Journal of Computer and Communications, 2023, 11, 14-30
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

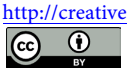
Black Box Adversarial Defense Based on Image Denoising and Pix2Pix

Zhenyong Rui, Xiugang Gong

School of Computer Science and Technology, Shandong University of Technology, Zibo, China
Email: 13056316310@163.com

How to cite this paper: Rui, Z.Y. and Gong, X.G. (2023) Black Box Adversarial Defense Based on Image Denoising and Pix2Pix. *Journal of Computer and Communications*, **11**, 14-30.
https://doi.org/10.4236/jcc.2023.1112002

Received: November 14, 2023
Accepted: December 16, 2023
Published: December 19, 2023



Abstract

Deep Neural Networks (DNN) are widely utilized due to their outstanding performance, but the susceptibility to adversarial attacks poses significant security risks, making adversarial defense research crucial in the field of AI security. Currently, robustness defense techniques for models often rely on adversarial training, a method that tends to only defend against specific types of attacks and lacks strong generalization. In response to this challenge, this paper proposes a black-box defense method based on Image Denoising and Pix2Pix (IDP) technology. This method does not require prior knowledge of the specific attack type and eliminates the need for cumbersome adversarial training. When making predictions on unknown samples, the IDP method first undergoes denoising processing, followed by inputting the processed image into a trained Pix2Pix model for image transformation. Finally, the image generated by Pix2Pix is input into the classification model for prediction. This versatile defense approach demonstrates excellent defensive performance against common attack methods such as FGSM, I-FGSM, DeepFool, and UPSET, showcasing high flexibility and transferability. In summary, the IDP method introduces new perspectives and possibilities for adversarial sample defense, alleviating the limitations of traditional adversarial training methods and enhancing the overall robustness of models.

Keywords

Deep Neural Networks (DNN), Adversarial attack, Adversarial Training, Fourier Transform, Robust Defense

1. Introduction

With the development of artificial intelligence and the improvement of hardware capabilities, deep neural networks have found extensive applications in fields such

as image recognition, object tracking, language translation, and more [1]. However, the existence of adversarial examples, as proposed by Szegedy *et al.*, poses a significant threat to the security of neural networks and AI due to their stealthiness and adversarial nature. There have been reports confirming the serious consequences of adversarial examples in the field of autonomous driving. Therefore, research on defense techniques against adversarial examples has become critically important.

Building upon the research and contributions of Pedro *et al.* [2] and Goodfellow *et al.* [3] in the study of adversarial examples, various adversarial attack methods have been continually developed, and corresponding defense methods have emerged. These defense methods can be broadly categorized into two groups: those based on input transformation and those based on robust optimization. Input transformation methods typically involve preprocessing the input image before feeding it into the prediction model to remove adversarial perturbations as much as possible. Common preprocessing techniques include image denoising, JPEG compression [4], image restoration, PCA dimensionality reduction, and more. These methods aim to recover the important information from uncontaminated regions of the image.

On the other hand, robust optimization-based methods primarily encompass adversarial training and model compression. Adversarial training involves incorporating adversarial examples during model training to enhance the model's robustness against such examples, thereby improving its generalization capability. Model compression, on the other hand, reduces the complexity of the model to enhance its robustness.

However, due to the varying noise distributions introduced by different adversarial attack methods, a single defense method often struggles to withstand a multitude of adversarial attacks. Thus, improving defense flexibility according to different models and attack methods to achieve robust generalization remains a significant research challenge [5].

This article focuses on the issue of adversarial attacks faced by deep neural networks (DNNs), and we note that traditional defense methods have limitations in terms of their robustness to multiple attack types. That is, they require excessive information about the parameters of adversarial attacks and are difficult to defend against various attacks. To address this challenge, we propose a novel black-box defense method based on image denoising and Pix2Pix technology (IDP). The contributions of this research are as follows:

1) This research innovatively combines image denoising with Pix2Pix models to enhance the similarity of feature distributions between adversarial samples and original samples, thereby improving the model's robustness.

2) Unlike traditional defense methods, the IDP method does not require cumbersome adversarial training, nor does it require knowledge of the attack type. It has better generalization performance and a broader range of application scenarios.

2. Related Work

2.1. Related Concepts and Attack Methods of Adversarial Samples

Extensive experimental results have shown that existing classification models, despite being trained on large amounts of data and achieving excellent performance on existing datasets, can still be highly susceptible to "misleading" adversarial examples, resulting in misclassification. These images that cause the models to make incorrect predictions are referred to as adversarial examples [6]. One of the most common attack methods, FGSM (Fast Gradient Sign Method), was proposed by Goodfellow *et al.* FGSM utilizes gradient information to quickly compute adversarial perturbations. This method has been extensively studied by researchers. Subsequently, Seyed-Mohsen *et al.* [7] introduced the DeepFool adversarial example generation method, which estimates the distance between input samples and the decision boundary of the classifier. This distance is used as the minimal perturbation to be generated, providing stronger adaptability [8].

1) FGSM (Fast Gradient Sign Attack) is a classic method that rapidly generates adversarial examples by leveraging gradient information. It belongs to the category of untargeted attacks, where the attack is considered successful as long as the predicted result is not the true class of the sample. In traditional optimization, we move in the opposite direction of the gradient to minimize the loss function, known as gradient descent. However, FGSM is designed to maximize the loss function, causing the classification result to differ from the true class. It can be seen as gradient ascent.

2) I-FGSM (Iterative Fast Gradient Sign Method) is an improved version of FGSM that increases the attack success rate by iteratively applying FGSM [9]. In each iteration, FGSM perturbations are added to the original sample, while constraining the magnitude of the perturbation within a certain range. I-FGSM typically achieves higher attack success rates compared to FGSM.

3) DeepFool attack is a method specifically designed to address the limitations of FGSM in non-linear models. It is based on a hyperplane classification approach and exhibits greater adaptability. The DeepFool algorithm iteratively computes the minimal perturbation that minimizes the distance from the sample to the decision boundary, gradually moving the sample towards the boundary until it is misclassified by the classifier.

4) UPSET (Universal Perturbations for Steering to Exact Targets) is a subset estimation-based black-box adversarial attack method [10]. It calculates a universal perturbation over the entire dataset and then adds this perturbation to each individual sample to attack the model.

2.2. Adversarial Defense Method

With the existence of adversarial examples being proven, research on defending against adversarial examples has gradually garnered attention. Papernot *et al.* [11] proposed defensive distillation, which reduces the sensitivity of neural networks to adversarial perturbations by smoothing the model through distilling

extracted knowledge during the training process. Ross *et al.* [12] introduced input gradient regularization, penalizing the magnitude of output changes when the input varies, to enhance the model's robustness for defense. Cisse *et al.* [13] presented ParsEvalNetwork, which adds regularization constraints to the model's weights to reduce the gradient of the model's output with respect to the input, thereby improving the model's robustness for defense. Currently, adversarial defense methods can be classified into two major categories: those targeting the image data itself and those targeting the network model. The former includes techniques such as image denoising and image compression, while the latter encompasses a series of methods that utilize gradient information for defense. However, the former requires prior knowledge of the attack types to achieve strong defense, and the latter relies on substantial computation. As a result, methods based on adversarial training have emerged. Adversarial training involves training neural networks with both adversarial examples and normal samples, aiming to capture the feature distribution of adversarial examples and enhance the model's robustness. However, as research progresses, flaws in adversarial training have also been discovered. The performance and classification "bias" of neural networks often heavily rely on the features of the dataset. If the added adversarial examples are generated using specific attack methods, the trained model's classification "bias" will lean towards the feature distribution of these attack-generated adversarial examples. This sensitivity makes the model highly susceptible to slight modifications, leading to a strong dependence of the performance of adversarial training-based models on the types and quantities of adversarial attack methods used. Therefore, it is a research question worth exploring how to improve the flexibility of defense techniques to adapt to different models and attack methods.

3. Approach

In response to the issue of reduced flexibility and weak transferability associated with some defense methods that require extensive adversarial training tailored to specific attack methods, we propose a black-box adversarial defense approach that combines image denoising with Pix2Pix. Since our method is a black-box defense technique, we do not possess knowledge about the specific attack type or any parameters associated with the adversarial samples.

In this paper, we take the original training samples X and feed them into an image denoising model A, resulting in denoised image samples X'. These denoised samples X' are then used as inputs to train a Pix2Pix model, while the original samples X serve as the target samples for the Pix2Pix model. Subsequently, we input the X' samples into the trained Pix2Pix model, obtaining the corresponding output samples Y. Finally, we employ these Y samples as training data for training a resnet50 [14] classification model, ultimately yielding the final defense model.

The process of image denoising model A is as follows:

S1: Firstly, perform an operation based on interpolation enlargement on the

image samples.

S2: Apply Fourier transform to the images obtained in S1 to obtain their corresponding frequency spectrum images.

S3: Perform Wiener filtering on the frequency spectrum images obtained in S2, followed by inverse Fourier transform to obtain the final denoised images.

When testing unknown samples, the same process is followed: the samples are first transformed through the IDP model before being input into the classification model. This is done to maximize the similarity in feature distribution between the images predicted by the model and the training samples.

3.1. Image Denoising Model A

3.1.1. Fourier Transform

The Fourier transform is the process of converting a time-domain signal into a frequency-domain signal [15]. For one-dimensional space, any periodic signal can be composed of a combination of sine waves with different phases and amplitudes. In the case of a two-dimensional image, the grayscale value of each pixel can be understood as the "amplitude" in one-dimensional space. Therefore, in the two-dimensional domain, it can be seen as the superposition of countless two-dimensional plane waves. Fourier transform can be used to decompose an image into its constituent parts of different frequencies, enabling image enhancement, filtering, compression, and other functionalities. In image denoising, the two-dimensional discrete Fourier transform is widely applied. Its expression is shown below:

$$F(u,v)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\exp\left[-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)\right]. \tag{1}$$

In the above Equation (1), u and v are the frequency variables corresponding to the x and y axes respectively, M and N are the height and width corresponding to the function $f(x, y)$ respectively, and $F(u, v)$ is the frequency spectrum of image $f(x, y)$. Fourier transform also has inverse transformation, that is, to convert the spectrum $F(u, v)$ into an image [16], the expression is as follows:

$$f(x,y)=\frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}F(u,v)\mathrm{e}^{\left[-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)\right]}. \tag{2}$$

In Equation (2), $x=0,1,2,\cdots,M-1$, $y=0,1,2,\cdots,N-1$.

The spectrum and energy spectrum of the two-dimensional discrete Fourier transform are shown in Equations (3) and (4) respectively. R represents the real part of the two-dimensional discrete Fourier transform coefficient, and I represents the imaginary part:

$$\left|F(u,v)\right|=\left[R^2(u,v)+I^2(u,v)\right]^{1/2}. \tag{3}$$

$$E(u,v)=R^2(u,v)+I^2(u,v). \tag{4}$$

After Fourier transform, the image often needs to go through a frequency centralization step, because most of the effective information of the image is concentrated in the low frequency part; The formation of low frequency in the

periphery, high frequency in the center of the distribution; The image can be easily restored after Fourier transform and frequency centralization.

3.1.2. Interpolation Based Amplification Operations

Since both the MNIST handwritten digit dataset and the CIFAR10 dataset consist of low-resolution images, comparing the spectrum obtained from directly applying the DFT transformation to the original low-resolution images with the spectrum obtained after enlarging the images, the latter contains more pixels that can be used for classification. As the MNIST dataset consists of single-channel images, the DFT transformation only needs to be applied to a single channel. However, for CIFAR10, the transformation needs to be applied to all three channels (R, G, and B). Here, we select single-channel images from the MNIST dataset to illustrate the differences in the corresponding spectra under different operations. The left side shows the adversarial samples and their corresponding spectra.

Figure 1 shows the two images of the digit "2". On the left is the original image with a size of 28 × 28 pixels, and on the right is the image after being enlarged to a size of 128 × 128 pixels. Although the visually perceived difference between the enlarged adversarial sample and the original adversarial sample may be minimal to the human eye, it is evident that the spectrum of the enlarged image, after undergoing the DFT transformation, contains more detailed information and less interference. The interpolation-based enlargement operation can be seen as a means of super-resolution, transforming the low-resolution image into a higher-resolution one. As a result, the enlarged image exhibits smoother texture and sharper lines, which is reflected in the clearer spectrum. Therefore, incorporating the operation of enlarging these paired images is necessary.

3.1.3. Wiener Filtering

Wiener Filtering is a signal processing technique used to estimate or recover a signal that has been corrupted by noise. Its primary objective is to model the signal and noise in the frequency domain and attempt to minimize the estimation error to restore the original signal. In Wiener Filtering, the signal is assumed to be a linear combination of the original signal and noise. This model can be represented as:

$$y(t) = s(t) + n(t). \tag{5}$$

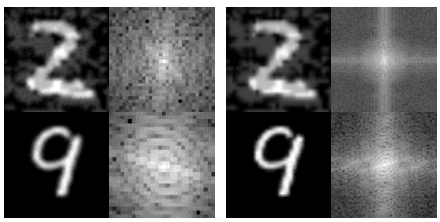


Figure 1. The spectrum diagram corresponding to the adversarial sample and the adversarial sample after adding the amplification operation.

Here, $y(t)$ is the observed signal, $s(t)$ is the original signal, and $n(t)$ is the noise.

Wiener Filtering typically operates in the frequency domain because the spectra of the signal and noise are often easier to model. The signal, noise, and observed signal can all be represented in the frequency domain. Wiener Filtering aims to determine the estimate by minimizing the Mean Square Error [17], which is the squared difference between the estimated signal and the original signal. This criterion is usually expressed as:

$$\hat{s}(t) = \arg\min_E \left[\left| s(t) - \hat{s}(t)^2 \right| \right]. \tag{6}$$

where E represents the expectation operation. The objective of this criterion is to find a result that minimizes the expected value of the mean square error, and Wiener Filtering is capable of addressing the issue of boundary blurring more effectively compared to other linear filters [18].

3.1.4. Analysis of the Advantages of Combining Fourier Transform and Wiener Filtering for Denoising

1) Frequency domain analysis: Fourier Transform converts the signal from the time domain to the frequency domain, allowing for a better understanding of the noise components in the signal. By analyzing the frequency characteristics of the signal and noise, we can selectively remove noise in the frequency domain while preserving the useful information of the signal.

2) Noise characterization: By incorporating Wiener Filtering in the frequency domain, it is possible to target the removal of noise within specific frequency ranges [19]. Through Fourier Transform, we can obtain the spectral information of the signal and noise, including their power spectral densities and frequency distributions. Accurate characterization of the noise allows for better guidance in the design of the Wiener filter, aiming to minimize the impact of noise.

3) Minimum mean square error filtering: Wiener Filtering is a minimum mean square error filter that minimizes the mean square error between the filtered signal and the original signal. In the frequency domain, the Wiener filter can be optimized based on the power spectral densities of the signal and noise, as well as the signal-to-noise ratio [20]. Fourier Transform provides the spectral information of the signal and noise, aiding in determining the optimal filter parameters to minimize the error [21].

Overall, the combination of Fourier Transform and Wiener Filtering provides a powerful approach for denoising images. It leverages the frequency domain representation and adaptive filtering capabilities to effectively remove noise while preserving important image features, leading to improved image quality.

3.2. Pix2Pix Model

The Pix2Pix model is a variant of Generative Adversarial Networks (GANs) used for image translation or image-to-image transformation tasks. Introduced by Phillip Isola *et al.* in 2016 [22], this model is designed to translate input images

into corresponding output images. Pix2Pix's primary capability lies in learning to convert one type of image into another, such as turning black and white images into color images or transforming real images into cartoon-style images.

The core concept of the Pix2Pix model involves training through two convolutional neural networks: a generator and a discriminator. The generator attempts to produce images that closely match the expected output, while the discriminator tries to differentiate between the images generated by the generator and the real target images. The generator takes input images and aims to generate images close to the desired output, while the discriminator takes both types of images (those generated by the generator and real target images) and attempts to distinguish between them. This process is a game where the generator continually strives to improve the quality of the generated images, while the discriminator continuously enhances its ability to differentiate, ultimately resulting in the generator producing realistic target images.

The Pix2Pix model excels in many image processing tasks, including semantic segmentation, image translation, transformation between different styles of images, image inpainting, and more. This model finds wide applications across various domains, and its core idea has provided essential inspiration for subsequent tasks in image translation and the development of generative adversarial networks. In summary, Pix2Pix is a deep learning model for image-to-image translation, achieved through the use of generative adversarial networks to perform the transformation task between input images and desired output images.

In this article, the reasons for choosing the Pix2Pix model can be summarized as follows:

1) The Pix2Pix model possesses powerful image transformation capabilities, enabling the conversion of images from one feature distribution to another, while maintaining consistency in size with the original images after transformation.

2) The image denoising model reduces the dissimilarity in feature distribution between adversarial samples and original samples. By combining the Pix2Pix model with image denoising, through the training of the generator and discriminator, the generator is capable of producing images closer to the desired output. This further diminishes the differences between adversarial samples and original samples. This combined strategy contributes to enhancing the robustness of the defense model, thereby better addressing black-box adversarial attacks.

Model Structure

The generator G adopts a Unet structure, which is fundamentally an encoder-decoder architecture. It involves a series of down-sampling convolutional operations followed by up-sampling transposed convolutional operations, ultimately producing the generated image at the output layer. The discriminator consists of four fundamental convolutional blocks. Each block comprises a convolutional operation, batch normalization, and a LeakyReLU activation function.

The network architecture corresponding to this description is depicted in the following diagram.

Figure 2 shows the constituent architecture of the Pix2Pix model. The generator takes one image as input and produces another image as output. The role of the discriminator is to score the input image and determine its authenticity. With an ample number of training samples and under the assumption of model convergence, once the training is complete, the model can achieve the transformation of one image A into another image B.

4. Experimental Results

4.1. Experimental Platform and Data Set

Experimental Platform. This study conducted experiments on an AMD R9-5900HX CPU and an NVIDIA GeForce RTX 3080 laptop GPU (16 GB), equipped with 32 GB of RAM. The proposed method was implemented using the open-source machine learning framework PyTorch.

Datasets. Two datasets were selected for this study, namely the MNIST handwritten digit dataset and the CIFAR-10 dataset. The MNIST dataset consists of black-background, white-digit images with a resolution of 28 × 28 pixels. The CIFAR-10 dataset contains RGB three-channel color images with a resolution of 32 × 32 pixels. Both datasets consist of ten distinct classes. The selected adversarial attack methods include FGSM, I-FGSM, DeepFool, and UPSET. A ResNet-50 classification model was employed for evaluation, with defense success rate as the performance metric.

4.2. Attack and Training Settings

Attack setting: For FGSM adversarial attack, although high disturbance intensity has strong attack effect, it is accompanied by a sharp decline in the overall picture quality. In order to give consideration to both picture quality and attack performance, the disturbance intensity is set at 0.20. For I-FGSM attack, the attack disturbance is also set to 0.20 and epochs to 10. For the DeepFool attack, the step length overshoot is set to 0.4 and the number of iterations max_iter is set to 50; For UPSET adversarial attacks, step size eta is set to 0.1, algorithm iteration number max_iter is set to 20, and disturbance size epsilon is set to 0.1.

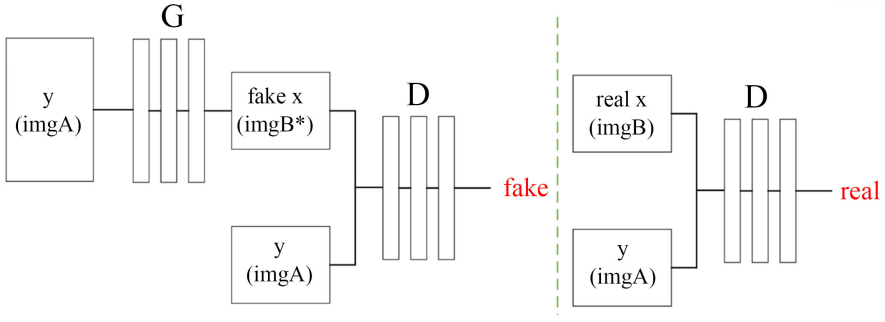


Figure 2. Pix2Pix model structure diagram.

Training Settings: Parameter information of Pix2Pix model: set batch_size to 64 and epoch to 100; MSE and L1 loss functions were selected for the loss function, and adam optimizer was selected for the optimizer.

Resnet50 model parameters: Set batch_size to 64 and epoch to 30. The cross entropy loss function was selected as the loss function, and the adam optimizer was selected as the optimizer. The data enhancement operations of random clipping and random rotation are added to the trained sample data.

4.3. Loss Function of Pix2Pix Model

In the generator, we employ both Mean Squared Error (MSE) and L1 loss functions. MSE loss, also known as Mean Squared Error loss, is used for training the generator (pix_G). The Mean Squared Error loss function calculates the squared differences at the pixel level between the generated image and the target image. It measures the overall pixel-wise differences between the generated image and the target image, with the aim of making the generated image as close as possible to the target image. During training, the generator's objective is to minimize the MSE loss. The MSE expression is as follows:

$$\mathrm{MSE} = \frac{1}{n} * \sum \left(y_i - \hat{y}_i \right)^2 . \tag{7}$$

The L1 loss function, also known as Mean Absolute Error (MAE) loss, is used for training the generator (pix_G) alongside the Mean Squared Error (MSE) loss. The Mean Absolute Error loss function calculates the absolute differences at the pixel level between the generated image and the target image. Unlike the MSE loss, the L1 loss places more emphasis on detailed differences between the generated image and the target image because it measures absolute differences at the pixel level. During training, the generator's objective is to minimize the L1 loss. The MAE expression is as follows:

$$\mathrm{MAE} = \frac{1}{n} * \sum \left| y_i - \hat{y}_i \right| . \tag{8}$$

By using both of these loss functions together, the generator can simultaneously focus on the overall structure and local features to generate high-quality images. The final generated image should be both globally similar to the target image and preserve fine details. This combination of losses is common in many image-to-image generation tasks, such as image translation, denoising, style transfer, and more.

In the case of the discriminator, we only use the MSE Loss function. The discriminator's role is to distinguish between the generated images and real images. Therefore, it needs to measure pixel-level differences between the two to help train the generator to produce more realistic images.

4.4. Training Process

We trained both the generator and discriminator of the Pix2Pix model using pairs of corresponding original samples and samples obtained by passing the

original samples through denoising model A. Here's a description of the loss changes for both the generator and discriminator during training:

Figure 3 shows that both the generator and discriminator losses have stabilized after 100 epochs of training indicating that the training process has converged. This convergence suggests that the Pix2Pix model has learned to generate images that are close to the target images and that the discriminator can no longer effectively distinguish between the generated and real images.

4.5. Defense Performance Analysis

We input the original samples into the trained Pix2Pix model, use the generated samples as input to the resnet50 classification model for training, and save the best results on the validation set during training as the final defense model.

In order to intuitively compare the defensive performance of the models, we conducted tests using four types of adversarial samples that succeeded in attacks. We compared the results of three different classification models: the model obtained through adversarial training by directly mixing normal samples with adversarial samples generated by different attack methods (Adv-train), the non-local means filtering algorithm (NL-means), and the matrix estimation-based effective adversarial robustness defense model (ME-Net). Among these, Adv-train is a white-box defense method, NL-means is a black-box defense method, and ME-Net is a semi-black-box defense method. The results are as follows:

Table 1 and Table 2 have revealed that the IDP model exhibits robust defensive performance. In some cases, it even outperforms white-box defense models like Adv-train. This is because our approach, which incorporates denoising and Pix2Pix image transformation during both training and testing phases, reduces the feature distribution disparities between different adversarial samples and the original samples. As a result, it delivers better performance. In contrast, Adv-train struggles to achieve good convergence during training due to the diversity and complexity of different adversarial samples, which lead to significant differences in feature distribution. NL-means, on the other hand, applies a consistent preprocessing method when facing unknown adversarial samples, but it cannot effectively remove the noise impact from adversarial samples. ME-Net's approach involves disrupting the structure of adversarial noise by using randomly masked images. It then employs matrix estimation techniques to recover the intrinsic structure of the images from noisy and incomplete observations, enhancing the neural network's robustness to adversarial noise. It also provides additional training data by generating variants to improve robustness. However, when dealing with noise of different feature distributions simultaneously, the significant differences in distribution can disrupt pixel correlations in the image, potentially affecting the final defensive performance. In contrast, the IDP approach in this paper focuses on minimizing the dissimilarity between adversarial samples and training samples rather than merely targeting noise removal. The combination of image denoising and image transformation represents two

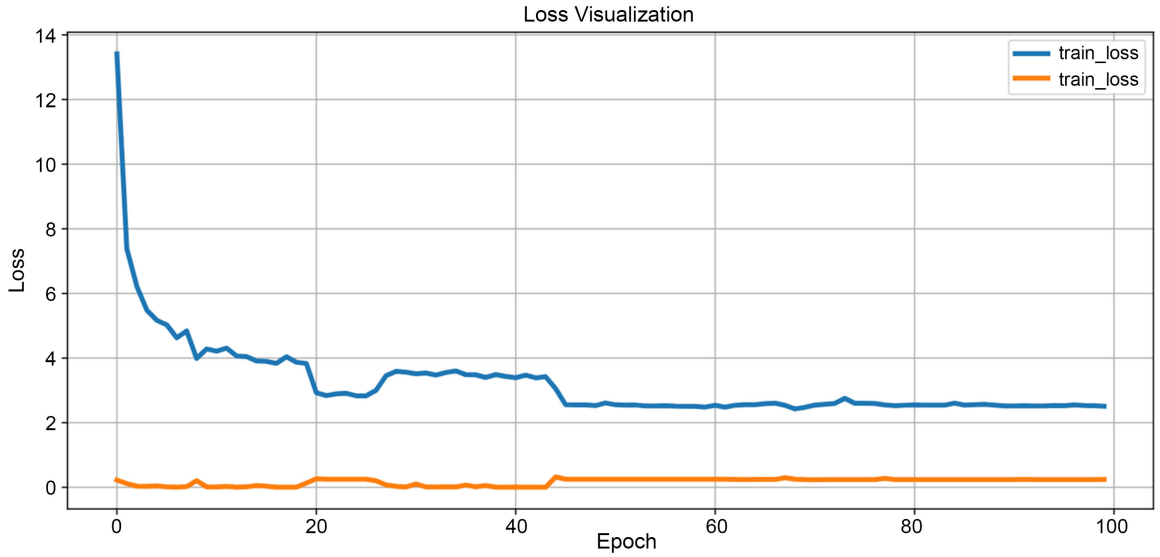


Figure 3. Changes in training loss of Pix2Pix model.

Table 1. Prediction accuracy of mnist images by different methods.

algorithm	FGSM	I-FGSM	DeepFool	UPSET
Adv-train	0.861	0.824	0.889	0.902
NL-means	0.467	0.396	0.453	0.381
ME-Net	0.669	0.678	0.796	0.645
IDP	0.761	0.685	0.843	0.836

Table 2. Prediction accuracy of CIFAR10 images by different methods.

algorithm	FGSM	I-FGSM	DeepFool	UPSET
Adv-train	0.686	0.634	0.757	0.728
NL-means	0.420	0.363	0.371	0.415
ME-Net	0.641	0.578	0.696	0.663
IDP	0.736	0.624	0.821	0.785

distinct defense methods that, when combined, further enhance the model's performance.

In order to compare the similarity of data distributions between datasets, we employed t-SNE (t-Distributed Stochastic Neighbor Embedding) [23], a nonlinear dimensionality reduction technique. T-SNE is widely used in data science and machine learning for clustering, classification, dimensionality reduction, and visualization. It maps each sample in the dataset to a point in a two-dimensional space, allowing us to observe the distances between corresponding points and compare their feature similarity.

Figure 4 shows the distribution among samples contained in different categories. Specifically, we conducted feature dimensionality reduction on the original samples, adversarial samples, and adversarial samples generated through IDP.

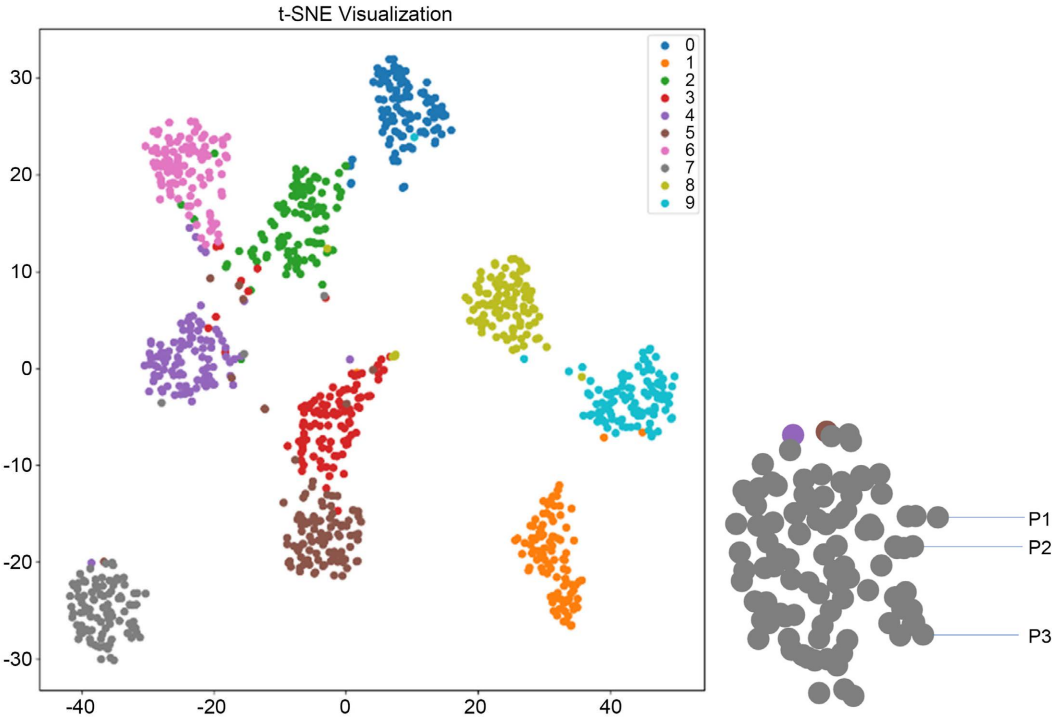


Figure 4. t-SNE diagram.

We then calculated the Euclidean distance between the reduced-dimensional corresponding samples for both the original and IDP-generated samples in comparison to the adversarial samples. The average Euclidean distance for each category was determined. By comparing these average Euclidean distances within the same category, we could assess the degree of feature distribution similarity. Smaller average Euclidean distances indicate higher similarity, while larger distances suggest lower similarity.

We selected the "7th" class and calculated the two-dimensional coordinates for each sample. For example, for samples P1, P2, and P3, let their corresponding coordinates be (x_1, y_1), (x_2, y_2), and (x_3, y_3) respectively. For the "7th" class cluster, the average coordinates are calculated as follows:

$$\left(\frac{x_1 + x_2 + \cdots + x_n}{n}, \frac{y_1 + y_2 + \cdots + y_n}{n}\right). \tag{9}$$

We calculate the average coordinates for each cluster and then compare the distances between the average coordinates of the same clusters.

Table 3 and Table 4 show the average coordinates of the same clusters. Typically, in a t-SNE plot, if the samples within each cluster are more tightly grouped, it indicates better robustness of the model. For adversarial samples, the distribution within each cluster is generally more scattered. We use the Average Cluster Inertia (ACI) metric to measure the tightness of sample distributions within the same category. A smaller ACI value suggests that the sample distribution is tighter, whereas a larger value indicates a more scattered distribution. The expression for Average Cluster Inertia is as follows, where $(\hat{x}, \hat{y})$ represents the

average coordinates of a particular category:

$$\mathrm{ACI} = \frac{1}{n}\sum\sqrt{\left(x_i - \hat{x}\right)^2 + \left(y_i - \hat{y}\right)^2} \ . \tag{10}$$

Table 5 and Table 6 show that samples generated by the IDP model have a smaller average Euclidean distance within the same original sample cluster compared to adduced samples. Additionally, smaller average intra-cluster and inter-cluster distances among different classes imply that the samples within each class are distributed more closely. The experimental results strongly validate the higher feature similarity between the samples generated by the IDP model and the original samples. As a black-box defense model, the IDP model demonstrates effective defense against adversarial attacks, particularly when facing unknown attack types, relying solely on the robust transformation capabilities of image denoising and Pix2Pix.

Table 3. Average distance between original sample and adversarial sample.

Class 0	37.069
Class 1	35.116
Class 2	64.660
Class 3	58.290
Class 4	65.758
Class 5	61.934
Class 6	53.944
Class 7	61.045
Class 8	40.001
Class 9	58.332

Table 4. Average distance between original sample and the sample generated by IDP.

Class 0	21.731
Class 1	27.026
Class 2	40.796
Class 3	41.618
Class 4	33.097
Class 5	39.059
Class 6	34.703
Class 7	41.469
Class 8	22.585
Class 9	30.533

Table 5. ACI value of the adversarial sample.

Class 0	34.31
Class 1	28.36
Class 2	50.39
Class 3	44.65
Class 4	45.42
Class 5	39.70
Class 6	45.03
Class 7	34.00
Class 8	49.06
Class 9	34.31

Table 6. ACI value of the sample generated by IDP.

Class 0	16.12
Class 1	19.59
Class 2	24.31
Class 3	30.40
Class 4	30.33
Class 5	31.71
Class 6	21.53
Class 7	25.05
Class 8	13.25
Class 9	24.06

5. Conclusion

This paper explores the challenges faced by current adversarial defense mechanisms and proposes a practical solution. Traditional defense models often rely on adversarial training, which may have practical issues such as high training costs and difficulties in defending against unknown attacks. To address these challenges, this paper introduces a black-box adversarial defense method based on image denoising and Pix2Pix. This method does not require adversarial training yet demonstrates significant robustness. While this approach performs well against various unknown attacks, it has certain limitations. Specifically, it is less effective in defending against adversarial perturbations that make significant global pixel modifications. This is an area for further research and improvement. Future work will focus on optimizing the model, exploring transfer learning, ensemble methods, and other techniques to enhance the overall performance of the defense method.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Guand, S. and Rigazio, L. (2015) Towards Deep Neural Network Architectures Robust to Adversarial Examples. *Proceedings of the* 2015 *International Conference on Learning Representations*, San Diego, 7-9 May 2015, 1-9.

[2] Tabacof, P. and Valle, E. (2015) Exploring the Space of Adversarial Images. 2016 *International Joint Conference on Neural Networks* (*IJCNN*), Vancouver, 24-29 July 2016, 426-433. https://doi.org/10.1109/IJCNN.2016.7727230

[3] Goodfellow, I.J., Shlens, J. and Szegedy, C. (2015) Explaining and Harnessing Adversarial Examples. 3*rd International Conference on Learning Representations*, San Diego, 7-9 May 2015, 1-11.

[4] Guo, C., Rana, M., Cisse, M. and van der Maaten, L. (2021) Countering Adversarial Images Using Input Transformations. https://arxiv.org/pdf/1711.00117.pdf

[5] Li, C.C. (2021) Research on Adversarial Sample Generation Algorithm for Deep Learning Models. Master's Thesis, Beijing University of Posts and Telecommunications, Beijing. (In Chinese)

[6] Szegedy, C., Zaremba, W., Sutskever, I., *et al.* (2014) Intriguing Properties of Neural Networks. 2*nd International Conference on Learning Representations*, Banff, 14-16 April 2014, 1-10.

[7] Moosavi-Dezfooli, S., Fawzi, A. and Frossard, P. (2016) DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, 27-30 June 2016, 2574-2582. https://doi.org/10.1109/CVPR.2016.282

[8] Dong, Y., Liao, F., Pang, T., *et al.* (2018) Boosting Adversarial Attacks with Momentum. 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 9185-9193. https://doi.org/10.1109/CVPR.2018.00957

[9] Kurakin, A., Goodfellow, I. and Bengio, S. (2016) Adversarial Examples in the Physical World. arXiv: 1607.02533.

[10] Sarkar, S., Bansal, A., Mahbub, U. and Chellappa, R. (2017) UPSET and ANGRI: Breaking High Performance Image Classifiers. arXiv: 1707.01159.

[11] Papernot, N., McDaniel, P., Wu, X., Jha, S. and Swam, A. (2016) Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks. *Proceedings of the IEEE Symposium on Security and Privacy*, San Jose, 22-26 May 2016, 582-597. https://doi.org/10.1109/SP.2016.41

[12] Ross, A. and Doshi-Velez, F. (2018) Improving the Adversarial Robustness and Interpretability of Deep Neural Networks by Regularizing their Input Gradients. *Proceedings of the* 32*nd AAAI Conference on Artificial Intelligence*, New Orleans, 2-7 February 2018, 1660-1669. https://doi.org/10.1609/aaai.v32i1.11504

[13] Cisse, M., Bojanowski, P., Grave, E., *et al.* (2017) Parsimonious Neural Networks for Improved Robustness to Adversarial Examples. *Proceedings of the International Conference on Machine Learning*, Sydney, 6-11 August 2017, 854-863.

[14] He, K.M., Zhang, X.Y., Ren, S.Q. and Sun, J. (2015) Deep Residual Learning for Image Recognition. arXiv: 1512.03385.

[15] Shan, B.L., *et al.* (2023) Graph Learning from Band-Limited Data by Graph Fourier Transform Analysis. *Signal Processing*, **207**, Article ID: 108950. https://doi.org/10.1016/j.sigpro.2023.108950

[16] Li, L. (2013) Image Denoising Algorithm Combining Fourier Transform and NSCT. Master's Thesis, Xiangtan University, Xiangtan. (In Chinese)

[17] Wu, F., Yang, W.X., Xiao, L.M., *et al.* (2020) Adaptive Wiener Filter and Natural Noise to Eliminate Adversarial Perturbation. *Electronics*, **9**, Article 1634. https://doi.org/10.3390/electronics9101634

[18] Salehi, H., Vahidi, J., Abdeljawad, T., *et al.* (2020) A SAR Image Despeckling Method Based on an Extended Adaptive Wiener Filter and Extended Guided Filter. *Remote Sensing*, **12**, Article 2371. https://doi.org/10.3390/rs12152371

[19] Guo, H., *et al.* (2022) Pixel-Based Approach to Delay Multiply and Sum Beamforming in Combination with Wiener Filter for Improving Ultrasound Image Quality. *Ultrasonics*, **128**, Article ID: 106864. https://doi.org/10.1016/j.ultras.2022.106864

[20] Guo, Y.C. and Wang, S.B. (2009) Denoising Method for Ultrasound Medical Images Based on Wiener Filtering and Wavelet Fusion. *China Medical Imaging Technology*, **25**, 1496-1499. (In Chinese)

[21] Jiang, S.P. and Hao, X.J. (2009) Mixed Fourier and Wavelet Image Denoising Using Gaussian Scale Mixture Model of Wavelet Coefficients. *Journal of Image and Graphics*, **14**, 448-451. (In Chinese)

[22] Isola, P., Zhu, J.Y., Zhou, T. and Efros, A.A. (2016) Image-to-Image Translation with Conditional Adversarial Networks. 2017 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Honolulu, 21-26 July 2017, 5967-5976. https://doi.org/10.1109/CVPR.2017.632

[23] Van, D. and Hinton, G. (2008) Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research*, **9**, 2579-2605.