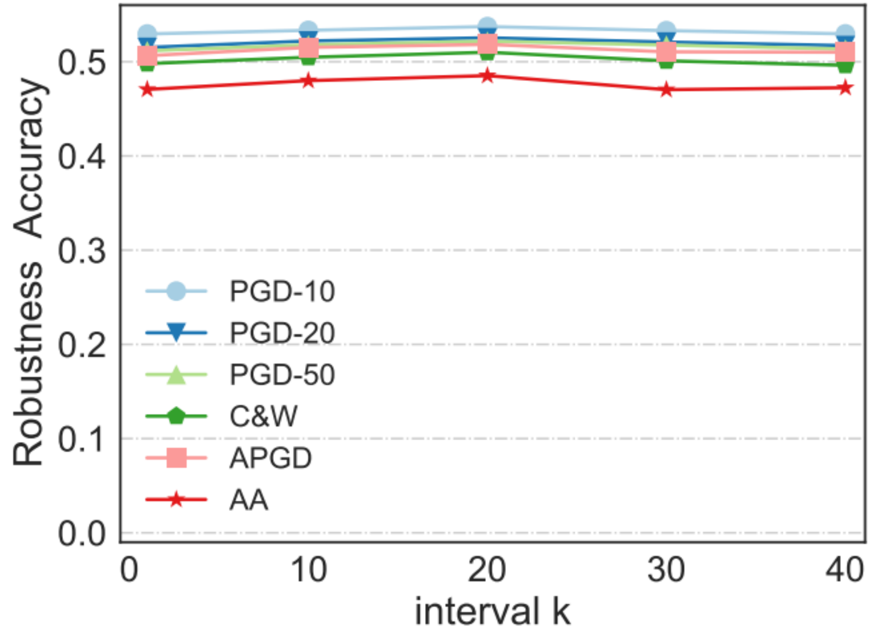


Fig. 4. Robustness accuracy of the proposed FGSM-SDI with different interval k. We adopt Resnet18 on the CIFAR10 database to conduct experiments

03

Experiments

Experiments

TABLE II
ABLATION STUDY OF THE INPUTS OF THE GENERATIVE NETWORK ON THE CIFAR10 DATABASE. NUMBERS IN TABLE REPRESENT PERCENTAGE. NUMBER IN BOLD INDICATES THE BEST.

Input		Clean	PGD-10	PGD-20	PGD-50	CW	AA
Benign	Best	73.34	42.63	41.82	41.66	42.31	36.72
	Last	**89.64**	21.34	13.72	7.59	4.04	0.00
Grad	Best	**86.08**	50.09	48.44	47.97	48.49	44.26
	Last	86.08	50.09	48.44	47.97	48.49	44.26
Benign+Grad	Best	84.86	**53.73**	**52.54**	**52.18**	**51.00**	**48.52**
	Last	85.25	**53.18**	**52.05**	**51.79**	**50.29**	**47.91**

Fig. 6. Attack success rate of FGSM-RS, PGD-AT, PGD2-AT and FGSM-SDI(ours) during the training process.

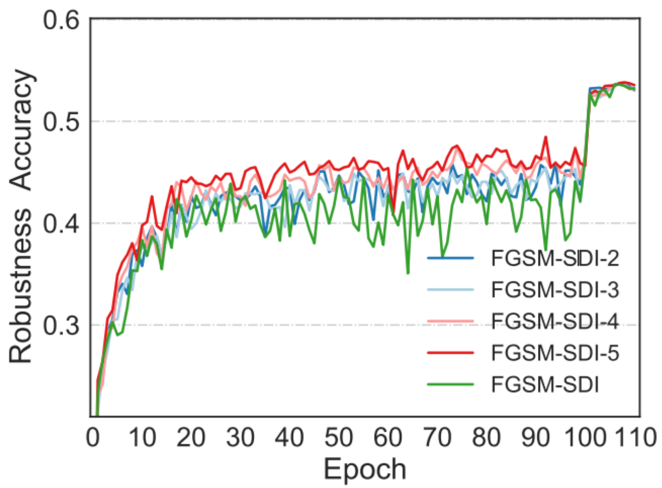


Fig. 7. The PGD-10 accuracy of FGSM-SDI with different m iterations of the generate network on the CIFAR10 database in the training phase.

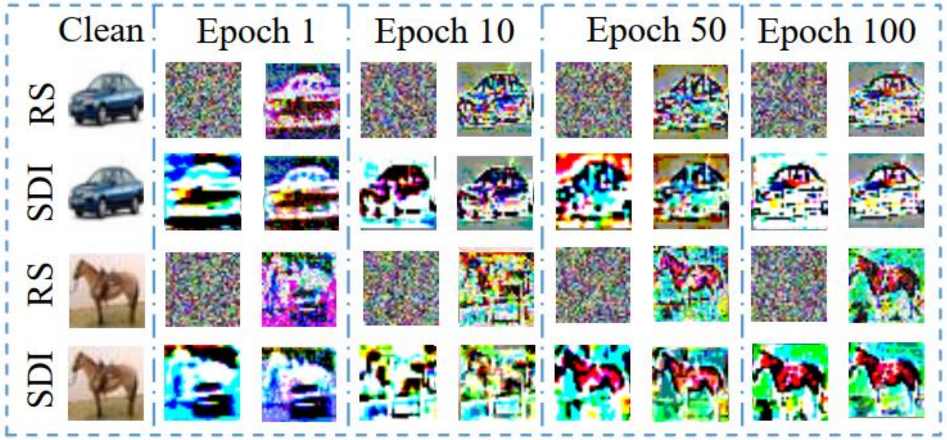


Fig. 8. Visualization of the adversarial initialization and FGSM-updated perturbations for the FGSM-RS and FGSM-SDI among continuous training epochs.

Experiments

TABLE III
COMPARISONS WITH PGD2-AT AND PGD4-AT ON CIFAR10 DATABASE. NUMBERS IN TABLE REPRESENT PERCENTAGE. NUMBER IN BOLD INDICATES THE BEST.

Method		Clean	PGD-10	PGD-20	PGD-50	C&W	APGD	AA	Time(min)
PGD2-AT	Best	**86.28**	49.28	47.51	47.01	47.73	46.56	44.47	77
	Last	**86.64**	48.49	47.05	46.46	47.31	45.98	44.14	
PGD4-AT	Best	86.15	49.44	48.08	47.56	48.11	47.22	45.11	119
	Last	86.61	48.94	47.27	46.88	47.82	46.63	44.60	
FGSM-SDI(ours)	Best	84.86	**53.73**	**52.54**	**52.18**	**51.00**	**51.84**	**48.50**	83
	Last	85.25	**53.18**	**52.05**	**51.79**	**50.29**	**51.30**	**47.91**	

TABLE IV
COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) WITH RESNET18 ON THE CIFAR10 DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS.

Target Network	Method		Clean	PGD-10	PGD-20	PGD-50	C&W	APGD	AA	Time(min)
Resnet18	PGD-AT	Best	82.32	53.76	52.83	52.6	51.08	52.29	48.68	265
		Last	82.65	53.39	52.52	52.27	51.28	51.90	48.93	
Resnet18	FGSM-RS	Best	73.81	42.31	41.55	41.26	39.84	41.02	37.07	51
		Last	83.82	00.09	00.04	00.02	0.00	0.00	0.00	
	FGSM-CKPT	Best	**90.29**	41.96	39.84	39.15	41.13	38.45	37.15	76
		Last	**90.29**	41.96	39.84	39.15	41.13	38.45	37.15	
	FGSM-GA	Best	83.96	49.23	47.57	46.89	47.46	45.86	43.45	178
		Last	84.43	48.67	46.66	46.08	46.75	45.05	42.63	
	Free-AT(m=8)	Best	80.38	47.1	45.85	45.62	44.42	42.18	42.17	215
		Last	80.75	45.82	44.82	44.48	43.73	45.22	41.17	
	FGSM-SDI(ours)	Best	84.86	**53.73**	**52.54**	**52.18**	**51.00**	**51.84**	**48.50**	83
		Last	85.25	**53.18**	**52.05**	**51.79**	**50.29**	**51.30**	**47.91**	

Experiments

TABLE V

COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) WITH WIDERESNET34-10 ON THE CIFAR10 DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS.

Target Network	Method	Clean	PGD-10	PGD-20	PGD-50	C&W	APGD	AA	Time(min)
WideResNet34-10	PGD-AT	85.17	56.1	55.07	54.87	53.84	54.15	51.67	1914
WideResNet34-10	FGSM-RS	74.29	41.24	40.21	39.98	39.27	39.79	36.40	348
	FGSM-CKPT	**91.84**	44.7	42.72	42.22	42.25	41.69	40.46	470
	FGSM-GA	81.8	48.2	47.97	46.6	46.87	46.27	45.19	1218
	Free-AT(m=8)	81.83	49.07	48.17	47.83	47.25	47.40	44.77	1422
	FGSM-SDI(ours)	86.4	**55.89**	**54.95**	**54.6**	**53.68**	**54.21**	**51.17**	533

TABLE VI

COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) ON THE CIFAR10 DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS. **ALL MODELS ARE TRAINED USING A CYCLIC LEARNING RATE STRATEGY.**

Target Network	Method		Clean	PGD-10	PGD-20	PGD-50	CW	APGD	AA	Time(min)
Resnet18	PGD-AT	Best	80.12	51.59	50.83	50.7	49.04	50.34	46.83	71
		Last	80.12	51.59	50.83	50.7	49.04	50.34	46.83	
Resnet18	FGSM-RS	Best	83.75	48.05	46.47	46.11	46.21	45.75	42.92	15
		Last	83.75	48.05	46.47	46.11	46.21	45.75	42.92	
	FGSM-CKPT	Best	**89.08**	40.47	38.2	37.69	39.87	37.16	35.81	21
		Last	**89.08**	40.47	38.2	37.69	39.87	37.16	35.81	
	FGSM-GA	Best	80.83	48.76	47.83	47.54	47.14	47.27	44.06	49
		Last	80.83	48.76	47.83	47.54	47.14	47.27	44.06	
	Free-AT(m=8)	Best	75.22	44.67	43.97	43.72	42.48	43.55	40.30	59
		Last	75.22	44.67	43.97	43.72	42.48	43.55	40.30	
	FGSM-SDI(ours)	Best	82.08	**51.63**	**50.65**	**50.33**	**48.57**	**49.98**	**46.21**	23
		Last	82.08	**51.63**	**50.65**	**50.33**	**48.57**	**49.98**	**46.21**	

Experiments

TABLE VII
COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) WITH RESNET18 ON THE CIFAR100 DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS.

Target Network	Method		Clean	PGD-10	PGD-20	PGD-50	C&W	APGD	AA	Time(min)
Resnet18	PGD-AT	Best	57.52	29.6	28.99	28.87	28.85	28.60	25.48	284
		Last	57.5	29.54	29.00	28.90	27.6	28.70	25.48	
Resnet18	FGSM-RS	Best	49.85	22.47	22.01	21.82	20.55	21.62	18.29	70
		Last	60.55	00.45	00.25	00.19	00.25	0.00	0.00	
	FGSM-CKPT	Best	**60.93**	16.58	15.47	15.19	16.4	14.63	14.17	96
		Last	**60.93**	16.69	15.61	15.24	16.6	14.87	14.34	
	FGSM-GA	Best	54.35	22.93	22.36	22.2	21.2	21.88	18.88	187
		Last	55.1	20.04	19.13	18.84	18.96	18.46	16.45	
	Free-AT(m=8)	Best	52.49	24.07	23.52	23.36	21.66	23.07	19.47	229
		Last	52.63	22.86	22.32	22.16	20.68	21.90	18.57	
	FGSM-SDI(ours)	Best	60.67	**31.5**	**30.89**	**30.6**	**27.15**	**30.26**	**25.23**	99
		Last	60.82	**30.87**	**30.34**	**30.08**	**27.3**	**29.94**	**25.19**	

TABLE VIII
COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) WITH PREACTRESNET18 ON THE TINY IMAGENET DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS.

Target Network	Method		Clean	PGD-10	PGD-20	PGD-50	CW	APGD	AA	Time(min)
PreActResNet18	PGD-AT	Best	43.6	20.2	19.9	19.86	17.5	19.64	16.00	1833
		Last	45.28	16.12	15.6	15.4	14.28	15.22	12.84	
PreActResNet18	FGSM-RS	Best	44.98	17.72	17.46	17.36	15.84	17.22	14.08	339
		Last	45.18	0.00	0.00	0.00	0.00	0.00	0.00	
	FGSM-CKPT	Best	**49.98**	9.20	9.20	8.68	9.24	8.50	8.10	464
		Last	**49.98**	9.20	9.20	8.68	9.24	8.50	8.10	
	FGSM-GA	Best	34.04	5.58	5.28	5.1	4.92	4.74	4.34	1054
		Last	34.04	5.58	5.28	5.1	4.92	4.74	4.34	
	Free-AT(m=8)	Best	38.9	11.62	11.24	11.02	11.00	10.88	9.28	1375
		Last	40.06	8.84	8.32	8.2	8.08	7.94	7.34	
	FGSM-SDI(ours)	Best	46.46	**23.22**	**22.84**	**22.76**	**18.54**	**22.56**	**17.00**	565
		Last	47.64	**19.84**	**19.36**	**19.16**	**16.02**	**19.08**	**14.10**	

Experiments

TABLE IX
COMPARISONS OF CLEAN AND ROBUST ACCURACY (%) AND TRAINING TIME (MINUTE) WITH RESNET50 ON THE IMAGENET DATABASE. NUMBER IN BOLD INDICATES THE BEST OF THE FAST AT METHODS.

ImageNet	Epsilon	Clean	PGD-10	PGD-50	Time(hour)
PGD-AT	$\epsilon=2$	64.81	47.99	47.98	211.2
	$\epsilon=4$	59.19	35.87	35.41	
	$\epsilon=8$	49.52	26.19	21.17	
Free-AT(m=4)	$\epsilon=2$	**68.37**	48.31	48.28	127.7
	$\epsilon=4$	63.42	33.22	33.08	
	$\epsilon=8$	52.09	19.46	12.92	
FGSM-RS	$\epsilon=2$	67.65	48.78	48.67	44.5
	$\epsilon=4$	**63.65**	35.01	32.66	
	$\epsilon=8$	**53.89**	0.00	0.00	
FGSM-SDI (ours)	$\epsilon=2$	66.01	**49.51**	**49.35**	66.8
	$\epsilon=4$	59.62	**37.5**	**36.63**	
	$\epsilon=8$	48.51	**26.64**	**21.61**	

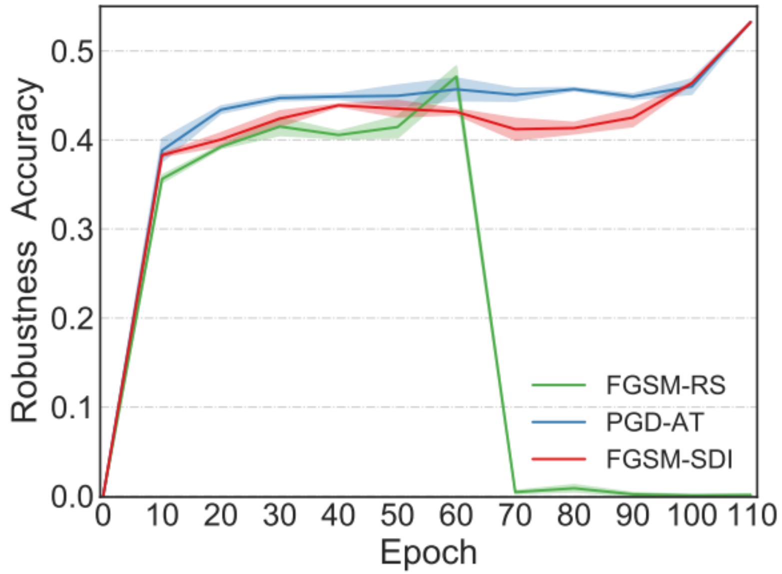


Fig. 11. The PGD-10 accuracy of FGSM-RS, PGD-AT and our FGSM-SDI with multiple training on the CIFAR10 database in the training phase.

Experiments

Fig. 9. The top row shows the clean images and the adversarial examples along with their corresponding heat-maps (generated by the Grad-CAM algorithm) on the FGSM-RS. The bottom row shows the results of our FGSM-SDI. Note that we adopt the same adversarial attack *i.e.,* PGD-10 , to conduct the visualization.

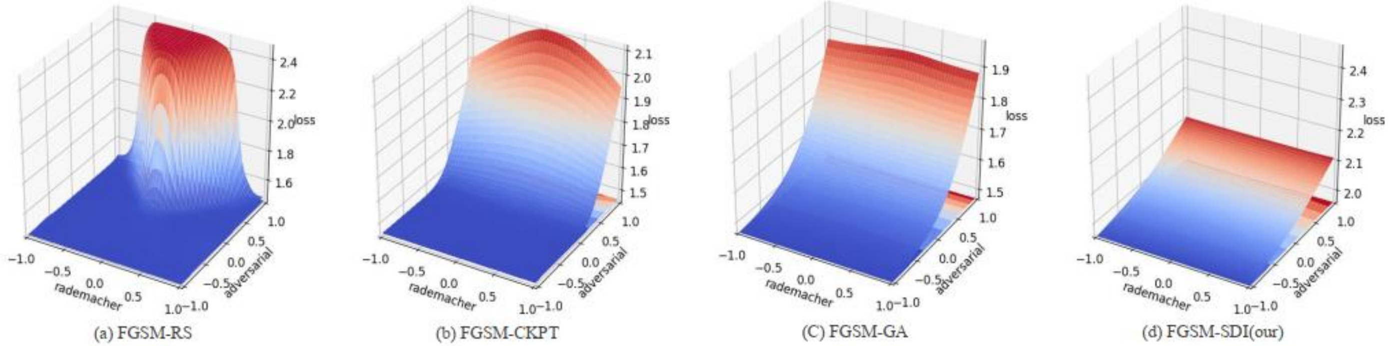


Fig. 10. Visualization of the loss landscape of on CIFAR10 for FGSM-RS, FGSM-CKPT, FGSM-GA, and our FGSM-SDI. We plot the cross entropy loss varying along the space consisting of two directions: an adversarial direction r_1 and a Rademacher (random) direction r_2. The adversarial direction can be defined as: $r_1 = \eta\,\mathrm{sign}(\nabla_x f(\hat{x}))$ and the Rademacher (random) direction can be defined as: $r_2 \sim \mathrm{Rademacher}(\eta)$, where η is set to $8/255$. Note that we adopt the same adversarial attack *i.e.,* PGD-10 , to conduct the visualization.

04

Conclusion

Conclusion

- **Adversarial Initialization:** we propose a sample-dependent adversarial initialization to boost fast AT.
- **Superiority:** extensive experimental evaluations are performed on three benchmark databases to demonstrate the superiority of the proposed method.
- The code is released at *https://github.com//jiaxiaojunQAQ//FGSM-SDI.*.

Prior-Guided Adversarial Initialization for Fast Adversarial Training

Xiaojun Jia[1,2,†], Yong Zhang[3,*], Xingxing Wei[4], Baoyuan Wu[5], Ke Ma[6], Jue Wang[3], Xiaochun Cao[1,7,*]

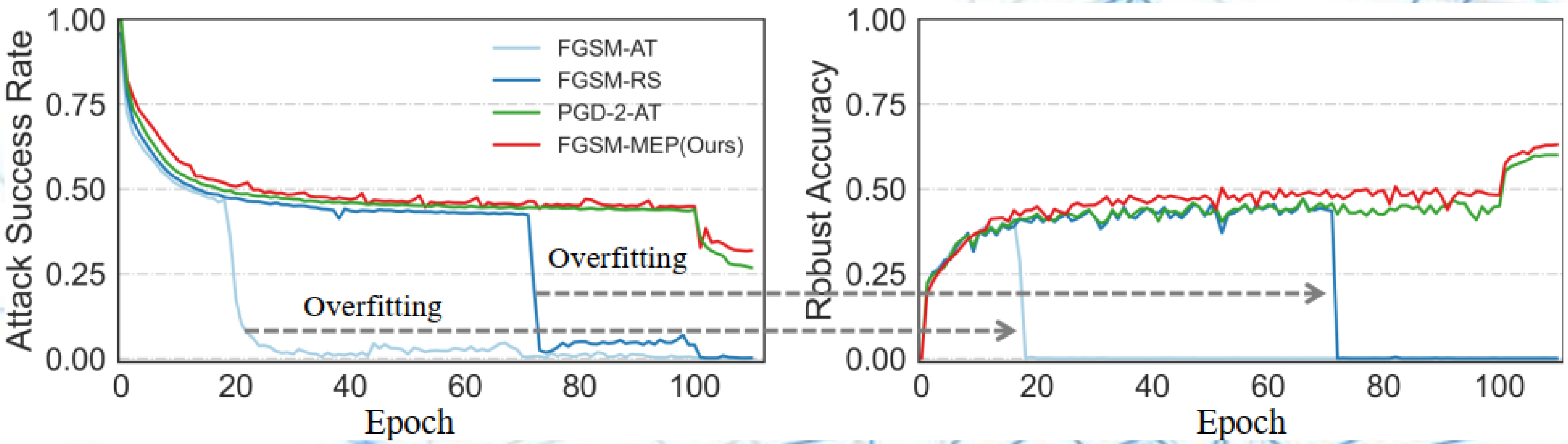
1. SKLOIS, Institute of Information Engineering, CAS 2. School of Cybers Security, University of Chinese Academy of Sciences 3. Tencent, AI Lab

4. Institute of Artificial Intelligence, Beihang University 5. School of Data Science, Secure Computing Lab of Big Data, Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong

6. School of Computer Science and Technology, University of Chinese Academy of Sciences 7. School of Cyber Science and Technology, Shenzhen Campus, Sun Yat-sen University, Shenzhen

$$\min_{\mathbf{w}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\max_{\boldsymbol{\delta}\in\Omega} \mathcal{L}(f_{\mathbf{w}}(\mathbf{x}+\boldsymbol{\delta}), y)]$$

Adversarial training is considered as one of the most effective defense methods to improve adversarial robustness through a minimax formulation.



Method

Prior From the Previous Batch (FGSM-BP):

$$\boldsymbol{\delta}_{B_{t+1}} = \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\delta}_{B_t} + \alpha \cdot \operatorname{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\delta}_{B_t};\mathbf{w}),\mathbf{y})\right)\right],$$

Prior From the Previous Epoch (FGSM-EP):

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

Prior From the Momentum of All Previous Epochs (FGSM-MEP):

$$\begin{aligned}
\mathbf{g}_c &= \operatorname{sign}\left(\nabla_{\mathbf{x}}\mathcal{L}(f(\mathbf{x}+\boldsymbol{\eta}_{E_t};\mathbf{w}),\mathbf{y})\right), \\
\mathbf{g}_{E_{t+1}} &= \mu \cdot \mathbf{g}_{E_t} + \mathbf{g}_c, \\
\boldsymbol{\delta}_{E_{t+1}} &= \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\eta}_{E_t} + \alpha \cdot \mathbf{g}_c\right], \\
\boldsymbol{\eta}_{E_{t+1}} &= \Pi_{[-\epsilon,\epsilon]}\left[\boldsymbol{\eta}_{E_t} + \alpha \cdot \operatorname{sign}(\mathbf{g}_{E_{t+1}})\right].
\end{aligned}$$

Method

The proposed regularization term can be added into the training loss to update the model parameters:

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} [\mathcal{L}(f(\mathbf{x} + \boldsymbol{\delta}_{adv}; \mathbf{w}), \mathbf{y}) + \lambda \cdot \|f(\mathbf{x} + \boldsymbol{\delta}_{adv}; \mathbf{w}) - f(\mathbf{x} + \boldsymbol{\delta}_{pgi}; \mathbf{w})\|_2^2],$$

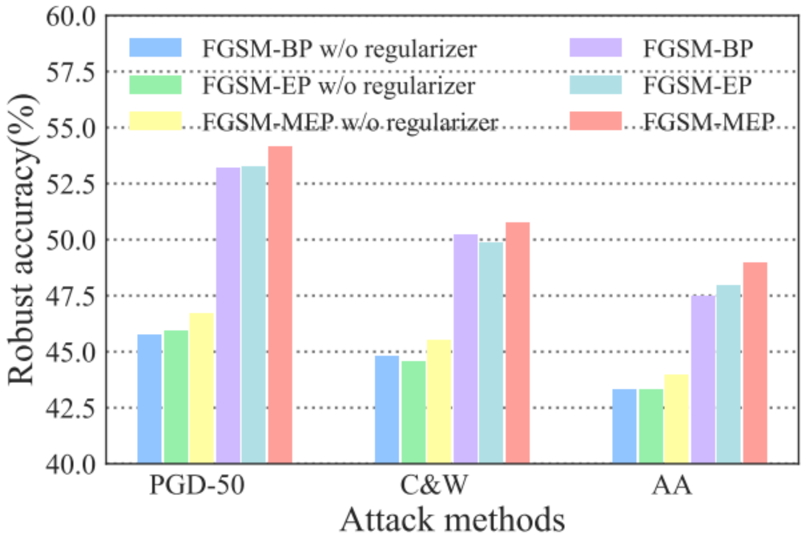


Table 1. Comparisons of clean and robust accuracy (%) and training time (minute) on the CIFAR-10 dataset. Number in bold indicates the best.

Method		Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time(min)
FGSM-BP	Best	**83.15**	54.59	53.55	53.2	50.24	47.47	73
	Last	**83.09**	54.52	53.5	53.33	50.12	47.17	
FGSM-EP	Best	82.75	54.8	53.62	53.27	49.86	47.94	73
	Last	81.27	55.07	54.04	53.63	50.12	46.83	
FGSM-MEP	Best	81.72	**55.18**	**54.36**	**54.17**	**50.75**	**49.00**	73
	Last	81.72	**55.18**	**54.36**	**54.17**	**50.75**	**49.00**	

Method

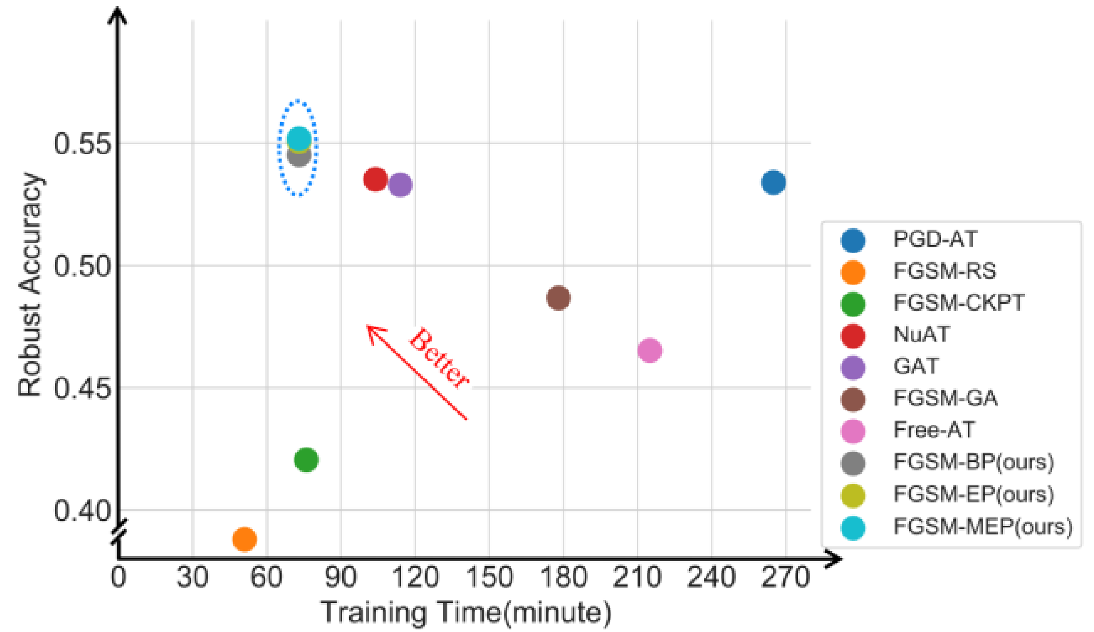
Algorithm 3 FGSM-MEP

Require: The epoch N, the maximal perturbation ϵ, the maximal label perturbation $\epsilon_{\mathbf{y}}$, the step size α, the dataset $\mathcal{D}$ including the benign sample $\mathbf{x}$ and the label $\mathbf{y}$, the dataset size M, the network $f(\cdot, \mathbf{w})$ with parameters $\mathbf{w}$, the decay factor μ, the hyper-parameter λ, the adversarial initialization set $\mathcal{D}^{\delta}$ and the historical model gradient $\mathcal{D}^{m}$.

for $n = 1, ..., N$ **do**
  for $i = 1, ..., M$ **do**
    if $n == 1$ **then**
      $\boldsymbol{\delta}_{pgi} = \mathbf{U}(-\epsilon, \epsilon)$
      $\mathbf{g}_c = \operatorname{sign}\left(\nabla_{\mathbf{x}_i} \mathcal{L}(f(\mathbf{x}_i + \boldsymbol{\delta}_{pgi}; \mathbf{w}), \mathbf{y}_i)\right)$
      $\mathcal{D}_i^m = \mathbf{g}_c$
      $\boxed{\boldsymbol{\delta}_{\mathbf{adv}} = \Pi_{[-\epsilon,\epsilon]}[\boldsymbol{\delta}_{pgi} + \alpha \cdot \mathbf{g}_c]}$ → **对抗初始化**
      $\mathcal{D}_i^{\delta} = \boldsymbol{\delta}_{adv}$
      $\mathbf{w} \leftarrow \mathbf{w} - \nabla_{\mathbf{w}}[\mathcal{L}(f(\mathbf{x}_i + \boldsymbol{\delta}_{adv}; \mathbf{w}), \mathbf{y}_i) + \boxed{\lambda \cdot \| f(\mathbf{x} + \boldsymbol{\delta}_{adv}; \mathbf{w}) - f(\mathbf{x} + \boldsymbol{\delta}_{pgi}; \mathbf{w}) \|_2^2]}$

模型正则

    else
      $\boldsymbol{\delta}_{pgi} = \mathcal{D}_i^{\delta}$
      $\mathbf{g}_c = \operatorname{sign}\left(\nabla_{\mathbf{x}_i} \mathcal{L}(f(\mathbf{x}_i + \boldsymbol{\delta}_{pgi}; \mathbf{w}), \mathbf{y}_i)\right)$
      $\mathcal{D}_i^m = \mu \cdot \mathcal{D}_i^m + \mathbf{g}_c$
      $\boldsymbol{\delta}_{adv} = \Pi_{[-\epsilon,\epsilon]}[\boldsymbol{\delta}_{pgi} + \alpha \cdot \mathbf{g}_c]$
      $\mathcal{D}_i^{\delta} = \Pi_{[-\epsilon,\epsilon]}[\boldsymbol{\delta}_{pgi} + \alpha \cdot \operatorname{sign}(\mathcal{D}_i^m)]$
      $\mathbf{w} \leftarrow \mathbf{w} - \nabla_{\mathbf{w}}[\mathcal{L}(f(\mathbf{x}_i + \boldsymbol{\delta}_{adv}; \mathbf{w}), \mathbf{y}_i) + \lambda \cdot \| f(\mathbf{x} + \boldsymbol{\delta}_{adv}; \mathbf{w}) - f(\mathbf{x} + \boldsymbol{\delta}_{pgi}; \mathbf{w}) \|_2^2]$
    end if
  end for
end for



Convergence Analysis

Proposition 1. *Let $\boldsymbol{\delta}_{pgi}$ be the prior-guided adversarial initialization in **FGSM-BP**, **FSGM-EP** or **FSGM-MEP**, $\hat{\boldsymbol{\delta}}_{adv}$ represents the current adversarial perturbation generated via FGSM using $\boldsymbol{\delta}_{pgi}$ as initialization, and α be the step size of (5), (6), (9) and (10). If $\boldsymbol{\Omega}$ is a bounded set like*

$$\boldsymbol{\Omega} = \left\{\hat{\boldsymbol{\delta}}_{adv} \ : \ \|\hat{\boldsymbol{\delta}}_{adv} - \boldsymbol{\delta}_{pgi}\|_2^2 \leq \epsilon^2\right\}, \tag{12}$$

and the step size α satisfies $\alpha \leq \epsilon$, it holds that

$$\begin{aligned} \mathbb{E}_{\hat{\boldsymbol{\delta}}_{adv}\sim\boldsymbol{\Omega}}\left[\|\hat{\boldsymbol{\delta}}_{adv}\|_2\right] &\leq \sqrt{\mathbb{E}_{\hat{\boldsymbol{\delta}}_{adv}\sim\boldsymbol{\Omega}}\left[\|\hat{\boldsymbol{\delta}}_{adv}\|_2^2\right]} \\ &\leq \sqrt{\frac{1}{d}} \cdot \epsilon, \end{aligned} \tag{13}$$

*where $\hat{\boldsymbol{\delta}}_{adv}$ is the adversarial perturbation generated by **FGSM-BP**, **FSGM-EP** or **FSGM-MEP**, and d is the dimension of the feature space.*

The proof is deferred to the **supplementary material**. The upper bound of the proposed method is $\sqrt{\frac{1}{d}} \cdot \epsilon$ which is less than the bound $\sqrt{\frac{d}{3}} \cdot \epsilon$ of FGSM-RS provided in [2]. Due to the norm of perturbation (gradient) can be treated as the convergence criteria for the non-convex optimization problem, the smaller expectation represents that the proposed prior-guided adversarial initialization will be converged to a local optimal faster than the random initialization with the same number of iterations.

Experiments

Comparisons on CIFAR-10

Method		Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time(min)
PGD-AT [37]	Best	82.32	53.76	52.83	52.6	51.08	48.68	265
	Last	82.65	53.39	52.52	52.27	51.28	48.93	
FGSM-RS [49]	Best	73.81	42.31	41.55	41.26	39.84	37.07	51
	Last	83.82	00.09	00.04	00.02	0.00	0.00	
FGSM-CKPT [25]	Best	**90.29**	41.96	39.84	39.15	41.13	37.15	76
	Last	**90.29**	41.96	39.84	39.15	41.13	37.15	
NuAT [42]	Best	81.58	53.96	52.9	52.61	**51.3**	**49.09**	104
	Last	81.38	53.52	52.65	52.48	50.63	48.70	
GAT [41]	Best	79.79	54.18	53.55	53.42	49.04	47.53	114
	Last	80.41	53.29	52.06	51.76	49.07	46.56	
FGSM-GA [2]	Best	83.96	49.23	47.57	46.89	47.46	43.45	178
	Last	84.43	48.67	46.66	46.08	46.75	42.63	
Free-AT(m=8) [39]	Best	80.38	47.1	45.85	45.62	44.42	42.17	215
	Last	80.75	45.82	44.82	44.48	43.73	41.17	
FGSM-BP (ours)	Best	83.15	54.59	53.55	53.2	50.24	47.47	73
	Last	83.09	54.52	53.5	53.33	50.12	47.17	
FGSM-EP (ours)	Best	82.75	54.8	53.62	53.27	49.86	47.94	73
	Last	81.27	55.07	54.04	53.63	50.12	46.83	
FGSM-MEP (ours)	Best	81.72	**55.18**	**54.36**	**54.17**	50.75	49.00	73
	Last	81.72	**55.18**	**54.36**	**54.17**	**50.75**	**49.00**	

Comparisons on CIFAR-100

Method		Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time(min)
PGD-AT [37]	Best	57.52	29.6	28.99	28.87	28.85	25.48	284
	Last	57.5	29.54	29.00	28.90	27.6	25.48	
FGSM-RS [49]	Best	49.85	22.47	22.01	21.82	20.55	18.29	70
	Last	60.55	00.45	00.25	00.19	00.25	0.00	
FGSM-CKPT [25]	Best	**60.93**	16.58	15.47	15.19	16.4	14.17	96
	Last	**60.93**	16.69	15.61	15.24	16.6	14.34	
NuAT [41]	Best	59.71	27.54	23.02	20.18	22.07	11.32	115
	Last	59.62	27.07	22.72	20.09	21.59	11.55	
GAT [42]	Best	57.01	24.55	23.8	23.55	22.02	19.60	119
	Last	56.07	23.92	23.18	23.0	21.93	19.51	
FGSM-GA [2]	Best	54.35	22.93	22.36	22.2	21.2	18.88	187
	Last	55.1	20.04	19.13	18.84	18.96	16.45	
Free-AT(m=8) [39]	Best	52.49	24.07	23.52	23.36	21.66	19.47	229
	Last	52.63	22.86	22.32	22.16	20.68	18.57	
FGSM-BP (ours)	Best	57.58	30.78	30.01	28.99	26.40	23.63	83
	Last	83.82	30.56	29.96	28.82	26.32	23.43	
FGSM-EP (ours)	Best	57.74	31.01	30.17	29.93	27.37	24.39	83
	Last	57.74	31.01	30.17	29.93	27.37	24.39	
FGSM-MEP (ours)	Best	58.78	**31.88**	**31.26**	**31.14**	**28.06**	**25.67**	83
	Last	58.81	**31.6**	**31.03**	**30.88**	**27.72**	**25.42**	

Experiments

Comparisons on Tiny ImageNet

Method		Clean	PGD-10	PGD-20	PGD-50	C&W	AA	Time(min)
PGD-AT [37]	Best	43.6	20.2	19.9	19.86	17.5	16.00	1833
	Last	45.28	16.12	15.6	15.4	14.28	12.84	
FGSM-RS [49]	Best	44.98	17.72	17.46	17.36	15.84	14.08	339
	Last	45.18	0.00	0.00	0.00	0.00	0.00	
FGSM-CKPT [25]	Best	**49.98**	9.20	9.20	8.68	9.24	8.10	464
	Last	**49.98**	9.20	9.20	8.68	9.24	8.10	
NuAT [42]	Best	42.9	15.12	14.6	14.44	12.02	10.28	660
	Last	42.42	13.78	13.34	13.2	11.32	9.56	
GAT [41]	Best	42.16	15.02	14.5	14.44	11.78	10.26	663
	Last	41.84	14.44	13.98	13.8	11.48	9.74	
FGSM-GA [2]	Best	43.44	18.86	18.44	18.36	16.2	14.28	1054
	Last	43.44	18.86	18.44	18.36	16.2	14.28	
Free-AT(m=8) [39]	Best	38.9	11.62	11.24	11.02	11.00	9.28	1375
	Last	40.06	8.84	8.32	8.2	8.08	7.34	
FGSM-BP (ours)	Best	45.01	21.67	21.47	21.43	17.89	15.36	458
	Last	47.16	20.62	20.16	20.07	15.68	14.15	
FGSM-EP (ours)	Best	45.01	21.67	21.47	21.43	17.89	15.36	458
	Last	46.00	20.77	20.39	20.28	16.65	14.93	
FGSM-MEP (ours)	Best	43.32	**23.8**	**23.4**	**23.38**	**19.28**	**17.56**	458
	Last	45.88	**22.02**	**21.7**	**21.6**	**17.44**	**15.50**	

Comparisons on ImageNet

ImageNet	Epsilon	Clean	PGD-10	PGD-50	Time (hour)
Free-AT(m=4)[39]	$\epsilon=2$	68.37	48.31	48.28	127.7
	$\epsilon=4$	63.42	33.22	33.08	
	$\epsilon=8$	52.09	19.46	12.92	
FGSM-RS [49]	$\epsilon=2$	67.65	48.78	48.67	44.5
	$\epsilon=4$	63.65	35.01	32.66	
	$\epsilon=8$	53.89	0.00	0.00	
FGSM-BP (ours)	$\epsilon=2$	**68.41**	**49.11**	**49.10**	63.7
	$\epsilon=4$	**64.32**	**36.24**	**34.93**	
	$\epsilon=8$	**53.96**	**21.76**	**14.33**	

Our main contributions are in three aspects:

1) We propose a prior-guided adversarial initialization to prevent overfitting after investigating several initialization strategies.

2) We also propose a regularizer to guide the model learning for better robustness by considering both the currently generated perturbation and the prior-guided initialization.

3) Extensive experiments on four datasets demonstrate that the proposed method can outperform state-of-the-art FAT methods in terms of both efficiency and robustness.