Boosting the Transferability of Adversarial Examples with More Efficient Data Augmentation

Boosting the Transferability of Adversarial Examples with More Efficient Data Augmentation

Chenxiang Gao[1,a], Wei Wu[2,b]

[1]Department of Information Engineering, Wuhan University of Technology, Wuhan, Hubei, 430000, China

[2]Department of Information Engineering, Wuhan University of Technology, Wuhan, Hubei, 430000, China

[a]email: GaoChenXiang@whut.edu.cn, [b]email: wuwei@whut.edu.cn

Abstract. Currently, Deep Neural Networks has achieved excellent results on many tasks. But recent studies have shown that these networks are easily influenced by adversarial examples, which are artificially crafted by adding perturbation to original image. Moreover, most of the models to which we can access are black-box, we don't know the internal structure and parameters of the model. Thus, it is more practical and more challenging to study how to attack these models. In this article, we propose a cam(class activation map)-guided data augmentation attack method, which can improve the transferability of adversarial examples. Specifically, first use the trained network to get the class activation maps for an input image, then binarize the cam to get the mask, finally implement the data augmentation attack method on the masked area of the image. Experiments based on ImageNet prove that our proposed method can generate more transferable adversarial examples, and the attack success rates of our method have a certain improvement compared with the latest methods.

1. Introduction

In recent years, adversarial examples bring many safety hazards to Deep Neural Networks, which seriously affects the implementation of these high-performance black boxes, especially in security domain, such as autonomous driving, face recognition. Research on how to improve the transferability of adversarial examples will cause people to consider the security of deep learning and explore the corresponding defensive methods. So, how do we attack these black boxes? There comes to the black-box adversarial attack. it can be divided into two parts: query-based and transfer-based. The former uses massive queries to get the output logits of the black-box model, and then simulates the attacked model. The latter is developed from the transferability of adversarial examples. Unfortunately, this method performs poor in black-box model, but we can use a variety of techniques to improve the transferability of adversarial examples. In this article, we only study the transfer-based. This method is convenient to craft adversarial examples, but its attack success rate is relatively low. A work[9] has proposed to apply random transformations to input images at each iteration, to improve the transferability of adversarial examples. Inspired by it, we proposed a cam-guided data augmentation attack method, which enabled us to achieve more efficient data augmentations and increase the diversity of adversarial examples, then achieve attacks on crucial region. Moreover, our method also considered the information of non-critical region. In summary, the main contributions of our work can be concluded as the following:

- We propose a CGDI^2-FGSM to make the attack more focused on the target region of image, and we have not forgotten the information of the non-target region.
- We propose an attention extraction module, which using visual attention techniques to obtain the target region and non-target region of different image

2. Related Works

According to the attacker's understanding of the attacked model, the adversarial attack methods can be divided into two major categories: white-box adversarial attack and black-box adversarial attack.

2.1. White-box adversarial attack

As the name implies, the white-box means that the attacker knows the internal structure and weight parameters of the attacked model. While the attacker can easily craft adversarial examples with a high attack success rate by using the gradient information of the model. Szegedy et al[1] first proposed and confirmed the existence of adversarial examples, and pointed out that adversarial examples are transferable. Then Goodfellow et al conducted an extended research base on [1], and proposed the famous FGSM(Fast Gradient Sign Method) attack[2]. Immediately afterwards, some scholars proposed an iterative version of the FGSM——BIM (Basic Iterative Method) attack[3]. Aleksander Madry et al[4] found that if the start point was randomly selected during each iteration, they could craft better adversarial examples. In addition to optimization method for gradients, there were works that studied the influence of disturbed pixel positions[5], different norms and different object functions[6] on adversarial examples. However, in real life, most of the practical deep learning models or APIs were black-box, and we could hardly know any information inside it.

2.2. Black-box adversarial attack

Then there came to black-box adversarial attack. It could be further divided into two categories: query-based and transfer-based. In this article, we only study the latter, which suffers from the low attack success rate on different networks. So, the main work was to improve the transferability of adversarial examples. A work[7] proposed MI(Momentum Iterative)-FGSM, which added the momentum term during performing the gradient update step of each iteration. Some scholars also proposed to combine the improved adam gradient descent algorithm with the iterative gradient attack method[8]. Inspired by data augmentation, Xie et al. proposed DI^2(Diverse Inputs Iterative)-FGSM attack[9], which increase the chance of attacking different networks by creating diverse input patterns. Wu et al. pointed out that the reason of the poor transferability was that, it had been over-fitting under the current model. Therefore, this work proposed the ATA(Attention-guided Transfer Attack) method[10].

When the DI^2-FGSM performs data augmentation, it does not consider the fact that the proportion of foreground target is small. In this case, the diversity caused by random data augmentation is poor because of not changing the target features. Moreover, a work in the fine-grained classification has proved that using attention map for data augmentation can get better results[11]. So, our work proposed to use the model's class activation map for the input image to implement more efficient data augmentation. We named it cam-guided DI^2-FGSM (CGDI^2-FGSM).

3. Approach

In this section, we will introduce the proposed attack method in detail, which mainly includes the following three parts: cam extraction module, cam-guided data augmentation attack method, and loss function design. We design a cam extraction module to extract the model's class activation maps for an input image, then binarize the cam according to the threshold to obtain a mask, finally use the mask to crop the input image. And the cropped input is what we need to attack during each iteration. The process is shown in Fig.1.

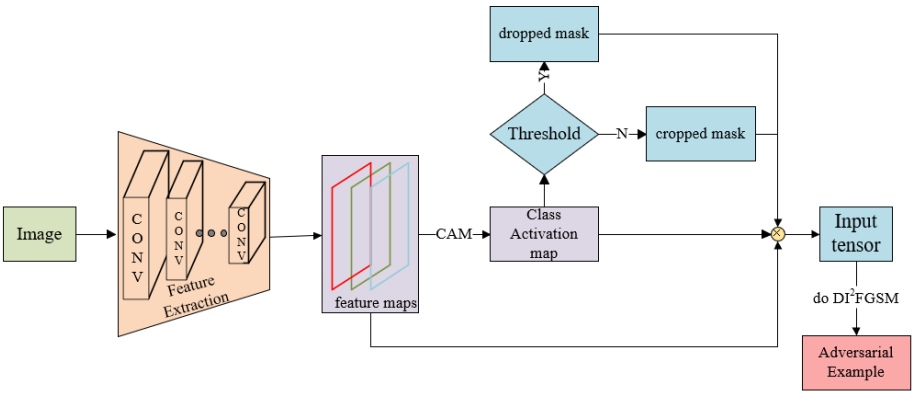

Figure.1 Overview of our approach

3.1. Cam Extraction module

For a picture, most people focus on the same area. Intuitively speaking, DNN(Deep Neural Network) as an artificial neural network will also have this feature. At present, there has been a lot of work devoted to the interpretation of the DNN training process. There are many techniques to visualize the attention of DNN, such as CAM, Grad-CAM, Grad-CAM++[12]. Therefore, we can easily obtain the model's class attention map for the image by using these techniques, and only perform data augmentation on the target region. In this way, we can ensure that the crucial features of the image are effectively changed to increase the transferability of adversarial examples. In this article, we choose Grad-CAM++. For an input image, we can get the class activation map for the label under a trained model. Then, use bilinear upsampling to resize it to the size of the original input image.

3.2. Cam-guided data augmentation attack method

Specifically, suppose the original input image of model f is X, its height and width are H_x, W_x. The generated class activation map is X_{cam}, and its height and width are H_c, W_c. Considering that the value distribution of each image's cam is different, we propose an adaptive threshold calculation method, which can calculate a more appropriate threshold according to each input image. Then, binarize the cam according to the threshold, record the minimum and maximum indexes H_{cmin}, H_{cmax}, W_{cmin}, W_{cmax} of non-zero values, and perform cropping to get the cropped input X_{crop} based on these indexes, as shown in Fig.2. Besides, considering that directly cropping may cut off some edge information of the target object, our work adds a buffer region outside the cropped region to solve this problem.

Figure.2 comparison of original image, cam-guided cropped image and mask-generation image (from left to right).

We define the above operation of obtaining X_{crop} as the function $CC(\mathrm{X})$, where CC is the abbreviation of cam-guided cropping, and the gradient of loss function as $\nabla_X L_m(X, y^{gt}; \theta)$, where L_m represents the loss function of the model, y^{gt} represents the grand truth label, and θ represents the parameters of the model. We replace the input X of DI^{2}FGSM with $CC(\mathrm{X})$ and replace the 0 padding

in DI^{2}FGSM with the value of non-target region. The general form of its iterative update formula is as following, where L represents the loss function of our work

$$X_{i+1}^{adv} = Clip_X^{\epsilon}\left\{X_i^{adv} + \alpha \cdot sign\left(\nabla_X L\left(T\left(CC\left(X_i^{adv}\right); p\right), y^{gt}; \theta\right)\right)\right\} \quad (1)$$

3.3. Loss function design

We observe that the perturbations of our crafted adversarial examples will only be distributed in the cropped region. To make the crafted adversarial examples smoother, we propose the mask smoothing loss, thereby improving the invisibility of the crafted adversarial examples. Specifically, if the adversarial example is X_{adv}, the mask smoothing loss can be defined as:

$$L_s = \sum_{mask} \|U_x - x\|_2^2 \quad (2)$$

where x represents the pixel value at any position in the mask region, and U_x represents a set of pixels adjacent to x. And the loss function of our work can be defined as the following, where γ and β are the hyper parameters, they are used to compromise these two losses.

$$L = -L_m\left(T\left(CC(X_{adv})\right)\right) - \gamma \cdot L_m(X_{adv}) + \beta \cdot L_s \quad (3)$$

4. Experiment

We have taken a lot of experiments to confirm that our proposed work can indeed improve the transferability of adversarial examples. And because of the different data sets, we also need to reproduce the latest adversarial attack methods such as FGSM(Fast Gradient Sign Method)[2], MI(Momentum Iterative)-FGSM[7], and DI^2(Diverse Inputs Iterative)-FGSM[9], which we have briefly introduced them in the Section 2.1 and 2.2.

4.1. Dataset

What we are concerned about is how to make highly transferable adversarial examples to make the predictions of model wrong. Therefore, examples that are originally classified incorrectly are almost meaningless to us. Therefore, we select 1243 images correctly classified by our models from the ImageNet validation set as our test dataset. And all images have been resized to 299×299×3.

4.2. Networks

We consider 4 normally trained networks, such as resnet101, Inception v3, Inception v4, Inception-Resnet-v2, to craft adversarial examples, and then based on the above 4 networks and 1 adversarial trained network, such as adv-denoise-resnet152, we evaluate the attack success rates.

4.3. Implement Details

The experimental environment is based on the Ubuntu 16.04 operating system, the CPU is Intel i7-9700, the GPU is 2 NVIDIA RTX 2080, and the running memory is 11G. We first follow some default settings of DI^2-FGSM. The difference is that, we set the weight decay to 0, and we choose $\gamma = 0.9$ and $\beta = 0.1$ after experiments. We set the maximum perturbation of each pixel $\epsilon = 32/255.0$.

4.4. Comparison

We compare our method with latest methods in same settings and same dataset. The results are shown in the table 1. Our method can achieve the best attack success rates when attacks the normally trained network, and for the adversarial trained networks, our method can also achieve comparable results.

Table1. The attack success rates on five different networks compared with latest methods.

Model	Attack	Res-101	Inc-v3	Inc-v4	IncRes-v2	Res-152$_{adv\text{-}denoise}$
Res-101	FGSM[2]	29.59%	68.44%	9.18%	9.10%	37.09%
	MI-FGSM[7]	80.98%	**100.00%**	60.08%	51.56%	39.23%
	DI^2-FGSM[9]	93.85%	**100.00%**	91.64%	80.25%	**40.95%**
	Ours	**95.16%**	**100.00%**	**94.02%**	**83.03%**	40.54%
Inc-v3	FGSM	46.07%	83.03%	16.23%	15.08%	**39.06%**
	MI-FGSM	**100.00%**	96.80%	81.89%	81.56%	38.49%
	DI^2-FGSM	**100.00%**	97.54%	97.31%	**90.49%**	38.49%
	Ours	**100.00%**	**97.76%**	**92.70%**	88.85%	38.33%
IncRes-v2	FGSM	55.00%	87.70%	27.13%	30.90%	**41.78%**
	MI-FGSM	78.36%	94.10%	64.84%	**100.00%**	38.16%
	DI^2-FGSM	**91.97%**	96.72%	88.03%	**100.00%**	39.23%
	Ours	90.66%	**96.89%**	**88.62%**	**100.00%**	38.82%

The comparison results are shown in Table 1. We can see: 1) the attack success rates of our method under normally trained networks are better than latest methods, which proves that our method is more effective than the random transformation and can improve the transferability of adversarial examples. 2) Even in the defensive networks, our method can achieve good attack success rates.

5. Conclusion

In this article, we propose to improve transferability of adversarial examples with a more efficient data augmentation. Specially, our method applies a cam(class activation map)-guided data augmentation attack method to improve the efficiency of creating diverse inputs, and propose a mask smooth function to improve the invisibility of the adversarial examples. Compared with the latest method, the results in ImageNet show that our method can get better attack success rates. In a word, our method can improve the transferability of adversarial examples. Therefore, our work can craft more transferable adversarial examples, which can serve as a strong benchmark when researching defensive methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China: Research on the visual perception enhancement method of unmanned surface ships in severe navigable environment.

References

[1] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. https://arxiv.org/abs/1312.6199.
[2] I. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. https://arxiv.org/abs/1412.6572.
[3] Kurakin, A., Goodfellow, I.J., & Bengio, S. (2017). Adversarial examples in the physical world. https://arxiv.org/abs/1607.02533.
[4] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. https://arxiv.org/abs/1706.06083.
[5] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. In: 2016 IEEE European symposium on security and privacy (EuroS&P). Saarbruecken. pp. 372-387.
[6] Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In: 2017 IEEE symposium on security and privacy. San Jose. pp. 39-57.

[7] Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., & Li, J. (2018). Boosting adversarial attacks with momentum. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). Salt Lake. pp. 9185-9193.
[8] Yin, H., Zhang, H., Wang, J., & Dou, R. (2020). Improving the Transferability of Adversarial Examples with the Adam Optimizer. https://arxiv.org/abs/2012.00567.
[9] Xie, C., Zhang, Z., Zhou, Y., Bai, S., Wang, J., Ren, Z., & Yuille, A. L. (2019). Improving transferability of adversarial examples with input diversity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach. pp. 2730-2739.
[10] Wu, W., Su, Y., Chen, X., Zhao, S., King, I., Lyu, M. R., & Tai, Y. W. (2020). Boosting the transferability of adversarial samples via attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Online. pp. 1161-1170.
[11] Hu, T., Qi, H., Huang, Q., & Lu, Y. (2019). See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification. https://arxiv.org/abs/1901.09891.
[12] Chattopadhay, A., Sarkar, A., Howlader, P., & Balasubramanian, V. N. (2018). Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE winter conference on applications of computer vision (WACV). Lake Tahoe. pp. 839-847.