AN EFFECTIVE FUSION METHOD TO ENHANCE THE ROBUSTNESS OF CNN

Yating Ma¹, Zhichao Lian²

1 School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
2 School of Cyberspace Security, Nanjing University of Science and Technology, Wuxi, China
yatingm@njust.edu.cn, lzcts@163.com

ABSTRACT

With the development of technology rapidly, applications of convolutional neural networks have improved the convenience of our life. However, in image classification field, it has been found that when some perturbations are added to images, the CNN would misclassify it. Thus various defense methods have been proposed. The previous approach only considered how to incorporate modules in the network to improve robustness, but did not focus on the way the modules were incorporated. In this paper, we design a new fusion method to enhance the robustness of CNN. We use a dot product-based approach to add the denoising module to ResNet18 and the attention mechanism to further improve the robustness of the model. The experimental results on CIFAR10 have shown that our method is effective and better than the state-of-the-art methods under the attack of FGSM and PGD.

Index Terms— model defense; denoising; attentional mechanisms; fusion method

1. INTRODUCTION

With the continuous development of convolutional neural networks (CNN), CNNs have achieved high performances in the field of object recognition, image classification and so on. However, in recent years, it has been found when some elaborate perturbations is added to the image, it will cause the CNNs to misclassify the objects[1,2]. This vulnerability has attracted a great deal of attentions and researchers attempt to defend the CNNs against such attacks.

The available defenses can be divided by the phases in which they are implemented. One type is before the images are fed into the CNNs. Goodfellow et al. in [1] show that the perturbations can be seen as an elaborate noise, so that a good defense can be achieved using denoising techniques. Liao et al. [24] proposes high-level representation guided denoiser (HGD) as a defense for image classification. Yan et al. [25] integrate an adversarial perturbation-based regularizer into the classification objective, such that the obtained models learn to resist potential attacks. Another type is to modify the network structure by introducing a network of attention mechanisms in the classification network, including spatial attention and channel attention, so that the robustness of the model can be improved by making the model learn the regions or activate the channels associated with the classification labels. Yan et al.[26] show neural ODE-based models are inherently more robust than conventional CNN models; Guo et al.[27] demonstrate that appropriately designed higher model sparsity implies better robustness of nonlinear networks. Inspired by [9], which integrated channel attention module and spatial attention module in a novel way, we propose a novel defense method combining the denoising module and attention mechanisms. More specifically, our contributions are as follows: 1) We propose a tighter dot product-based module fusion approach that allows the denoising effect to be better transmitted to the deeper layers of the network. 2) We apply this approach to ResNet18 and attention mechanism networks respectively, and experimentally show that this approach can improve the robustness of both networks.

2. RELATED WORK

In terms of channel attention mechanism, CAS [3] trained an inherent model through a channel activation strategy to suppress redundant activations being activated by adversarial perturbations, solving the problem that adversarial samples activate channels more uniformly than natural samples. CBAM [9] inferred the attention graph sequentially along two separate dimensions (channel and space) given an intermediate feature map, and then multiplied the attention map for adaptive feature refinement. Channel Transformer Network [4] proposed a new parameter-free method called Channel Transformer Network (CTN) to reduce or increase the number of channels in a convolutional neural network module while maintaining most of the information with low computational complexity, which can be used for many vision tasks such as image classification and object detection, etc. The work DFA in [5] proposed a double-fused attention module that can adjust the distribution of features by generating a double-fused attention mask that
depends on spatial location and channel information, so that each corresponding feature representation can adaptively enrich its discriminative region and minimize the effect of background noise.

In the denoising defense domain, block matching 3D (BM3D) based filters [6] were fed by optimal thresholds estimated by convolutional neural networks (CNNs) to project samples detected as AE back into their data stream shapes. Denoiser-based class activation features (CAFD) [7] eliminated adversarial noise by implementing a self-supervised adversarial training mechanism in the class activation feature space, minimizing the distance between the adversarial and natural samples in the class activation feature space to achieve denoising defense. Using an ensemble color space model to tackle adversarial examples using L1 parametric penalty term, only noisy pixels were processed and recovered to noise-free images using MAP estimator, and the denoised images were LAB, HSV and YUV transformed to obtain feature maps that are integrated to achieve defense. AdvFilter [8] proposed predictive perturbations perception and pixel-by-pixel filtering, and designed a dual perturbations filtering and un-certainty-aware fusion module to automatically sense the perturbations magnitude during training and testing. But about the combination of the two defense methods has not been proposed.

3. APPROACH

According to previous works[10,11,12], added perturbations tended to show strong activation in semantically irrelevant places, and we can assume that strong activation indicates the presence of semantic information about the content of the image (as is often assumed [19]). So, as [13] shown, feature denosing could be a way to reduce perturbations, which make CNN classify image precisely.

We embed the denoising module into the attention mechanism network in a more tightly integrated way, so that the denoising effect can be transferred to the attention mechanism network.

3.1. Denoising Module

The perturbations added by the attacker are constrained to be invisible to the human eye within a certain paradigm, but the feature maps derived from the middle layer of the network show that regions unrelated to the classification labels are activated. At the same time, these perturbations are amplified as the number of layers of the network deepens and the effect of the perturbations is amplified. And the experience of [13] shows that this carefully crafted perturbations can be seen as noise. The robustness of the model can be improved by simply using some denoising operations before the adversarial samples are fed into the recognition model. For this reason, we propose to incorporate a denoising module in the shallow network to implement the defense. This proposed denoising module combines a self-attention transformer and some common filters and finally combines them using residuals.

The self-attention is implemented through a $1 \times 1$ convolutional layer, combined with residual connections for feature combination. For denoising, we chose two simple and effective operations: the mean filter and the median filter. The mean filter is an average pooling with a stride of 1. The median filter is defined as follows, where $F$ denotes feature map and $D$ denotes feature map after denoised. The median filter is defined in a local region $\Omega(i)$, and is performed separately for each channel. In the experimental section we will show the effect of both defenses.

$$D_i = \text{median}\{\forall j \in \Omega(i) : F_j\}$$  \hspace{1cm} (1)

3.2. Connection Type

From [9], we understand that the transformations performed by the layers in the network amplify the perturbations, which can be avoided by the denoising module. In this section, we propose a new way of module fusion to suppress perturbations. We incorporate the denoising results into ResNet18 in a dot product manner and residual connection in which the denoising value can be made to pass to the deeper layers of the network, instead of placing them directly in the middle layer of the network. The fusion method of the denoising module is shown in Figure 2. For a

![Figure 1. Overall structure based ResNet8.](image-url)
given feature map $F \in R^{C \times H \times W}$, the denoising result is $\text{denoising}(\cdot)$, and the overall process is as follows,

$$F' = \text{denoising}(F) \otimes F$$

$$F'' = \text{denoising}(\text{res}_1(F')) \otimes \text{res}_1(F')$$

where $\otimes$ denotes the dot product, $\text{res}_1(\cdot)$ denotes the first residual structure of ResNet18.

The location of the denoising module in ResNet18 is shown in Figure 1. We put denoising module before first layer and second layer to remove some of the perturbations. In addition to this, we also apply this incorporation to the attention mechanism network. We use a channel attention network named CIFS[3] to detect the effect of the denoising module in combination with it. CIFS[3] achieves defense by activating the channels related to classification labels and suppresses the channels unrelated to classification. Experimental results show that it has defense effect in both architectures of the network.

4. EXPERIMENTS

In this section, we adopt adversarial training[16] to generated adversarial examples. We use FGSM[1] and LinfPGD[17] to check the defense performance of our method on ResNet18 and CIFS. Our experimental model is divided into four versions, the first one places the denoising module directly in ResNet18, and the second uses our way to embed it in ResNet18, and the same comparison is done for CIFS to verify its effect on the attention mechanism network. All experiments are conducted on CIFAR10[18].

We have done the comparison under LinfPGD. We set the number of iterations to 10 and 20 and test their defense effectiveness at iterations of 10, 20 and 100, respectively. And the learning rate is 0.01 and we train 120 epochs. The experimental results are shown in Table 1, from which we can see that our method performs well under LinfPGD10 and LinfPGD20. The experiments show that the robustness of ResNet18 under adversarial training is improved for both LinfPGD10 and LinfPGD20. This improvement is more evident in the attention mechanism network, where we can see that if the denoising module is added directly to CIFS, its corresponding defense capability under adversarial training with LinfPGD10 does not improve but decreases, while using our fusion approach can compensate for this deficiency with a defense rate of 54.54%. Similarly, under adversarial training with LinfPGD20, our approach achieves a breakthrough in defense capability in both cases (see Table 3,4).

Table 1. Defense rate under LinfPGD10 using adversarial training. The first line is the basic CIFS, the second joins denoising module with mean filter, and the last is our method.

| Defense Method | Natural Mean | LinfPGD10 Mean | LinfPGD Mean | LinfPGD Mean |
|----------------|--------------|----------------|--------------|--------------|
| CIFS           | 82.88        | 54.67          | 51.41        | 50.19        |
| Mean-CIFS      | 83.05        | 54.19          | 50.97        | 50.58        |
| Mean-CIFS-dot  | 83.46        | 54.54          | 51.38        | 50.85        |

Table 2. The defense rate based ResNet18 under LinfPGD10 using adversarial training.

| Defense Method | Natural Mean | LinfPGD10 Mean | LinfPGD Mean | LinfPGD Mean |
|----------------|--------------|----------------|--------------|--------------|
| Mean-ResNet18  | 83.89        | 52.71          | 50.12        | 48.76        |
| Mean-ResNet18-dot | 84.38     | 53.22          | 50.74        | 49.11        |

Table 3. The defense rate under LinfPGD20 using adversarial training. The first line is the basic CIFS, the second joins denoising module with mean filter, and the last is our method.

| Defense Method | Natural Mean | LinfPGD20 Mean | LinfPGD Mean | LinfPGD Mean |
|----------------|--------------|----------------|--------------|--------------|
| CIFS           | 81.97        | 54.53          | 51.61        | 51.24        |
| Mean-CIFS      | 81.11        | 54.72          | 51.62        | 50.85        |
| Mean-CIFS-dot  | 82.21        | 55.14          | 52.4         | 51.81        |

Table 4. Defense rate based ResNet18 under LinfPGD20 using adversarial training.

| Defense Method | Natural Mean | LinfPGD20 Mean | LinfPGD Mean | LinfPGD Mean |
|----------------|--------------|----------------|--------------|--------------|
| Mean-ResNet18  | 83.68        | 53.36          | 50.86        | 49.29        |
| Mean-ResNet18-dot | 84.06      | 53.59          | 51.37        | 49.64        |
Figure 3. Compare the fusion effect under ResNet18 with mean filter.

Figure 4. The fusion effect under three types with mean filter against FGSM. The size of perturbation is 8/255 in adversarial training.

Figure 5. The fusion effect under three types with median filter against FGSM.

We compare all the four versions against FGSM and join basic CIFS. We set the perturbations to 8/255, train the model using adversarial training, and then test the defense effect under three different perturbation attacks like 8/255, 12/255, 20/255. In Figure 3, with ResNet18 and mean filter, except for the large improvement of the defense effect for perturbation 12, our method has different degrees of degradation for the defense under other perturbations, which may be due to the fact that FGSM adds perturbations in a different way than PGD. Meanwhile, we add mean filter and median filter to CIFS to examine the defense effect respectively, and the differences between them can be seen in Figure 4 and 5. The results show that the direct addition of the denoising module not only do not assist to the attention mechanism network, but even weaken its effect, while our method can solve this problem well and improve a lot in defense rate.

Compared to LinfPGD and FGSM, FGSM computes perturbations through the parameter of epsilon after one step and the result of experiments shows our method works well for single-step attacks. And it has been widely used because of the advantage of fast attack speed and effect. So we think defense against FGSM is meaningful. While PGD computes perturbations by multi steps, the difficulty of defense is getting harder. However, our method improves the defense rate contrast to baseline. The magnitude of the noise is relatively small which causes the improvements are relatively small. This also provide a new direction for our future work.

5. CONCLUSION

We propose a new fusion method that can be fused with different network structures including ResNet18 and the attention mechanism network CIFS, indicating the superior generalization. Our experiments demonstrate that it has a significant improvement effect under the defense against FGSM and PGD perturbation attacks and achieves the state-of-the-art robust results so far in combination with CIFS. In future, we plan to explore the effect of this combination in other network models. Based on the improvements of FGSM, we will further exploit the improvements of PGD.

6. ACKNOWLEDGEMENTS

This work is supported by the National Key R&D Program of China (2021YFF0602104-2).

7. REFERENCES

[1] Goodfellow, Ian J. et al. “Explaining and Harnessing Adversarial Examples.” CoRR abs/1412.6572 (2015).
[2] Szegedy, Christian et al. “Intriguing properties of neural networks.” CoRR abs/1312.6199 (2014).
[3] Bai, Yang et al. “Improving Adversarial Robustness via Channel-wise Activation Suppressing.” ArXiv abs/2103.08307 (2021).
[4] H. Zhu, M. Long, J. Wang, and Y. Cao, “Deep hashing network for efficient similarity retrieval,” in AAAI, pp. 2415-2421, 2016.
[5] Z. Cao, M. Long, J. Wang, et al, “Hashnet: Deep learning to hash by continuation,” in CVPR, pp. 5608-5617, 2017.
Decaf: A deep convolutional

[6] L. K. Huang, J. Chen, S. J. Pan, “Accelerate learning of deep hashing with gradient attention,” in CVPR, pp. 5271-5280, 2019.

[7] A. Andoni and P. Indyk, “Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions,” in IEEE Symposium on. IEEE, pp. 459–468, 2006.

[8] A. Gionis, P. Indyk, R. Motwani, et al, “Similarity search in high dimensions via hashing,” VLDB, vol. 99, pp. 518–529, 1999.

[9] Y. Weiss, A. Torralba, and R. Fergus, “Spectral hashing,” in NIPS, 2009, pp. 1753–1760.

[10] X. Liang, J. Shen, J. Han, Z. Lei, S. Ling, “Dynamic multi-view hashing for online image retrieval,” in IJCAI, 2017.

[11] A. Gionis, P. Indyk, R. Motwani, “Similarity search in high dimensions via hashing,” in Proc. 30th Int. Conf. Very Large Data Bases, pp. 518–529, 1999.

[12] Y. Weiss, A. Torralba, R. Fergus, “Spectral hashing,” Advances in neural information processing systems, vol. 21, pp. 1753-1760, 2008.

[13] Y. Gong, S. Lazebnik, “Iterative quantization: A procrustean approach to learning binary codes,” in CVPR, pp. 817–824, 2011.

[14] J. Wang, S. Kumar, and S.-F. Chang, “Semi-supervised hashing for large scale search,” IEEE Trans. Pattern Anal. Mach. Intell, vol. 34, pp. 2393–2406, 2012.

[15] F. Shen, C. Shen, W. Liu, et al, “Supervised Discrete Hashing,” in CVPR, pp. 37–45, 2015.

[16] A. Krizhevsky, I. Sutskever, G. E. Hinton, et al, “ImageNet Classification with Deep Convolutional Neural Networks,” in neural information processing systems, pp.1097-1105, 2012.

[17] X. Wang, Y. Shi, K. M. Kitani, “Deep supervised hashing with triplet labels,” in Asian conference on computer vision, pp.70-84, 2016.

[18] L. Jin, X. Shu, K. Li, Z. Li, G. Qi, J. Tang, “Deep Ordinal Hashing with Spatial Attention,” IEEE Trans. Image Process, vol.28, pp.2173-2186, 2019.

[19] J. Fu, J. Liu, H. Tian, et al, “Dual Attention Network for Scene Segmentation,” in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[20] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” in ICML, 2014.

[21] A. Paszke, S. Gross, S. Chintala, et al, “Automatic differentiation in PyTorch,” in NIPS, pp.1-4, 2017.

[22] T. S. Chua, J. Tang, R. Hong, et al, “NUS-WIDE: a real-world web image database from National University of Singapore,” in Proceedings of the ACM international conference on image and video retrieval, pp. 1-9, 2009.

[23] T. Y. Lin, M. Maire, S. Belongie, et al, “Microsoft coco: Common objects in context,” in European conference on computer vision, pp. 740-755, 2014.

[24] Liao, Fangzhou et al. “Defense Against Adversarial Attacks Using High-Level Representation Guided Denoiser.” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018): 1778-1787.

[25] Yan, Ziang et al. “Deep Defense: Training DNNs with Improved Adversarial Robustness.” NeurIPS (2018).

[26] Yan, Hanshu et al. “On Robustness of Neural Ordinary Differential Equations.” ArXiv abs/1910.05513 (2020): n. pag.

[27] Guo, Yiwen et al. “Sparse DNNs with Improved Adversarial Robustness.” ArXiv abs/1810.09619 (2018): n. pag.