Defense against Adversarial Attacks Using High-Level Representation Guided Denoiser
SUPPLEMENTARY MATERIALS

Fangzhou Liao*, Ming Liang*, Yinpeng Dong, Tianyu Pang, Xiaolin Hu†, Jun Zhu
Department of Computer Science and Technology, Tsinghua Lab of Brain and Intelligence
Beijing National Research Center for Information Science and Technology, BNRist Lab
Tsinghua University, 100084 China
{liaofangzhou, liangming.tsinghua}@gmail.com, {dyp17, pty17}@mails.tsinghua.edu.cn, {xlhu, dcszj}@tsinghua.edu.cn

More results and further analysis of the proposed method are presented here.

1. Robustness to ensemble black box attacks

To evaluate the robustness of our method to powerful ensemble black box attack, we gather 5 models (Vgg16[3], Resnet-v1-152[1], IncV2[5], IncV4, ensIncResV2[6], and use FGSM and IFGSM8 to generate adversarial examples by attacking them. Then we test LGD and ensV3 on this new dataset, the result is shown in Table S1. It turns out that LGD still outperforms ensV3 by a large margin.

Table S1. The accuracy of different methods against blackbox ensemeble attack.

Defense	FGSM,ϵ4	FGSM,ϵ16	IFGSM8,ϵ4	IFGSM8,ϵ16
NA	54.5%	33.6%	53.0%	22.8%
ensV3	66.7%	50.2%	67.6%	59.3%
LGD	**72.8%**	**70.5%**	**69.2%**	**59.4%**

2. Combination of HGD and adversarially trained model

As two different approaches, denoising networks and adversarial training may have complementary effects, use a HGD to process the distorted images before they are inputted to an adversarially trained model may further improve the performance. To test this idea, we train an LGD with ensemble adversarially trained Inception V3 (ensV3) [6] as the target model. For fair comparison, we replace the attack methods targeting at IncV3 (and other models) with attack methods targeting at ensV3 (and other models) and make a new dataset correspondingly. The LGD+ensV3 model is trained and tested on this new dataset.

*Equal contribution.
†Corresponding author.

Table S2. The influence of adding a LGD before ensV3. For fair comparison, the white-box attacks (WhiteTestSet) used in the two blocks are targeting at IncV3 and ensV3 respectively.

Defense	Clean	WhiteTestSet		BlackTestSet	
		$\epsilon = 4$	$\epsilon = 16$	$\epsilon = 4$	$\epsilon = 16$
IncV3	76.7%	14.5%	14.4%	61.2%	41.0%
LGD + IncV3	76.2%	**75.2%**	69.2%	**75.1%**	**72.2%**
ensV3	**76.9%**	71.4%	61.7%	72.4%	62.0%
LGD + ensV3	**76.9%**	75.0%	**72.5%**	74.7%	72.1%

Because ensV3 is more robust than IncV3, we expect to see higher robustness in LGD+ensV3. However, the results show that although LGD helps the ensv3 to improve the robustness, their combination is not significantly better than the LGD+IncV3 (Table S2) (The results of LGD+IncV3 are copied from Table 3.).

3. Ensembles of HGD protected models

Ensemble is an effective method to boost the performance of classifiers. We explore the ensemble of several HGDs and target models. Specifically, we train two LGDs. The first one is denoted by LGD1, which is trained with IncV3 as the target model and the second one is denoted by LGD2, which is trained with ensV3 as the target model[1]. Different combinations of the two denoisers and various target models are tested:

- **IncV3&ensV3**. The adversarial images (x^*) are fed directly to the two models without protection, and the output logits of IncV3 and ensV3 are averaged as the result. The symbol & indicate an ensemble.

  Motivation: This method serves as the baseline ensemble.

- **LGD1$\rightarrow$ IncV3&ensV3**. x^* is firstly fed to the LGD1, resulting in a denoised image $\hat{x}$, which is then fed to the ensemble of IncV3&ensV3. The $\rightarrow$ indicates the flow of data.

  Motivation: LGD shows certain transferability across different models (Section 5.3 in main paper). So it is possible to use an LGD to protect multiple models.

- **(LGD1$\rightarrow$IncV3&ensV3)&(LGD2$\rightarrow$IncV3&ensV3)**. LGD1 and LGD2 give two denoised images $\hat{x}_1$ and $\hat{x}_2$, which are then fed to IncV3&ensV3 independently. The four output logits are averaged as the result.

  Motivation: This method is similar with the last one, but make use of LGD2.

- **LGD1&LGD2$\rightarrow$IncV3&ensV3**. The output of LGD1 and LGD2 are averaged ($\hat{x} = (\hat{x}_1 + \hat{x}_2)/2$). The averaged denoised image is then fed to IncV3&ensV3.

  Motivation: Each LGD give an independent estimation of the adversarial perturbation, so averaging the outputs of two LGD may result in a better estimation of perturbation.

[1]Different from Section 2, the LGD2 used in this section is trained and evaluated on the default dataset.

Table S3. The classification accuracy on the test set obtained by different methods.

Defense	Clean	WhiteTestSet		BlackTestSet	
		$\epsilon = 4$	$\epsilon = 16$	$\epsilon = 4$	$\epsilon = 16$
LGD1→IncV3	76.2%	75.2%	69.2%	75.1%	72.2%
LGD2→ensV3	76.9%	74.4%	71.8%	75.2%	73.8%
IncV3&ensV3	**78.8%**	35.6%	30.0%	70.2%	56.6%
LGD1→IncV3&ensV3	77.6%	75.6%	71.3%	75.5%	72.7%
(LGD1→IncV3&ensV3)&(LGD2→IncV3&ensV3)	78.5%	69.9%	67.8%	77.0%	74.7%
LGD1&LGD2→IncV3&ensV3	78.2%	70.6%	67.6%	76.5%	74.5%
(LGD1→IncV3)&(LGD2→ensV3)	78.6%	**77.4%**	**73.4%**	**77.7%**	**75.7%**

Table S4. The classification accuracy on test sets obtained by different defenses. NA means no defense.

Defense	Clean	WhiteTestSet		BlackTestSet	
		$\epsilon = 4$	$\epsilon = 16$	$\epsilon = 4$	$\epsilon = 16$
NA	**76.7%**	14.5%	14.4%	61.2%	41.0%
PGD	75.3%	20.0%	13.8%	67.5%	55.7%
PGD x2	73.7%	40.7%	50.8%	70.9%	67.4%
LGD	76.2%	**75.2%**	**69.2%**	**75.1%**	**72.2%**

- **(LGD1→IncV3)&(LGD2→ensV3)**. $\hat{x}_1$ and $\hat{x}_2$ are fed to IncV3 and ensV3 respectively. The logits of the two models are averaged as result.

  Motivation: The most straightforward way of ensemble.

Except for these ensembles, two single model baselines **LGD1→IncV3** and **LGD2→ensV3** are also tested.

The results of these methods are shown in Table S3. (LGD1→IncV3)&(LGD2→ensV3) performs the best and shows consistent improvement comparing to baselines. Other ensemble methods achieve little improvements comparing with the single models. Some of them even have degraded performance.

4. Remedy for the low slope of PGD

In Section 5.4 of the paper, we show the different denoising properties of PGD and LGD ($d\hat{x} = kdx^*$), and it may be inferred that the low k value of PGD is an important factor for PGD's poor performance. To validate this assumption, we replace the output of PGD with $\hat{x} = x^* - 2d\hat{x}$ (i.e. replace $-d\hat{x}$ in Fig. 2 by $-2d\hat{x}$), so that its k is close to 1. The results (denoted by PGD x2 in Table S4) are significantly higher than those of the original PGD but still not as high as those of LGD. Besides, this change also significantly decreases the accuracy on clean images. Therefore the low k value of PGD is indeed a reason for its poor robustness, but it cannot explain all difference between PGD and LGD.

5. The details of NIPS 2017 solution

In this section, we list the attacks we used to generate the training and validation sets for training the denoiser. ϵ is the L_∞ constraint of the adversarial perturbation. In our experiment, $\epsilon = 16$ is fixed.

- **No operation.** The clean images are directly adopted.

- **Images withs random noise.** Each pixel of an original image is randomly perturbed with ϵ or $-\epsilon$ with equal probability.
- **FGSM x IncV3.** The word before "x" indicates the attacking method, and the word after it indicates the target model. FGSM meas fast gradient sign method (defined in the main paper).
- **FGSM x ensV3&advV3.** advV3 is an adversarially trained Inception V3 model [2].
- **FGSM x IncV3&IncV4&ensIncResV2.** IncV4 is Inception V4 [4]. EnsIncResV2 is an ensemble adversarially trained InceptionResnet V2 [4, 6].
- **IFGSM^2 x 7 models.** IFGSM^k means k step iterative FGSM. The step size of each step is ϵ/k. 7 models means the ensemble of IncV3, advV3, ensV3, ens4V3[6], IncV4, IncResV2 and ensIncResV2. ens4V3 is another ensemble adversarially trained Inception V3 [6]. IncResV2 is the InceptionResnet V2 [4].
- **IFGSM^4 x 7 models.**
- **IFGSM^8 x 7 models.**
- **IFGSM^2 x 8 models.** 8 models means the ensemble of the 7 models mentioned above and Resnet101[1].
- **IFGSM^8 x 8 models.**
- **dIFGSM^4 x 8 models.** dIFGSM means IFGSM with decayed step size, which is set as 0.4ϵ, 0.3ϵ, 0.2ϵ, 0.1ϵ for dIFGSM^4.
- **adaIFGSM^1 x 8 models.** adaIFGSM means adaptive IFGSM. In adaIFGSM, there is a list of IFGSM attacks with increasing power and running time (i.e. more iterations). An original image is firstly perturbed by the simplest attack. If this adversarial image successfully fools all the target models, it is used as output and the procedure ends, else the original image would be perturbed by the attack in the next level. This procedure continues until the adversarial image fools all the target models or the last attack in the list is used.

  In adaIFGSM^1, the attack list is (dIFGSM^4 x 8 models, IFGSM^{20} x 8 models).
- **adaIFGSM^2 x 8 models.** In adaIFGSM^2, the attack list is (dIFGSM^3 x 8 models, dIFGSM^4 x 8 models, IFGSM^{20} x 8 models). The step size is set as $\frac{1}{2}\epsilon$, $\frac{1}{3}\epsilon$, $\frac{1}{6}\epsilon$ for dIFGSM^3.
- **adaIFGSM^3 x 8 models.** In adaIFGSM^3, the attack list is (FGSM x 8 models, IFGSM^2 x 8 models, IFGSM^4 x 8 models, IFGSM^8 x 8 modelS, IFGSM^{20} x 8 models).

These attacks are applied to 15,000 original images, resulting in a training set of 210,000 adversarial images. And they are applied to other 5,000 original images, resulting in a validation set of 70,000 adversarial images.

6. Detailed performance on the test set

Here we provide the detail version of tables in the main text, where the performance of each defense against each attack is shown. The name of attacks is abbreviated in the following way:

- F: FGSM,
- I4: IFGSM^4,
- v3: IncV3,
- v4: IncV4,
- ens: ensemble of IncV3, Resnetv2, IncResnetv2.
- e4: $\epsilon = 4$,
- e16: $\epsilon = 16$.

For example, F_v3_e4 denotes the attack FGSM x IncV3, $\epsilon = 4$. From these results we can conclude that HGD is better than PGD and ensV3 consistently, no matter the attack is black-box or white-box, FGSM or IFGSM.

Table S5. The details of Table 2.

Defense	Clean	F_v3_e4	I4_ens_e4	F_v3_e16	I4_ens_e16	F_v4_e4	I4_v4_e4	F_v4_e16	I4_v4_e16
Naive	0.0000	0.0209	0.0146	0.0613	0.0260	0.0209	0.0143	0.0613	0.0289
DAE	0.0360	0.0358	0.0360	0.0367	0.0354	0.0359	0.0361	0.0372	0.0367
DUNET	0.0157	0.0183	0.0170	0.0205	0.0219	0.0184	0.0169	0.0216	0.0237
DAUNET	0.0150	0.0142	0.0138	0.0159	0.0168	0.0143	0.0136	0.0162	0.0201
Naive	76.7%	21.6%	7.4%	23.9%	4.8%	60.3%	62.1%	47.4%	34.7%
DAE	58.3%	51.0%	51.8%	37.6%	35.8%	55.5%	56.3%	46.8%	50.8%
DUNET	76.2%	23.2%	8.9%	24.5%	5.7%	63.2%	65.7%	58.9%	50.5%
DAUNET	75.3%	31.0%	8.9%	21.8%	5.9%	68.3%	66.7%	60.0%	51.4%

Table S6. The details of Table 3.

Defense	Clean	F_v3_e4	I4_ens_e4	F_v3_e16	I4_ens_e16	F_v4_e4	I4_v4_e4	F_v4_e16	I4_v4_e16
Naive	76.7%	21.6%	7.4%	23.9%	4.8%	60.3%	62.1%	47.4%	34.7%
PGD	75.3%	31.0%	8.9%	21.8%	5.9%	68.3%	66.7%	60.0%	51.4%
ensV3	76.9%	71.7%	68.0%	58.3%	57.7%	71.5%	73.4%	57.3%	66.6%
FGD	76.1%	75.2%	72.2%	71.6%	63.2%	74.4%	74.3%	73.9%	69.8%
LGD	76.2%	76.0%	74.5%	74.2%	64.1%	75.2%	74.9%	74.0%	70.4%
CGD	74.9%	76.2%	75.5%	75.8%	70.5%	74.8%	74.2%	73.0%	69.2%

Table S7. The details of Table 4.

Denoiser for Resnet	Clean	F_v3_e4	I4_ens_e4	F_v3_e16	I4_ens_e16	F_v4_e4	I4_v4_e4	F_v4_e16	I4_v4_e16
NA	78.5%	68.5%	58.2%	51.4%	25.3%	66.2%	69.4%	50.3%	46.9%
IncV3 guided LGD	77.4%	76.7%	75.0%	76.2%	67.1%	76.4%	75.8%	75.1%	70.3%
Resnet guided LGD	78.4%	77.8%	74.5%	77.7%	68.1%	76.7%	76.2%	77.2%	72.0%

Table S8. The details of Table 5.

Defense	Clean	F_v3_e4	I4_ens_e4	F_v3_e16	I4_ens_e16	F_v4_e4	I4_v4_e4	F_v4_e16	I4_v4_e16
NA	76.6%	22.8%	8.0%	25.5%	5.0%	60.7%	62.4%	48.2%	35.1%
LGD	76.3%	74.8%	73.0%	68.4%	63.0%	74.9%	74.7%	74.7%	69.7%

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 4

[2] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. 4

[3] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[4] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, pages 4278–4284, 2017. 4

[5] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016. 1

[6] F. Tramèr, A. Kurakin, N. Papernot, D. Boneh, and P. McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017. 1, 4