Ensemble-based Defensive Pipeline Against Adversarial Attacks

Junbo Huang (1004280422), Jason Tang (1004221326)

Abstract—Adversarial attacks present a significant security risk to most modern usages of machine learning, especially in the field of computer vision. To our knowledge, there are no defense systems that can provide robustness against white-box adversarial attacks with a computationally efficient run time. For the white-box setting, we propose GPEnsemble, a gaussian pyramid inspired ensemble defense system combined with an adversarially trained DnCNN denoiser preprocessor. Our system outperforms previous defense systems (Ensemble Adversarial Training and Fast Adversarial Training) at the strongest tested attack (C&W), with our system retaining 72.6% accuracy at the highest tested perturbation level while the other systems falling to 7.4% and 9.6% accuracy, respectively.

The code for this project is available at: https://github.com/JasonTang99/csc2529_project

1 Introduction

In recent years, machine learning (ML) models have improved dramatically in their ability to perform visual tasks on naturally occurring images. However, most ML systems in use today were designed with minimal consideration of the exploits a malicious actor could employ against these systems. One class of these adversarial attacks focuses specifically on tampering with model integrity such that it outputs incorrect predictions, potentially in a manner benefiting attackers. Some potential exploits include: falsifying cheques, bypassing facial recognition systems, and causing abnormal self-driving car behavior by altering detected road signs.

These model integrity attacks generally utilize model weights to perform constrained gradient steps in the input image space or to solve constrained optimization problems, both of which generate imperceptibly small perturbations to input images that cause models to produce erroneous outputs. Even in black-box scenarios where attackers have no access to model weights and architecture, there are methods that can learn a substitute clone of the target model and exploit the transferability of adversarial examples to attack black-box models without knowledge of model architecture or even access the same training data [1]. This nullifies many attempts of security through obscurity, causing a shift towards defenses in gray-box or white-box scenarios in recent research, where attackers have partial, or even complete access to model architecture, weights, and training data.

In our project, we propose and analyze the robustness of a novel ensemble-based defense system utilizing different input sizes in the white-box setting.

2 Related Work

2.1 Adversarial Attacks

Attacks can be generated by solving a constrained optimization problem, which minimizes or bounds a distance metric (e.g. L_0, L_2, and L_∞) between the clean image and adversarial example. The minimization of image distance encourages imperceptible changes, such that human analysis of adversarial images does not raise concerns.

For standard convolutional neural networks (CNNs), the Fast Gradient Sign Method (FGSM) presents a simple and efficient attack that exploits the existing backpropagation architecture in modern neural networks to efficiently calculate and ascend along the gradient of the loss function with respect to the input image [2]. This gradient step can also be repeated iteratively within the constrained space using Projected Gradient Descent (PGD), and with random nearby initializations to create more powerful attacks [4]. Carlini and Wagner (C&W) present another strong attack which breaks several previously effective defenses by directly optimizing the constrained optimization problem using a margin loss [5]. Lastly, the Skip Gradient Method (SGM) is a recent technique which exploits skip connections in neural networks with residual connections to pass gradients through the model more directly [6].

2.2 Ensemble Defenses

To address this, many defenses have been proposed and defeated in an ongoing arms race within this field. A popular defense is to smooth out the model gradients to near 0 such that the adversarial gradient steps are no longer useful [16] [17]. However, these gradient masking techniques only obfuscate the gradient information, they do not remove the existence of adversarial images. Attackers can effectively bypass this defense by learning a substitute model with useful gradients which can exploit the transferability of adversarial examples to attack the masked target model [1]. Another common defense relies on numerically unstable or non-differentiable functions such as sigmoids, median filters, and step functions. However, attackers can simply replace these problematic layers with a differentiable approximation, and optimize along the slightly incorrect gradients to create adversarial examples [9].

As such, some researchers have begun using ensemble-based methods with the goal of improving model robustness to adversarial examples through diverse ensemble

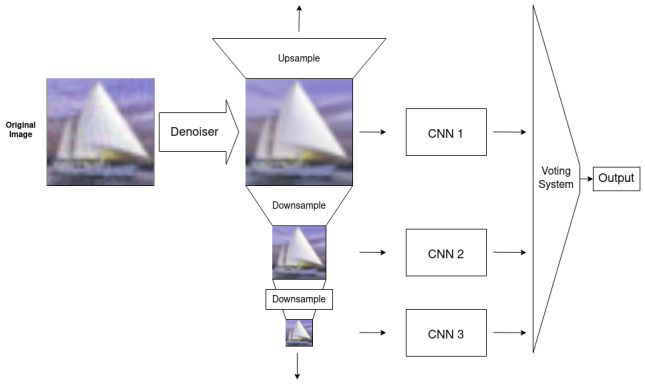

Fig. 1. Illustration of our proposed method.

members. One method uses a regularizer term encouraging orthogonality in non-maximal predictions between ensemble members [10], and another uses different numerical precisions between members [8]. However, both of these methods were broken by running PGD until convergence and by approximating the numerically unstable majority vote with an average, respectively [7]. A successful black-box defense is Ensemble Adversarial Training, which utilizes adversarial examples generated using R+FGSM (a FGSM variation with a random step) on other ensemble members to perform adversarial training [3]. This corresponds to the method attackers used to attack vanilla adversarial training [1], which likely leads to the superior robustness seen with this defense. This idea was extended to the white-box setting with appropriate attack parameter tuning in Fast Adversarial Training [18].

2.3 Denoising Defenses

Many different denoising techniques have been proposed as a defensive mechanism against adversarial attacks. Traditional denoising methods such as Gaussian, Bilateral, Non-local Means, and Total Variation have all proven useful in removing noise in adversarially perturbed images. Simply applying defensive denoising with TV and NLM can remove major parts of the universal adversarial perturbations in images and improve classification performance [11]. A more recent feature denoising [12] method was designed to incorporate NLM as intermediate blocks in the convolutional network of the classifiers; these denoising blocks can be trained to directly remove noises caused by adversarial perturbations on intermediate features.

Deep learning based architectures are also very effective as denoising preprocessors. Convolutional neural networks such as Denoising Autoencoders (DAE) and U-Net have a great capacity for learning and removing adversarial noise. The deep denoising sparse autoencoder (DDSA) [13] method intends to learn a representation extracted from the autoencoder that is robust to adversarial perturbations by adding a sparsity constraint in the bottleneck phase to enforce the extraction of only meaningful and relevant features. However, such bottleneck structures between the encoder and the decoder might hinder the transmission of fine details necessary for high-resolution image reconstruction. A denoising U-net (DUNET) [14] addresses this problem by adding lateral connections and incorporating residual learning. DUNET learns the adversarial noise with a loss function guided by a high-level representation of the images. The DnCNN [15] architecture has also seen lots of success in its capability and efficiency in image denoising. It is a feed-forward neural network that leverages batch normalization and residual learning techniques to estimate the residual in the image, which suggests that it could also potentially be an effective denoising preprocessor.

3 Proposed Method

3.1 Gaussian Pyramid Ensemble

We introduce GPEnsemble, a gaussian pyramid inspired ensemble-based defense system for image classification. Each ensemble member receives a different resized version of the input image (see Fig. 1), where the resampling is done using the same multiplicative scaling factor in each direction for simplicity. We explored using different input scaling factors $[2.0, 1.1]$, as smaller scales allow for more ensemble members with better space efficiency.

The motivation behind this ensemble construction is that the attacker will need to essentially trick each ensemble member at the same time, where each member is diverse and difficult to simultaneously fool due to the varied input sizes. Additionally, downsampling destroys image information so attacks may need to attack entire patches of the input image to target a downsampled model. However, this also produces a drop in accuracy, which is why we also include upsampled input images to alleviate this performance loss.

Each input is then passed through a vanilla Resnet [19] model (18-layer), which is pre-trained on ImageNet and then finetuned on the associated input size. The resulting outputs are then combined in the following ways: leftmargin=*

- *Uniform Average Outputs:* A linear baseline averaging all outputs uniformly.
- *Weighted Average Outputs:* Performance scaled weights will likely improve performance on clean inputs, but could lead to targeted attacks on the highest performing models.
- *Uniform Majority Vote* (non-differentiable): Equal votes introduces a non-differentiable step.
- *Weighted Majority Vote* (non-differentiable): Better accuracy on clean inputs, but may be more vulnerable.

We choose not to consider any random selection methods where a single ensemble model is selected to make the classification decision as it allows attackers to target a single ensemble member and to simply repeatedly submit the adversarial attack until the targeted member is inevitably randomly selected.

3.2 Denoising Preprocessor

Before passing the input images through the ensemble model, a DnCNN-based denoiser is applied as a preprocessor in the first stage of our defensive pipeline. The DnCNN denoiser was trained with adversarial examples generated by different attack methods on a range of perturbation levels, which should remove major parts of the adversarial

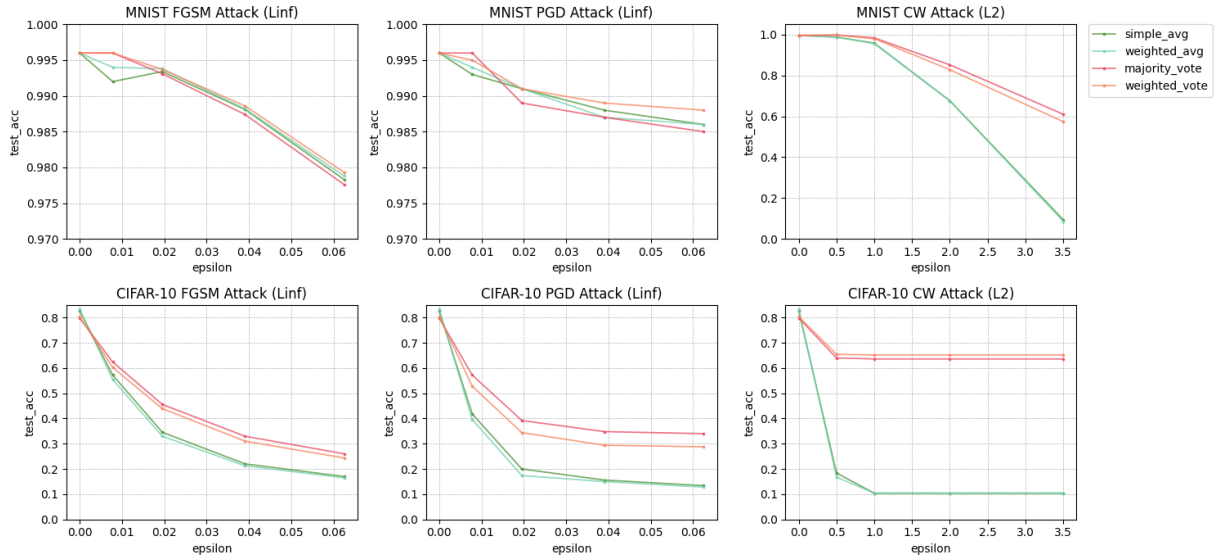

Fig. 2. Comparison of different voting methods. (Ensemble only)

perturbations and restore the original structures in the input images. As seen in recent literature, deep denoisers can be quite effective as a defensive measure. We adopted the stock 17-layer DnCNN architecture and cut it down to a 7-layer model to save training time. The biases in the model are also disabled as we think it would provide better generalization, especially on unseen noises.

4 Evaluation

We examine the robustness of our proposed model with FGSM and PGD attacks on the L_∞ norm, and with C&W attacks on the L_2 norm. Recent literature generally considers L_∞ limits of $[2, 5, 10, 16]$ pixel value changes in FGSM and PGD attacks. Since there is little guidance on similar L_2 limits, we calculated the L_2 norm of the maximal change in the associated L_∞ limit on a $(3, 32, 32)$ sized CIFAR-10 image to produce the L_2 limits of $[0.5, 1.0, 2.0, 3.5]$ for C&W attacks.

Additionally, since C&W doesn't allow the specification of a L_2 limit, we instead run the attack with default parameters and measure the L_2 norm of the perturbations. We only count an attack as successful if the perturbation is within the L_2 limit and the model prediction changes.

Since voting methods have no definable gradient, we generate adversarial examples on the associated differentiable substitute and transfer attacks to the target model.

We evaluate our proposed system on both MNIST and CIFAR10 datasets, and compare against both Ensemble Adversarial Training (black-box) and Fast Adversarial (white-box) methods, which we will refer to as EnsAdv and FastAdv, respectively, on CIFAR10.

5 Results

5.1 Ensemble

In Fig 2 we analyze the performance of only the ensemble portion (no denoiser) using the different voting methods. We note that our non-differentiable voting methods (majority_vote, weighted_vote) generally outperform their differentiable counterparts, especially in the iterative PGD and C&W attacks. This is likely due to the fact that the more powerful iterative attacks often demonstrate less transferability than one-step attacks [3].

TABLE 1
Comparison of different numbers up samplers and down samplers with a scaling factor of 2 on the FGSM attack. Test accuracies averaged over all epsilons. (Ensemble only)

dataset	up_samplers	down_samplers	test_acc
mnist	0.0	0.0	96.94%
	0.0	1.0	97.13%
	0.0	2.0	97.65%
	0.0	3.0	96.41%
	1.0	0.0	97.90%
	2.0	0.0	98.82%
	3.0	0.0	**99.15%**
	1.0	1.0	98.36%
	2.0	2.0	98.88%
	3.0	3.0	**99.01%**
cifar10	0.0	0.0	33.21%
	0.0	1.0	36.43%
	0.0	2.0	39.91%
	0.0	3.0	**40.95%**
	1.0	0.0	37.37%
	2.0	0.0	39.93%
	3.0	0.0	**40.34%**
	1.0	1.0	40.36%
	2.0	2.0	44.54%
	3.0	3.0	**45.48%**

In Table 1, we see that upsamplers are generally more advantageous to have compared to downsamplers. However, due to the exponential space increase required to use these upsamplers, we also utilize downsamplers to increase the ensemble size and robustness.

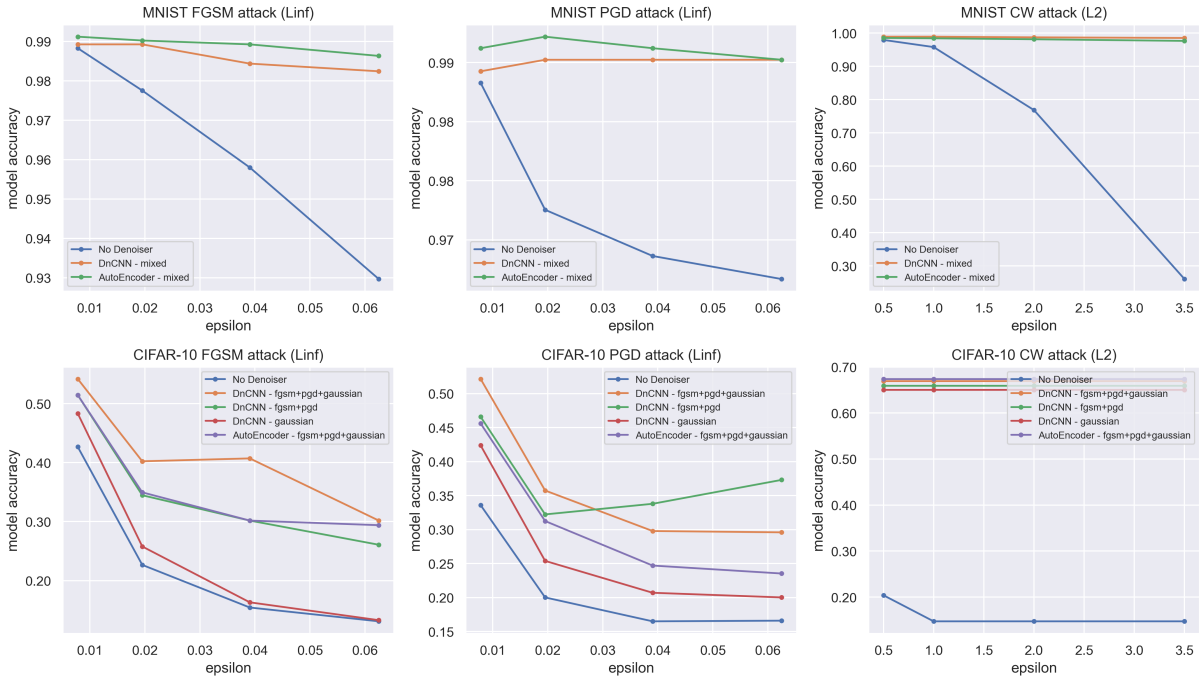

Fig. 3. Comparison of denoisers trained with different dataset/architecture (denoiser only)

TABLE 2
Comparing scaling factors of 2.0 and 1.1. Test accuracy averaged over epsilons. (Ensemble only)

dataset	attack	scaling, up, down	test_acc
mnist	fgsm	1.1, 3, 3	98.44%
		1.1, 5, 5	98.65%
		1.1, 7, 7	98.79%
		2.0, 3, 3	**99.01%**
	pgd	1.1, 3, 3	98.65%
		1.1, 5, 5	98.70%
		1.1, 7, 7	98.62%
		2.0, 3, 3	**99.10%**
	cw	1.1, 3, 3	67.20%
		1.1, 5, 5	75.99%
		1.1, 7, 7	77.09%
		2.0, 3, 3	**81.15%**
cifar10	fgsm	1.1, 3, 3	43.25%
		1.1, 5, 5	43.93%
		1.1, 7, 7	**45.25%**
		2.0, 3, 3	**45.48%**
	pgd	1.1, 3, 3	47.78%
		1.1, 5, 5	49.89%
		1.1, 7, 7	**51.86%**
		2.0, 3, 3	40.62%
	cw	1.1, 3, 3	37.91%
		1.1, 5, 5	45.95%
		1.1, 7, 7	**47.17%**
		2.0, 3, 3	**46.98%**

We also considered lowering the scale factor from 2.0 to 1.1, with results in Table 2. The lower scaling factor allows the addition of more upsamplers and downsamplers before running into memory space constraints or complete loss of image information. The results show that using a factor of 2.0 will always dominate a factor of 1.1 when both have the same ensemble member counts, but using the ability to have more ensemble members in 1.1 produces better results in the more challenging CIFAR10 dataset.

5.2 Denoising

For evaluating the denoising performance, we were primarily focusing on the CIFAR10 dataset as it is more representative in real-world scenarios. Our custom training dataset contains adversarial examples generated by FGSM and PGD methods on a range of l_∞ norms, and additional noisy examples with added Gaussian noise. It is worth noting that we did not include C&W examples, as images perturbed by the C&W attack have no perceivable noise and are structurally similar to the original images (See Table 3 and Fig. 4). Therefore, training the denoisers on these examples will give us no benefit in learning and image reconstruction performance.

We conducted ablation studies by comparing denoiser models trained on different combinations of training examples and evaluated their respective PSNR, SSIM value in image restoration, and classification accuracy in each attack scenario. We have discovered that training with gaussian noise examples generally leads to improved denoising and generalization performance. As an additional comparative analysis, we have compared the denoising performance of our DnCNN model with a 10-layer convolutional autoencoder (ConvDAE) [20] trained on the same dataset.

In Table 3 and 4, we see that the DnCNN denoiser trained with FGSM, PGD, and gaussian noise examples had the best overall performance under FGSM and PGD attacks, averaging a 17.9% and 15.2% accuracy increase compared

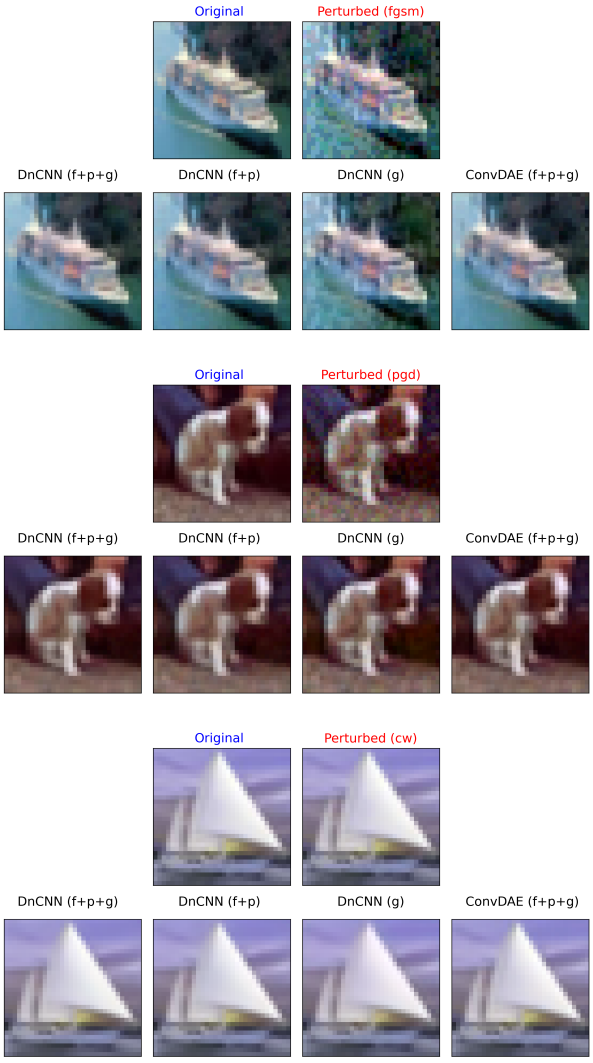

Fig. 4. Sample adversarial images and denoised outputs

to the baseline model without a denoiser. We also observed that the effect of C&W attack was alleviated almost completely with all 4 denoisers, suggesting that image smoothing can be the major factor in reverting the perturbations in C&W attacked images. In Fig. 4, we can visually observe that denoisers trained with mixed training examples yielded the best overall image reconstruction quality. Overall, the DnCNN model achieved better defensive denoising performance compared to the ConvDAE model on the CIFAR10 dataset, while being more lightweight and easy to train.

5.3 Denoiser + Ensemble

The combination of the denoiser and ensemble components produces significant improvements over simply the ensemble model, particularly at high perturbations and with stronger attacks (See Fig. 5). We note up to a 40.8% and 21.0% accuracy improvement in MNIST and CIFAR10, respectively, in the C&W attack.

Lastly, in Fig. 6, we compare the performance of our strongest ensemble models across both scaling factors against Ensemble Adversarial Training (EnsAdv) and Fast Adversarial Training (FastAdv). We can see that FastAdv, which improves upon EnsAdv, outperforms our method on both FGSM and PGD attacks. However, the combination of our denoiser and ensemble steps is able to outperform both FastAdv and EnsAdv by up to 63% at the maximum perturbation level.

TABLE 3
Average PSNR and SSIM values of restored images in FGSM (l_∞, eps=0.0625), PGD (l_∞, eps=0.0625), and C&W attacks (l_2, eps=3.5) on CIFAR-10

Attack	Denoiser	PSNR (dB)	SSIM
FGSM	DnCNN - (fgsm+pgd+gaussian)	**31.816**	**0.974**
	DnCNN - (fgsm+pgd)	30.648	0.965
	DnCNN - (gaussian)	25.857	0.888
	ConvDAE - (fgsm+pgd+gaussian)	28.883	0.948
	Baseline (No Denoiser)	24.207	0.843
PGD	DnCNN - (fgsm+pgd+gaussian)	**33.626**	**0.979**
	DnCNN - (fgsm+pgd)	32.072	0.971
	DnCNN - (gaussian)	32.074	0.970
	ConvDAE - (fgsm+pgd+gaussian)	33.398	0.979
	Baseline (No Denoiser)	28.730	0.931
C&W	DnCNN - (fgsm+pgd+gaussian)	39.656	0.994
	DnCNN - (fgsm+pgd)	39.764	0.994
	DnCNN - (gaussian)	36.873	0.987
	ConvDAE - (fgsm+pgd+gaussian)	**44.432**	**0.998**
	Baseline (No Denoiser)	∞	0.999

TABLE 4
CIFAR-10 classification accuracy averaged over epsilons (denoisers only)

Denoiser	Test Accuracy		
	FGSM	PGD	C&W
DnCNN - (fgsm+pgd+gaussian)	**41.33%**	**36.82%**	66.89%
DnCNN - (fgsm+pgd)	35.55%	37.48%	65.92%
DnCNN - (gaussian)	25.93%	27.12%	65.04%
ConvDAE - (fgsm+pgd+gaussian)	36.5%	31.27%	**67.38%**
Baseline (No Denoiser)	23.46%	21.68%	16.16%

6 Conclusion and Future Work

We have shown that the combination of the ensemble defense system and an adversarially trained denoising preprocessor provides a robust layer of protection against adversarial attacks. Our system was able to outperform the previously proposed ensemble-based defense system (Ensemble Adversarial Training) in all attacks tested, and edge over the Fast Adversarial Training method in the strongest tested C&W attack.

Our system's capability can also be extended to more complex models and larger datasets, such as ImageNet. There is also the opportunity to explore non-differentiable up and down sampling methods for the image resizing step to add another layer of non-differentiability against potential attacks. Additionally, adversarial training could potentially be incorporated into the training we performed in each ensemble member.

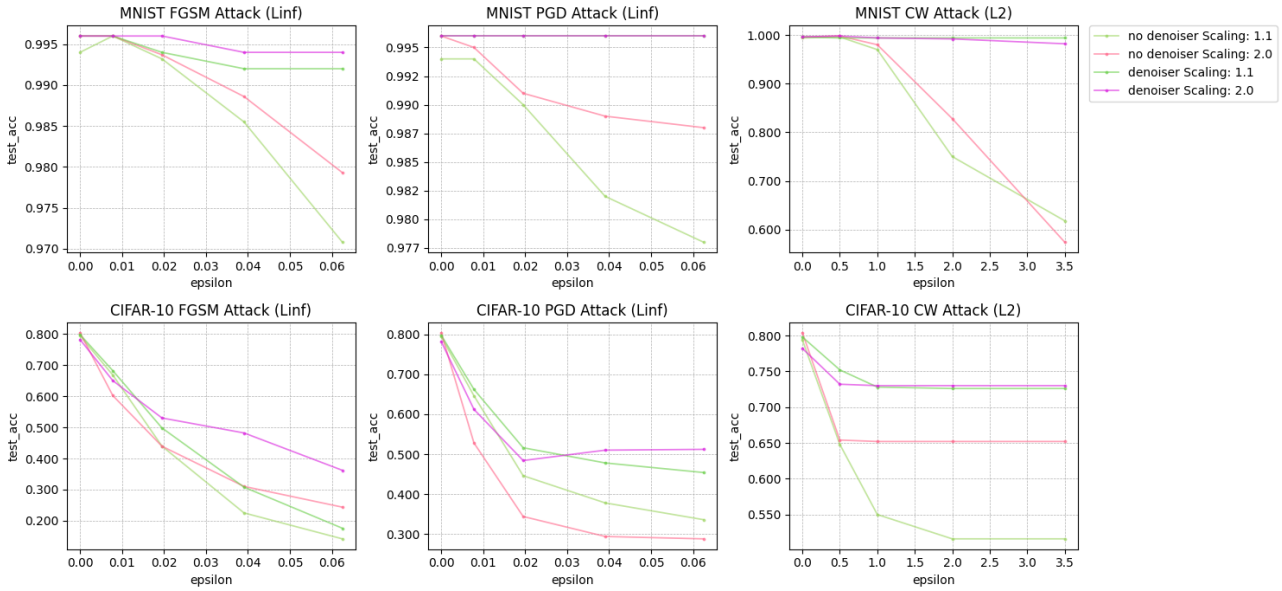

Fig. 5. Effects of including a denoiser to the ensemble.

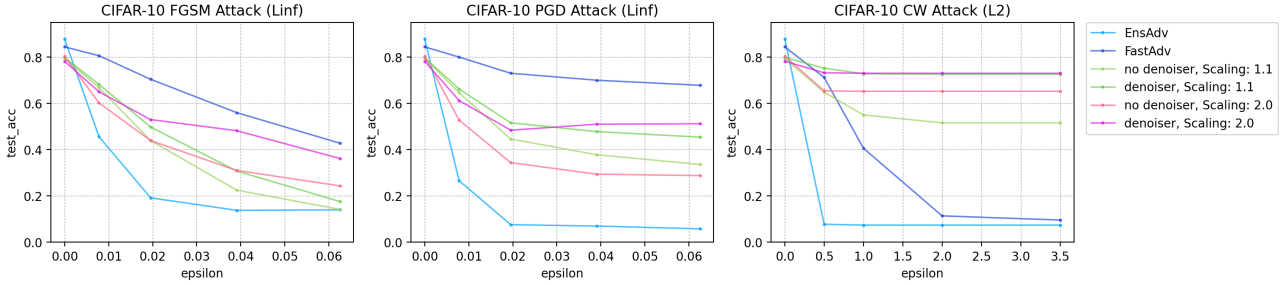

Fig. 6. Comparison with Ensemble Adversarial Training (EnsAdv) and Fast Adversarial Training (FastAdv).

REFERENCES

[1] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in Proceedings of the 2017 ACM on Asia conference on computer and communications security, 2017, pp. 506–519.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014.

[3] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," arXiv preprint arXiv:1705.07204, 2017.

[4] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," arXiv preprint arXiv:1706.06083, 2017.

[5] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in 2017 ieee symposium on security and privacy (sp), 2017, pp. 39–57.

[6] D. Wu, Y. Wang, S.-T. Xia, J. Bailey, and X. Ma, "Skip connections matter: On the transferability of adversarial examples generated with resnets," arXiv preprint arXiv:2002.05990, 2020.

[7] F. Tramer, N. Carlini, W. Brendel, and A. Madry, "On adaptive attacks to adversarial example defenses," Advances in Neural Information Processing Systems, vol. 33, pp. 1633–1645, 2020.

[8] S. Sen, B. Ravindran, and A. Raghunathan, "Empir: Ensembles of mixed precision deep networks for increased robustness against adversarial attacks," arXiv preprint arXiv:2004.10162, 2020.

[9] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in International conference on machine learning, 2018, pp. 274–283.

[10] T. Pang, K. Xu, C. Du, N. Chen, and J. Zhu, "Improving adversarial robustness via promoting ensemble diversity," in International Conference on Machine Learning, 2019, pp. 4970–4979.

[11] Lee, S., Lee, J., and Park, S. (2018). Defensive denoising methods against adversarial attack.

[12] Maaten, L. V. D., Yuille, A. L., and He, K. (2019). Feature Denoising for Improving Adversarial Robustness. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[13] Bakhti, Y., Fezza, S. A., Hamidouche, W., and Deforges, O. (2019). DDSA: A Defense Against Adversarial Attacks Using Deep Denoising Sparse Autoencoder. IEEE Access, 7, 160397–160407.

[14] Liao, F., Liang, M., Dong, Y., Pang, T., Hu, X., and Zhu, J. (2018). Defense Against Adversarial Attacks Using High-Level Representation Guided Denoiser. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[15] Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017). Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. IEEE Transactions on Image Processing, 26(7), 3142–3155.

[16] C. Szegedy et al., "Intriguing properties of neural networks," arXiv preprint arXiv:1312.6199, 2013.

[17] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in 2016 IEEE symposium on security and privacy (SP), 2016, pp. 582–597.

[18] E. Wong, L. Rice, and J. Z. Kolter, "Fast is better than free: Revisiting adversarial training," 2019.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in European conference on computer vision, 2016, pp. 630–645.

[20] X.-J. Mao, C. Shen, and Y.-B. Yang, "Image Restoration Using Convolutional Auto-encoders with Symmetric Skip Connections," CoRR, vol. abs/1606.08921, 2016