

Adversarial Attack Defense Based on the Deep Image Prior Network



Richard Evan Sutanto  and Sukho Lee

Abstract Several recent studies have shown that artificial intelligence (AI) systems can be malfunctioned by deliberately crafted data entering through the normal route. For example, a well-crafted sticker attached on a traffic sign can lead a self-driving car to misinterpret the meaning of a traffic sign from its original one. Such deliberately crafted data which cause the AI system to misjudge are called adversarial examples. The problem is that current AI systems are not stable enough to defend adversarial examples when an attacker uses them as means to attack an AI system. Therefore, nowadays, many researches on detecting and removing adversarial examples are under way. In this paper, we proposed the use of the deep image prior (DIP) as a defense method against adversarial examples using only the adversarial noisy image. This is in contrast with other neural network based adversarial noise removal methods where many adversarial noisy and true images have to be used for the training of the neural network. Experimental results show the validness of the proposed approach.

Keywords Adversarial noise · Deep image prior · Neural network

1 Introduction

Deep learning prevails now in many applications such as smart cars, person identification systems through a security cameras, guiding systems for disabled persons, voice/image generators using different kinds of inputs, etc. The decision that the AI system makes is critical in some applications such as the smart car system, where the system uses an image as input and process it through a network to do the decision-making task [1]. Unfortunately, some researches show that deep learning methods are sensitive to noise called adversarial noise, and the data which include adversarial noises are called adversarial examples. Adversarial examples perturb the input by small scaled patterns, so that the difference is not perceptible to human eyes but affects the performance of the neural network [3]. In the object detection case, the

R. E. Sutanto · S. Lee (✉)
Dongseo University, Busan 47011, Korea
e-mail: petrasuk@gmail.com

© Springer Nature Singapore Pte Ltd. 2020
K. J. Kim and H.-Y. Kim (eds.), *Information Science and Applications*,
Lecture Notes in Electrical Engineering 621,
https://doi.org/10.1007/978-981-15-1465-4_51

small adversarial noise (inside the adversarial example in images) leads the classifier to miss-classify the object to a wrong label. This could cause a disaster, e.g., a traffic accident of a smart car as it misinterprets a traffic sign [2]. There also exists the threat that the adversarial examples can be used as an attacking tool to the AI system. To defend such kind of attacks, effective defense methods have to be used.

Defense methods can be categorized according to the approach they use for defense. The first approach is to enhance the performance of the classifier so that it becomes robust against adversarial examples [8]. Adversarial training [3] belongs to this category, which uses adversarial images as an additional training set to provide the classifier with extra knowledge about adversarial examples. Another approach converts the one-hot labels for label smoothing [10] so that the labels are not so sensitive to the adversarial noise. Yet another approach is the direct denoising of the adversarial noise. For example, in [6] the defense GAN (Generative Adversarial Network) is used to filter the adversarial noise from the input image. However, the above mentioned methods all use extra adversarial noisy data to train the network. This makes the network robust against some kinds of adversarial noise but may be still weak against some adversarial examples that are not used in the training of the network. Furthermore, the training needs many adversarial examples of as many as possible types. In this paper, we propose the use of the deep image prior as a defense method which eliminates the adversarial noise using only the input image alone. Experimental results show the validity of the proposed method.

2 Preliminaries

In this section, to understand the proposed approach, we first explain the concept of the adversarial attack, and then introduce the deep image prior (DIP) network. After that, we propose in the next section the use of DIP network for defending an adversarial attack.

2.1 *Adversarial Attack on Neural Networks*

An adversarial noise is a carefully designed small perturbation which when added to the original input to the network can lead the neural network to make a false decision. When used as a tool of attack, the adversarial example can arouse serious critical harms to the system which depends on the decision of the neural network. Figure 1 shows an example of the adversarial example. Even though the noise added to the image is small so that in the eye of the human the original image and the adversarial example look still similar, the neural network gives different decisions to the two images.

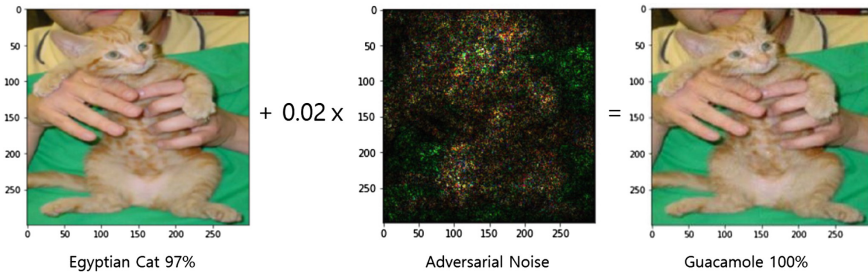


Fig. 1 Example of adversarial example. A small adversarial noise added to the original image can make the neural network to classify the image as a Guacamole instead of an Egyptian cat

There are many ways that an adversarial example can be generated. In [3], a simple method is proposed which generates the adversarial noise by moving the image in the direction which increases the distance between the true label and the output of the neural network the most:

$$\hat{x} = x + \epsilon \cdot \text{sign}(\nabla_x J(x, y_{true})). \tag{1}$$

Here, x refers to the input image, y_{true} is the true label of the input image, ∇_x is the gradient with respect to x , and ϵ is a small positive value. An alternative to (1) is

$$\hat{x} = x - \epsilon \cdot \text{sign}(\nabla_x J(x, y_{fool})). \tag{2}$$

which aims to decrease the distance between the false label and the output of the neural network, so that the network falsely classifies the image as the targeted false label. In [4], Kurakin et al. propose an iterative scheme which clips all the pixels after each iteration:

$$\begin{aligned} \hat{x} &= x, \\ \hat{x}_{N+1} &= \text{Clip}_{x,\epsilon}\{\hat{x}_{N+1} + \alpha \cdot \text{sign}(\nabla_x J(x_N, y_{true}))\}. \end{aligned} \tag{3}$$

where

$$\text{Clip}_{x,\epsilon}\{x'\} = \min\{255, x(i, j, k) + \epsilon, \max\{0, x(i, j, k) - \epsilon, x'(i, j, k)\}\}, \tag{4}$$

where $x(i, j, k)$ is the value at the position (i, j) in the k 's channel, and the clipping function $\text{Clip}_{x,\epsilon}\{x'\}$ keeps x' stay inside the ball with radius ϵ with the original image x as the center.

2.2 Previous Works of Using Deep Neural Networks for Adversarial Noise Removal

In [5], a method for adversarial noise removal based on the high-level representation has been introduced which uses the well-known Unet architecture. Since the adversarial noise has a very small noise level, normally, a pixel-wise loss function is not enough to eliminate the noise. Therefore, in [5], a high-level representation loss function is proposed,

$$L = \sum_i \|f(x_i) - f(d_\theta(\hat{x}_i))\|^2, \quad (5)$$

where x_i is the original image, \hat{x}_i is the adversarial noisy image, $d_\theta(\cdot)$ is the denoising network which we want to train, and $f(\cdot)$ refers to the output of the classifier neural network which we want to defend from the adversarial noise. The parameters θ of the network $d_\theta(\cdot)$ are trained to minimize the loss function in (5) with a large number of pairs of (x_i, \hat{x}_i) , $i = 1, 2, \dots, N$. Therefore, this kind of network needs many adversarial noisy images and their counterpart true images to be trained. In contrast, we propose the use of the deep image prior network which can be trained by using only the adversarial noisy image.

2.3 Deep Image Prior Network

The work in [9] proposes a deep image prior (DIP) network which converts a random noisy vector z into a restored image $g_\theta(z)$, where $g_\theta(\cdot)$ denotes the deep image prior network with parameter θ . As shown in the experiments in [9], the deep image prior (DIP) network has a high impedance against noise. Therefore, during the training of the DIP network by minimizing (8), the trajectory of the parameter θ passes through a good solution θ^* which results in a well denoised image $g_\theta(z)$. It should be taken care of that θ^* is not the minimizer of (8), since the image using the minimizer will result in the noisy image again. The parameters are obtained by the way of minimizing the following energy functional:

$$\min_{\theta} E(g_\theta(z); \hat{x}) \quad (6)$$

where $E(\cdot)$ is often set to the square of the L_2 norm:

$$E(x; x_0) = \|x - x_0\|^2, \quad (7)$$

then (6) becomes

$$\min_{\theta} \|g_\theta(z) - x_0\|^2. \quad (8)$$

The restored image x^* is obtained by

$$x^* = g_{\theta} \cdot (z), \tag{9}$$

where θ^* denotes the parameters of the network which are obtained in the way of minimizing $E(g_{\theta}(z); \hat{x})$.

3 Adversarial Example Defense Based on the DIP

In this section, we propose a defense method against the adversarial example via the deep image prior network. Unlike other deep neural network using adversarial elimination methods, the defense method do not need a neural network to be trained with many adversarial and true image pairs, but can eliminate the adversarial noise using only the adversarial noisy image. The main idea is that the projection of the adversarial noisy image onto the deep image prior space will eliminate the adversarial noise.

Let I_{in} be the input we want to put through the target classifier network f . Here, I_{in} may or may not contain the adversarial noise. Before putting I_{in} through the network, we first put I_{in} through the deep image prior g_{θ} , where the parameters θ get updated by minimizing the following loss function,

$$L = \|I_{in} - g_{\theta}(I_{in})\|^2, \tag{10}$$

The deep image prior network is trained on a single image I_{in} , which is different from other networks which are trained on many images. Let x_o be the initial input to the deep image prior network, and let the input be the image which want to put through the target classifier neural network, i.e., let $x_o = I_{in}$. Let $\hat{x}_0, \hat{x}_1, \dots, \hat{x}_t$ be the outputs of the DIP according to the updates in the training. After each iteration, the update in the parameters θ will add new high frequency components $g(\Delta\theta_{k+1}, x_0)$ to $g(\theta_k, x_0)$ as can be seen in Fig. 2.

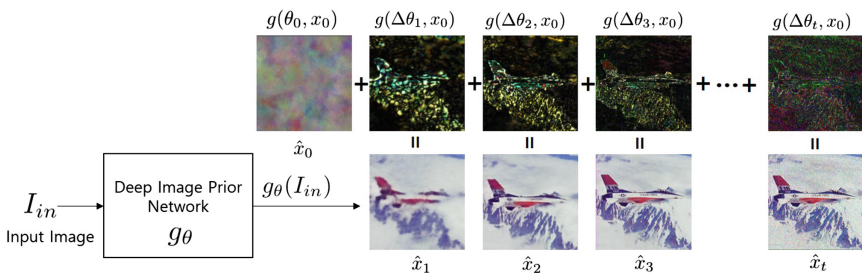


Fig. 2 Construction of the noiseless image with the deep image prior network

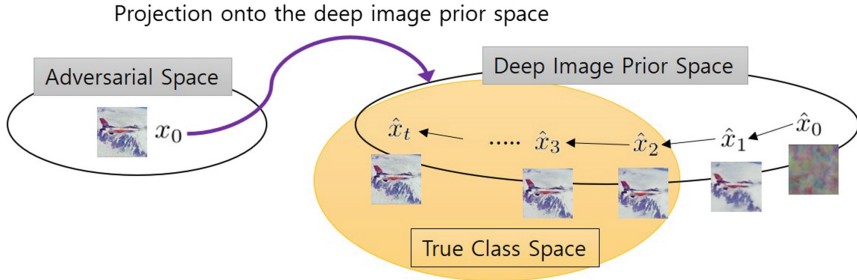


Fig. 3 Projection onto the true class space

We interpret the output of the deep image prior network as a projected version of the input image on the deep image prior space (Fig. 3). That is, the output of the deep image prior network is an approximation of the input image constrained by the deep image prior. The early outputs $g_\theta(I_{in})$ lack many high frequency components so that $f(g_\theta(I_{in})) \neq f(I_{true})$, where I_{true} denotes the image without any adversarial noise. However, after a sufficient update, $g_\theta(I_{in})$ approaches I_{in} in the L_2 norm sense, but remains in the true class space which does not include the adversarial noise. When putting $g_\theta(I_{in})$ through the classifier f , this will give the correct, or at least, a similar classification result as I_{in} , i.e., $f(I_{in}) \approx f(g_\theta(I_{in}))$.

4 Experimental Results

In order to examine performance of the algorithm, we do some experiments using the pre-trained Inception-V3 [7] model. We test the defense system on clean images as well as adversarial images, to verify that the DIP does not change the high frequency components in the direction such that the classification result changes. We used 100 cat images (of size of 299 by 299, RGB color) and used the targeted fast gradient sign method (FGSM) shown in (2) for the attacking method. We used two different settings for the ϵ value, where dataset A is made with $\epsilon = 0.008$, and dataset B with $\epsilon = 0.08$. Figure 4 shows the original image, the adversarial image, the difference between the original and the adversarial image, the image denoised with the DIP, and the difference between the denoised image and the original image.



Fig. 4 Example figure

Table 1 Accuracy results (%) of the CNN classifier (Inception-V3) before/after applying the DIP

Dataset (noise level)	Before applying DIP	After applying DIP
Original dataset	95	90
Dataset A (epsilon 0.008)	1	86
Dataset B (epsilon 0.08)	1	87

and the difference between the adversarial and the denoised image, respectively. It can be observed that the original and the adversarial image cannot be distinguished with the eye. However, the difference image shows certain patterns which leads to the miss-classification. The difference image is multiplied by 200 for visualization. As can be seen the denoised image also shows some difference with the original image, but the difference image shows no adversarial pattern but only difference in the edge regions. Therefore, it can be said that the adversarial pattern is eliminated in the denoised image.

In Table 1, we examine the classifier accuracy of the adversarial images, clean images, and the DIP reconstructed images. We reconstructed also the clean images with the DIP and measured the accuracy to verify that the reconstructed images show the same classification result as the clean ones. The classification accuracy result of original clean images before applying the DIP is about 95%, while the classification accuracy for the adversarial images has dropped down to 1%. After applying the DIP based denoising to the adversarial image, the classification accuracy recovers up to 87%, showing that the DIP defends the adversarial examples.

5 Conclusion

We proposed the use of the deep image prior network to remove and detect the adversarial noise embedded in the image. The deep image prior network makes it possible to denoise the adversarial noise using only the adversarial noisy image. One major drawback of the proposed method is that the training of the deep image prior network has to be done for every incoming image. However, the number of iterations in the training is small enough to make it work in real time when combined with multi-core GPU systems. Issues on how to accelerate the speed of the proposed defense system are topics for further study.

Acknowledgements This work was supported by Institute for Information and Communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2018-0-00245, Development of prevention technology against AI dysfunction induced by deception attack) and the Dongseo University Research Fund of 2018.

References

1. Badue C, Guidolini R, Carneiro RV, Azevedo P, Cardoso VB, Forechi A, Jesus LFR, Berriel RF, Paix ao TM, Mutz FW, Oliveira-Santos T, de Souza AF (2019) Self-driving cars: a survey. CoRR abs/1901.04407. <http://arxiv.org/abs/1901.04407>
2. Boloor A, He X, Gill CD, Vorobeychik Y, Zhang X (2019) Simple physical adversarial examples against end-to-end autonomous driving models. CoRR abs/1903.05157. <http://arxiv.org/abs/1903.05157>
3. Goodfellow IJ, Shlens J, Szegedy C (2015) Explaining and harnessing adversarial examples. In: 3rd International Conference on Learning Representations, ICLR2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings. <http://arxiv.org/abs/1412.6572>
4. Kurakin A, Goodfellow IJ, Bengio S (2016) Adversarial examples in the physical world. CoRRabs/1607.02533. <http://arxiv.org/abs/1607.02533>
5. Liao F, Liang M, Dong Y, Pang T, Zhu J, Hu X (2018) Defense against adversarial attacks using high-level representation guided denoiser. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1778–1787 (June 2018). <https://doi.org/10.1109/CVPR.2018.00191>
6. Samangouei P, Kabkab M, Chellappa R (2018) Defense-gan: protecting classifiers against adversarial attacks using generative models. CoRRabs/1805.06605. <http://arxiv.org/abs/1805.06605>
7. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2015) Rethinking the inception architecture for computer vision. CoRRabs/1512.00567. <http://arxiv.org/abs/1512.00567>
8. Tian S, Yang G, Cai Y (2018) Detecting adversarial examples through image transformation. <https://aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17408>
9. Ulyanov D, Vedaldi A, Lempitsky V (2018) Deep image prior. CoRRabs/1711.10925. <https://arxiv.org/abs/1711.10925>
10. Warde-Farley D (2016) 1 adversarial perturbations of deep neural networks