Enhancing the Robustness of Deep Neural Networks by Meta-Adversarial Training

You-Kang Chang, Hong Zhao, and Wei-Jie Wang
(Corresponding author: You-Kang Chang)

School of Computer and Communications, Lanzhou University of Technology
36 Peng-jia-ping Road, Lanzhou, Gansu 730050, China
Email: 2507576651@qq.com

(Received July 29, 2022; Revised and Accepted Dec. 3, 2022; First Online Dec. 13, 2022)

Abstract

Adversarial training can effectively defend against the impact of adversarial attacks on deep neural networks but suffers from poor generalization ability and low defense efficiency. To address this problem, this paper proposes a method combining meta-learning with adversarial training to enhance the robustness of deep neural networks. Firstly, a training dataset containing adversarial examples and clean examples is constructed, and conduct adversarial training on the deep neural network. Secondly, the features extracted from the adversarial training are learned using the meta-learning method, and the problem of the need to continuously input a large number of adversarial examples for training in adversarial training is solved by using the feature that meta-learning has strong adaptability in the face of new tasks. Experimental results show that this method can improve the robustness of deep neural networks and effectively resist standard classes of adversarial attacks.

Keywords: Adversarial Training; Adversarial Attack Defense; Meta-learning; Neural Networks; Robustness Studies

1 Introduction

Deep neural networks play an increasingly important role as deep learning is applied to an increasingly wide range of scenarios. For example, it has shown good performance in autonomous driving [22,26], medical image analysis [1,27] and image recognition [31]. However, research and practical applications have shown that deep neural networks are vulnerable to adversarial attacks [6,10], and are deceived by adversarial examples to produce wrong results. During the training phase of a deep neural networks, the attacker attacks by modifying the training dataset, changing the characteristics of the input data or the data labels. In the testing phase of deep neural networks, white-box attacks and black-box attacks can be used, where white-box attacks are performed by obtaining the structure of deep neural networks to generate adversarial examples, and black-box attacks are performed by querying the structure of network models and exploiting the transferability between adversarial examples.

In response to the vulnerability of deep neural networks to adversarial example attacks, researchers have successively proposed a variety of defense methods, which are mainly divided into three categories. The first category is data preprocessing, such as adversarial example denoising [29] and data compression [14], which are computationally fast and do not require modification of the network structure of the model. The disadvantage is that when modifying the input examples, the high-frequency information of the examples will be lost, making the network model unable to extract the correct feature regions and leading to the wrong classification of the neural network. The second category is to enhance the robustness of deep neural networks, such as adversarial training [8] defensive distillation methods [17] and deep compression network [7]. Such methods improve the stochasticity of the network model and the cognitive performance of the network to a certain extent. but their defensive efficiency decreases significantly if specific attacks are performed on a particular network. The third category of methods is the detection of adversarial examples before they are fed into the deep neural network, such as based on Generative Adversarial Network (GAN) [23], based on MagNet [16] and Defense Perturbation [21]. These methods have good generalization ability and good defense against black and gray box attacks in particular, however, their performance decreases substantially in the case of white box attacks.

For the second category of defense methods in the adversarial training defense mechanism, which serves as one of the most promising defense methods to improve the robustness of deep neural networks [13], it is necessary to add newly emerged adversarial examples to the training set for adversarial training in the face of never-appeared adversarial examples, and this method to improve the robustness of deep neural networks through violent training has the problems of long training time and poor gen-

eralization ability. To tackle this problem, this paper proposes an adversarial training defense method that introduces meta-learning technology, combining adversarial training with meta-learning method, and using the characteristics of meta-learning with strong generalization and high recognition accuracy to solve the problem of poor generalization of adversarial training.

A brief overview of our contributions is as follows:

1) Application of meta-learning methods to the adversarial training process of deep neural networks to enhance the robustness of deep neural networks;

2) The method is not only effective in defending against adversarial samples, but also has no impact on the accuracy of clean samples, which are the original data set, not generated by the adversarial attack algorithm;

3) The method can still maintain high accuracy with strong generalization in the face of unprecedented adversarial examples;

4) The method is not only applicable to white-box attacks, but also has strong defense capability in the face of black-box attacks.

This paper is organized as follows: The seco section reviews related work. The third section describes the Meta-adversarial training (Meta-adv training) defense method. The fourth section conducts the experimental design as well as the analysis of the results. Finally, the full paper is summarized in section fifth.

2 Related Work

2.1 Adversarial Attacks

Kurakin *et al.* [10] proposed the Basic Iterative Method (BIM) method, where the generated perturbations are added to the input image multiple times incrementally through multiple iterations along the direction of the gradient and the gradient direction is recalculated after each iteration. Carlini and Wagner [2] proposed the Carlini and Wagner (C&W) method to generate adversarial examples using the Adam-Optimizer optimizer. Moosavi-Dezfooli *et al.* [19] proposed the DeepFool method, which is based on a binary classification problem where the minimum perturbation vector added is the vertical distance vector between x0 and the straight line. Xie *et al.* [30] proposed Diverse-Input-Iterative FGSM(DI^2FGSM) and Momentum-Diverse-Input-Iterative FGSM(MDI^2FGSM), where DI^2FGSM performs a random transformation of the image with probability p during the generation of the adversarial example, MDI^2FGSM method improves the efficiency of the attack by adding momentum to the DI^2FGSM method to avoid local maximums.

Figure 1 illustrates the different adversarial examples and observes the differences between them from a visual perspective and finds that the effect of the added adversarial perturbations is not significant. However, their pixel values are displayed in three-dimensional coordinates for comparison, as shown in Figure 2, where yellow indicates the pixel value is higher and blue indicates the pixel value is lower, and it can be seen that the adversarial examples after adding the perturbation differ from the normal examples in terms of pixel intensity, and the pixel values of the clean examples are smoother compared to the adversarial examples. For example, the pixel values of the adversarial examples generated by the BIM method are continuous and constant in some areas, while the pixel values of the adversarial examples generated by the MDI^2FGSM attack method fluctuate more. The variation of pixel values causes the deep neural network to extract the wrong feature regions and eventually output the wrong results.

2.2 Adversarial Training and Meta-Learning

In response to the influence brought by adversarial attacks, adversarial training and its optimization methods have been successively proposed as the most effective defense methods at present. Zhang *et al.* [32] proposed a feature scattering-based adversarial training defense method to generate adversarial examples for training by feature scattering in the potential space. Zhang *et al.* [33] proposed Friendly Adversarial Training (FAT) defense method, they believed that in using PGD attack method to generate adversarial examples for adversarial training, it will affect the accuracy of clean examples and even cause the neural network not to converge, so the proposed method will stop in time during the process of generating adversarial examples using PGD iterations and return to the adversarial examples vicinity the decision boundary for training, gradually enhancing the robustness of the deep neural network and ensuring the accuracy of clean examples. The existing adversarial training uses the adversarial examples generated by one attack method to train the deep neural networks, which cannot effectively cover other types of adversarial examples, Kwon *et al.* [11] proposed a diverse adversarial training method using a combined training set of FGSM, I-FGSM, DeepFool and C&W for adversarial training to enhance the robustness against unknown adversarial attacks.

The defense method based on adversarial training enhances the robustness of the network model by improving the randomness and cognitive properties of the deep neural network, but this method needs to retrain the network model when facing unknown types of adversarial examples, which has the problems of poor generalization ability and large computational resource consumption. For this reason, this paper introduces the meta-learning method in the process of adversarial training.

Meta-learning is a learning approach that imitates biological use of prior knowledge to quickly learn new and unseen things, and it can be a good solution to the prob-

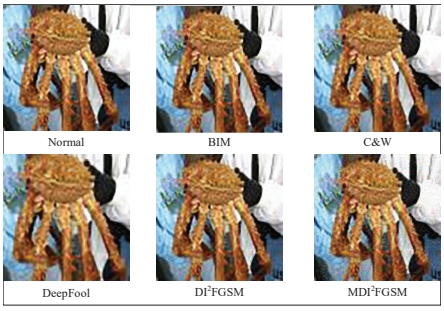

Figure 1: Comparison between different adversarial examples

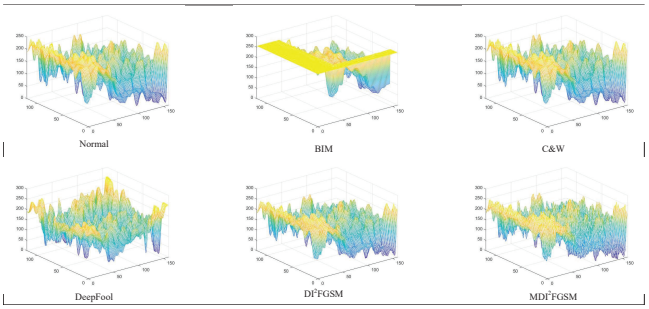

Figure 2: Three-dimensional visualization of different adversarial example pixels

lems of deep neural networks such as low robustness, poor generalization, difficulty in learning and adapting to unobserved tasks, and dependence on large-scale data. Research on meta-learning can improve the robustness and generalization of network models.

Meta-learning has now been applied to many fields, and good experimental results have been achieved by integrating meta-learning with other network models. Finn *et al.* [5] proposed the Model-Agnostic Meta-Learning (MAML) for Fast Adaptation of Deep Networks method, which is a model-independent meta-learning method for different learning problems. Firstly, The parameters of the network model are optimized using gradient descent, and then in a new task, the parameters are fine-tuned by training a small amount of data to make the network model with good generalization performance. Later, Zhang *et al.* [34] proposed the MetaGAN meta-learning method, which combined meta-learning with the Generative Adversarial Network(GAN) model to help classifiers learn clearer decision boundaries in small example data and improve the generalization performance of the network model by introducing an adversarial generative model. Li *et al.* [12] proposed Adversarial Feature Hallucination Networks (AFHN) to ensure the distinguishability and diversity of few shot data. Mandal *et al.* [15] combined Graph Neural Networks (GNNs) with meta-learning to improve the generalization performance of GNNs in the face of few shot data.

3 Meta-Adversarial Training Methods

This paper proposes to enhance the robustness of deep neural networks using meta-adversarial training, which is divided into three main phases: firstly, clean examples are combined with generated adversarial examples into a training set for feature extraction using adversarial training; secondly, the extracted features are quickly adapted to a few shot learning task in the meta-learning training phase; and finally, the defense method is tested against the adversarial examples in the meta-learning testing phase, and the robustness of the deep neural network is evaluated at the same time.

Adversarial training: adversarial training is trained by fusing adversarial examples with clean examples, which can regularize the deep neural network to certain extent and adapt the network model to this change and enhance the generalization ability. Huang *et al.* [9] defined the Min-Max problem for the first time, where: Min refers to minimizing the classification error of the network model during training process. Max refers to finding the adversarial perturbation of the input example that maximizes the classification error of the network model and states that the key to solving the Min-Max problem is to find the adversarial example with stronger attack performance. Later, Shaham *et al.* [24] considered the Min-Max problem from the perspective of robust optimization and proposed a framework for adversarial training, as shown in Equation (1):

$$\min_{\theta} \mathbb{E}_{(Z,y)\sim\mathcal{D}} \left[\max_{\|\delta\|\leqslant\varepsilon} L\left(f_\theta(X+\delta), y\right) \right] \tag{1}$$

where the inner layer denotes maximization, X denotes the input example, δ denotes the perturbation added to the input example,$f_\theta()$ denotes the deep neural network, and y denotes the true label of the clean example. $L\left(f_\theta(X+\delta), y\right)$ denotes the loss between the output label of the adversarial example X+δ passing through the deep neural network and the true label. max (L) denotes the optimization objective, which aims to find the perturbation that maximizes the loss function so that the added perturbation should disturb the deep neural network as much as possible.

The outer layer represents the minimization formulation of the optimized deep neural network, which trains the deep neural network to minimize its loss on the training data when the adversarial perturbation has been determined, adversarial perturbation has been determined, allowing the network model to have some robustness to adapt to the perturbation. Equation (1) describes the idea of adversarial training, but it does not describe how to design a perturbation δ with strong attack performance. therefore, the researchers proposed a variety of attack methods to find the perturbation δ. In fact, during adversarial training, the stronger the attack performance of the perturbation δ can make the deep neural network more robust.

In the meta-adversarial training defense method, feature extraction is first performed on the data in the training set using the convolution operation, then in the meta-learning phase, the parameters of the feature extractor are learned by scaling and shifting transformations to make the deep neural network quickly adapt to few shot tasks; finally, the accuracy of the test data is output in the meta-testing phase. The specific process is as follows:

Feature extraction: The parameters of the feature extractor Θ and classifier θ are first initialized, and then some of the clean examples in the mini-ImageNet training set are replaced with the generated adversarial examples, and the parameters of feature extractor Θ and classifier θ are learned by gradient descent method using the ResNet network model, as shown in Equation (2):

$$[\Theta;\theta] = [\Theta;\theta] - \alpha\nabla \lim_{x\to\infty} \mathcal{L}_D([\Theta;\theta]) \tag{2}$$

where α denotes the learning rate and $\mathcal{L}_D$ denotes the cross-entropy loss function, as shown in Equation (3):

$$\mathcal{L}_D([\Theta; \quad \theta]) = \frac{1}{\mathcal{D}} \sum_{(x,y)\in\mathcal{D}} l\left(f_{[\Theta;\theta]}(x), y\right). \tag{3}$$

Meta-learning stage: The feature extractor parameters Θ learned in the feature extraction phase remain fixed during the few shot learning process, and they are scaled and shifted transformed in the meta-learning phase to quickly adapt to unseen data examples; however, the classifier parameters θ need to be reinitialized and updated due to the inconsistency in the number of categories between the feature extraction phase and the meta-learning phase. as shown in Equation (4):

$$\theta' \leftarrow \theta - \beta\nabla_\theta \mathcal{L}_{\mathcal{T}^{(tr)}}\left([\Theta; \quad \theta], \Phi_{S_{\{1,2\}}}\right) \tag{4}$$

where Φ_{S_1} denotes the scaling transformation, which is initialized to 1, Φ_{S_2} denotes the shifting transformation, initialized to 0, $\Phi_{S_{\{1,2\}}}$ denotes the scaling and shifting transformation, $\mathcal{T}^{(tr)}$ denotes the training data, and β denotes the learning rate. Different from θ in Equation (2), θ in Equation (4) focuses on a small number of classes in the meta-learning training task to classify in a few shot of data, θ' denoting the parameters of the current classification task.

During the test process, the parameters of the scaling and shifting are optimized by calculating the loss values using the test data $\mathcal{T}^{(te)}$, while updating the parameters θ, as shown in Equations (5) and (6):

$$\Phi_{S_i} = \Phi_{S_i} - \gamma\nabla_{\Phi_{S_i}} \mathcal{L}_{\mathcal{T}^{te}}\left([\Theta;\theta'], \Phi_{S\{1,2\}}\right) \tag{5}$$

$$\theta = \theta - \gamma\nabla_\theta \mathcal{L}_{\mathcal{T}^{te}}\left([\Theta;\theta'], \Phi_{S\{1,2\}}\right) \tag{6}$$

For a given Θ, the i-th layer of the feature extractor Θ contains K neurons, that is, it contains K parameter pairs and $\{(\mathrm{W_{i,k}}, \mathrm{b_{i,k}})\}$ denotes the weights and bias respectively, and if the input is X, the formula for applying $\Phi_{S_{\{1,2\}}}$ to (W, b) is shown in Equation (7):

$$SS\left(X; W, b; \Phi_{S_{\{1,2\}}}\right) = \left(W \odot \Phi_{S_1}\right) X + \left(b + \Phi_{S_2}\right) \tag{7}$$

The weights trained on large-scale datasets are migrated to the meta-learning task using the already optimized scaling and shifting. which ensure fast convergence of the deep neural network in the face of few shot data and effectively reduce overfitting.

4 Experimental Design and Analysis of Results

4.1 Experimental Platform

The experimental platform for this study is based on ubuntu 18.04, with 128G of experimental running memory. Hardware equipment using a graphics card NVIDIA Tesla V100 GPU with 32G of video memory. The experimental environment uses the PyTorch deep learning framework that supports GPU accelerated computing, and the cuda environment is configured with NVIDIA CUDA 11.3 and cuDNN V8.2.1 deep learning acceleration library.

4.2 Dataset Setup

This experiment uses the miniImageNet [28] dataset to verify the effectiveness of the model. mini-ImageNet contains a total of 100 categories, with 64 categories in the training set, 16 categories in the validation set, and 20 categories in the test set, each containing 600 images, for a total of 60,000 data samples of size 84 × 84. During the experiments, the data are first preprocessed and the samples are upsampling to 299 × 299 pixel size, and then the adversarial examples are generated using the white-box adversarial attack methods BIM, C&W, DeepFool, DI^2FGSM, MDI^2FGSM, and the black-box adversarial attack methods P-RGF, RGF [4] and Parsimonious [18].

4.3 Parameter Setting

In the pre-training phase, the parameters of the model were optimized using the SGD optimizer, setting the learning rate α=0.1, the momentum set to 0.9, and the weight decay value to 0.0005; the parameters were updated using the cross-entropy loss function, specifying that the loss value decays 0.2 in the learning rate when the model does not decline in 30 rounds. The parameter settings are shown in Table 1.

Meta-learning training phase using the Adam optimizer for optimization of parameters, setting the learning rate $\beta = 0.01$. The cross-entropy loss function is also used for parameter updating, and the learning rate is specified to decay by 0.5 when the loss value does not decline in

Table 1: Pre-training phase parameter setting

Parameters	Setting
Learning rate	0.1
Epoch	100
Weight decay	0.0005
Batch_size	128
Learning rate decay	0.2
Momentum	0.9
Step_size	30

Table 2: Meta-learning training phase parameter setting

Parameters	Setting
Learning rate	0.01
Epoch	100
Train_query	15
Val_query	15
Learning rate decay	0.5
Step_size	10
Num_batch	100

10 consecutive rounds. The training process uses 100 different tasks, with 15 examples per category in each task selected for training and 15 examples selected for validation. The parameters are set as shown in Table 2.

4.4 Analysis of Experimental Results

4.4.1 Effect of the Proportion of Adversarial Examples

During the experiments, the effects of different proportions of adversarial examples on the robustness of the deep neural network ResNet-12 are compared. Firstly, clean examples in the training set are replaced with adversarial examples in different proportions of 10%, 30%, 50%, 70% and 90% for adversarial training. The test set uses the generated adversarial examples. The experimental results of meta-adversarial training are shown in Table 3 and Table 4.

Table 3 and Table 4 show the experimental results for 1shot-5way and 5shot-5way during the meta-adversarial training, respectively. It can be seen that with the increasing proportion of adversarial examples, the accuracy of the experimental results shows an overall increasing trend, the reason being that the deep neural network treats the adversarial examples as clean examples, fitting the distribution of the data, and the loss generated by the adversarial examples as part of the loss of the deep neural network, increasing the loss of the model without modifying the structure of the network model, producing a regularization effect.

Table 3: Accuracy of 1shot-5way on the adversarial example (%)

	10%	30%	50%	70%	90%
BIM	0.6588	0.6743	0.6694	0.6729	0.6720
C&W	0.6096	0.6090	0.6112	0.6030	0.6001
DeepFool	0.6068	0.5845	0.5577	0.5905	0.5661
DI^2FGSM	0.4528	0.4691	0.5285	0.5342	0.5932
MDI^2FGSM	0.4375	0.4579	0.4983	0.5264	0.5811

Table 4: Accuracy of 5shot-5way on the adversarial example (%)

	10%	30%	50%	70%	90%
BIM	0.8243	0.8086	0.8099	0.8281	0.8269
C&W	0.7695	0.7697	0.7494	0.7644	0.7604
DeepFool	0.7673	0.7496	0.7223	0.7536	0.7276
DI^2FGSM	0.6221	0.6282	0.7024	0.7047	0.7525
MDI^2FGSM	0.5888	0.6224	0.6742	0.6992	0.7407

Table 5: Accuracy of 1shot-5way on clean examples (%)

	Clean	10%	30%	50%	70%	90%
BIM	0.6045	0.6109	0.6043	0.6030	0.5956	0.5800
C&W		0.6029	0.5997	0.6122	0.6092	0.6082
DeepFool		0.6096	0.5969	0.5905	0.5922	0.5826
DI^2FGSM		0.6060	0.5832	0.5505	0.4297	0.4180
MDI^2FGSM		0.5970	0.5656	0.5359	0.3932	0.3819

The adversarial training has achieved good results against adversarial examples, and the next step will test the effect of adversarial training in the face of clean examples. and the experimental results are shown in Table 5 and Table 6.

Clean in Table 5 and Table 6 indicates that both the training and test sets are clean examples, and can achieve 60.45% and 76.25% accuracy in the 1shot-5way and 5shot-5way cases, respectively. However, when the adversarial examples are continuously added to the training set for adversarial training, the accuracy of clean examples shows an overall decreasing trend, such as when 90% of MDI^2FGSM adversarial examples are added for adversarial training, the accuracy of clean examples in the 1shot-5way and 5shot-5way cases is only 38.19% and 55.94%, respectively, which is different from the accuracy of clean examples. Therefore, after trade-off between the accuracy of the deep neural network against adversarial examples and clean examples, we add 20% of the adversarial examples randomly to the training set for adversarial training to ensure both the accuracy of clean examples and the robustness of the deep neural network against adversarial examples. Next, we will use 20% of the adversarial examples for adversarial training to validate the defensive

Table 6: Accuracy of 5shot-5way on clean examples (%)

	Clean	10%	30%	50%	70%	90%
BIM	0.7625	0.7674	0.7628	0.7656	0.7602	0.7368
C&W		0.7612	0.7604	0.7696	0.7693	0.7673
DeepFool		0.7697	0.7603	0.7514	0.7546	0.7474
DI^2FGSM		0.7673	0.7502	0.7271	0.6280	0.6126
MDI^2FGSM		0.7589	0.7305	0.7098	0.5707	0.5594

capability against migration between attack methods.

4.4.2 Migratory Defense Against Attacks

The proposed meta-adversarial training defense method allows the deep neural network to still show good robustness against unprecedented adversarial examples, and for this reason, this section verifies the migration between the meta-adversarial training method defense against different adversarial attacks. The experimental results are shown in Table 7 and Table 8. Training indicates that 20% of the adversarial examples are added to the clean examples for training, and Test verifies the adversarial training defense against different adversarial attack algorithms.

It can be seen from Table 7 and Table 8 that the meta-adversarial training method can effectively defend against different adversarial examples and enhance the robustness of the deep neural network. For example, meta-adversarial training using 20% of BIM adversarial examples can achieve recognition accuracy of 59.59%, 60.68%, 60.30%, and 57.28% for 1shot-5way in the face of C&W, DeepFool, DI^2FGSM, and MDI^2FGSM attack methods, and this result is slightly lower than that of using C&W adversarial examples for meta-adversarial training of 61.74%, but all are higher than the results of meta-adversarial training using DeepFool, DI^2FGSM and MDI^2FGSM adversarial examples.

In the 5shot5way case, the recognition accuracy has been improved substantially in all cases. When using C&W adversarial examples for meta-adversarial training, the recognition accuracy of BIM, DeepFool, DI^2FGSM, and MDI^2FGSM remains stable, and it is able to reach 74.22% even when facing the MDI^2FGSM attack algorithm, which has a strong attack capability. The reason for the good results is that the meta-learning method can fine-tune the parameters so that the deep neural network can quickly adapt to new tasks and has good generalization performance when facing unseen adversarial examples.

After the above-mentioned comparison experiments, the following experiments will verify the effectiveness of the proposed method in defending against white-box attacks and black-box attacks.

4.4.3 Defending Against White-box Attacks and Black-box Attacks

(1) Defending Against White-Box Attacks

For BIM, C&W, DeepFool, DI^2FGSM and MDI^2FGSM white-box attack algorithms, the proposed meta-adversarial training defense method is compared with CompareNets [25], Self-Supervised Learning (SSL) [3], and Neural Representation Purifier (NRP) [20] defense methods for performance comparison, where CompareNets and SSL are meta-learning methods that use the dataset consistent with the proposed meta-adversarial training method. The experimental results are shown in Table 9.

From Table 9, it can be concluded that meta-adversarial training achieves better accuracy on BIM, C&W, DeepFool, and DI^2FGSM white-box attack algorithms compared to CompareNets, but slightly lower accuracy in the face of the stronger MDI^2FGSM attack algorithm. Meta-adversarial training achieves better accuracy on BIM, C&W, DeepFool white-box attack algorithms compared to SSL, and slightly lower performance than SSL defense methods in the face of DI^2FGSM and MDI^2FGSM attack algorithms. Compared with NRP, meta-adversarial training was lower in accuracy than NRP in the 1shot5way case, but higher in accuracy than the NRP defense method in the 5shot5way case.

(2) Defense Against Black-Box Attacks

For the defense against RGF, P-RGF and Parsimonious black box attacks, the same three defense methods of CompareNets, SSL and NRP are used for comparison with meta-adversarial training. The experimental results are shown in Table 10.

It can be seen from Table 10 that meta-adversarial training outperforms CompareNets across the board in terms of defense effectiveness. Compared to SSL, meta-adversarial training is 0.36% and 0.6% less accurate in the 5shot-5way case when facing RGF and P-RGF adversarial attacks, however, all other metrics are higher than SSL. Compared with NPR, it can be seen that NRP cannot effectively defend against black box attacks and its accuracy is lower than meta-adversarial training in both 1shot-5way and 5shot-5way cases.

Compared with CompareNets, SSL, and NRP defense methods, the proposed meta-adversarial training shows better overall defense performance for both white-box and black-box attacks. The reason is that CompareNets simply superimposes the feature maps directly during feature extraction, and the extracted feature information is destroyed, resulting in its poor adaptability and low accuracy. SSL adopts a self-supervised learning method for few shot learning. Self-supervision can effectively prevent the overfitting phenomenon and enhance the generalization ability and robustness of deep neural networks, so it can maintain stable robustness in the face of white-box attacks and black-box attacks. NRP effectively uses the

Table 7: Migrability of 1shot-5way defense against attacks (%)

Training	Test				
	BIM	C&W	DeepFool	DI^2FGSM	MDI^2FGSM
BIM	**0.6601**	0.5959	0.6068	0.6030	0.5728
C&W	0.6079	**0.6174**	0.6006	0.6048	0.5800
DeepFool	0.5844	0.5887	**0.5825**	0.5881	0.5512
DI^2FGSM	0.5937	0.5921	0.5918	**0.4761**	0.3893
MDI^2FGSM	0.5856	0.5841	0.5820	0.5556	**0.4604**

Table 8: Migrability of 5shot-5way defense against attacks (%)

Training	Test				
	BIM	C&W	DeepFool	DI^2FGSM	MDI^2FGSM
BIM	**0.8241**	0.7541	0.7661	0.7617	0.7351
C&W	0.7748	**0.7735**	0.7649	0.7660	0.7422
DeepFool	0.7554	0.7510	**0.7453**	0.7511	0.7166
DI^2FGSM	0.7604	0.7585	0.7577	**0.6507**	0.5141
MDI^2FGSM	0.7560	0.7512	0.7487	0.7293	**0.6213**

Table 9: Comparison of performance against white-box attacks (%)

	Meta adv-training		CompareNets		SSL		NRP
	1shot5way	5shot5way	1shot5way	5shot5way	1shot5way	5shot5way	
BIM	0.6601	0.8241	0.5894	0.7327	0.6416	0.8080	0.6631
C&W	0.6174	0.7735	0.4935	0.6510	0.5736	0.7677	0.6802
DeepFool	0.5825	0.7453	0.4887	0.6548	0.5553	0.7522	0.6813
DI^2FGSM	0.4761	0.6507	0.4953	0.6397	0.5722	0.6634	0.6152
MDI^2FGSM	0.4604	0.6213	0.4757	0.6361	0.4633	0.6529	0.5936

Table 10: Comparison of performance against white-box attacks (%)

	Meta adv-training		CompareNets		SSL		NRP
	1shot5way	5shot5way	1shot5way	5shot5way	1shot5way	5shot5way	
RGF	0.6031	0.7632	0.4772	0.6243	0.5818	0.7668	0.5359
P-RGF	0.5960	0.7583	0.4780	0.6420	0.5772	0.7643	0.3648
Parsimonious	0.5672	0.7259	0.4925	0.6471	0.5534	0.7039	0.4089

information contained in the feature space of deep neural networks for self-supervised learning, and the method can effectively defend against white box attacks, but is less effective in defending against black box attacks.

5 Conclusions

In this paper, we propose a deep neural network defense method based on meta-adversarial training to defend against the ever emerging adversarial attack methods, which combines meta-learning with adversarial training. First of all, adversarial training as an effective defense method against attacks, it can effectively defend against most of the adversarial attack methods, but its generalization ability in the face of emerging adversarial examples is poor and its robustness is low. To tackle this problem, the meta-learning method is used in the process of adversarial training to improve the robustness of deep neural networks using its better adaptability and generalization ability in few shot tasks. The experimental results show that the overall defense performance of the proposed defense method is stronger compared with other defense methods.

Acknowledgments

This research is supported by the National Natural Science Foundations of China (62166025); Science and technology project of Gansu Province(21YF5GA073); Gansu Provincial Department of Education: Outstanding Graduate Student "Innovation Star" Project(2021CXZX-511, 2021CXZX-512)

References

[1] S. Budd, E. C. Robinson, and B. Kainz, "A survey on active learning and human-in-the-loop deep learning for medical image analysis," *Medical Image Analysis*, vol. 71, p. 102062, 2021.

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

[3] D. Chen, Y. Chen, Y. Li, F. Mao, Y. He, and H. Xue, "Self-supervised learning for few-shot image classification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1745–1749. IEEE, 2021.

[4] S. Cheng, Y. Dong, T. Pang, H. Su, and J. Zhu, "Improving black-box adversarial attacks with a transfer-based prior," *Advances in neural information processing systems*, vol. 32, 2019.

[5] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

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

[7] S. Gu and L. Rigazio, "Towards deep neural network architectures robust to adversarial examples," *arXiv preprint arXiv:1412.5068*, 2014.

[8] J. Ho, B.-G. Lee, and D.-K. Kang, "Attack-less adversarial training for a robust adversarial defense," *Applied Intelligence*, vol. 52, no. 4, pp. 4364–4381, 2022.

[9] R. Huang, B. Xu, D. Schuurmans, and C. Szepesvári, "Learning with a strong adversary," *arXiv preprint arXiv:1511.03034*, 2015.

[10] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*, pp. 99–112, Chapman and Hall/CRC, 2018.

[11] H. Kwon and J. Lee, "Diversity adversarial training against adversarial attack on deep neural networks," *Symmetry*, vol. 13, no. 3, p. 428, 2021.

[12] K. Li, Y. Zhang, K. Li, and Y. Fu, "Adversarial feature hallucination networks for few-shot learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13470–13479, 2020.

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

[14] R. Mahfuz, R. Sahay, and A. ElGamal, "Mitigating gradient-based adversarial attacks via denoising and compression," *arXiv preprint arXiv:2104.01494*, 2021.

[15] D. Mandal, S. Medya, B. Uzzi, and C. Aggarwal, "Metalearning with graph neural networks: Methods and applications," *ACM SIGKDD Explorations Newsletter*, vol. 23, no. 2, pp. 13–22, 2022.

[16] D. Meng and H. Chen, "Magnet: a two-pronged defense against adversarial examples," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 135–147, 2017.

[17] A. Mirzaeian, J. Kosecka, H. Homayoun, T. Mohsenin, and A. Sasan, "Diverse knowledge distillation (dkd): A solution for improving the robustness of ensemble models against adversarial attacks," in *2021 22nd International Symposium on Quality Electronic Design (ISQED)*, pp. 319–324. IEEE, 2021.

[18] S. Moon, G. An, and H. O. Song, "Parsimonious black-box adversarial attacks via efficient combinatorial optimization," in *International Conference on Machine Learning*, pp. 4636–4645. PMLR, 2019.

[19] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

[20] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "A self-supervised approach for adversarial robustness," in *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pp. 262–271, 2020.

[21] F. Nesti, A. Biondi, and G. Buttazzo, "Detecting adversarial examples by input transformations, defense perturbations, and voting," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[22] F. Nezhadalinaei, L. Zhang, M. Mahdizadeh, and F. Jamshidi, "Motion object detection and tracking optimization in autonomous vehicles in specific range with optimized deep neural network," in *2021 7th International Conference on Web Research (ICWR)*, pp. 53–63, IEEE, 2021.

[23] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," *arXiv preprint arXiv:1805.06605*, 2018.

[24] U. Shaham, Y. Yamada, and S. Negahban, "Understanding adversarial training: Increasing local stability of supervised models through robust optimization," *Neurocomputing*, vol. 307, pp. 195–204, 2018.

[25] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1199–1208, 2018.

[26] C. Tian, L. Wang, E. Zhou, K. Liu, S. Du, and J. Liu, "Integration and experimental study of automatic driving system for bus," in *2021 7th International Symposium on Mechatronics and Industrial Informatics (ISMII)*, pp. 96–103. IEEE, 2021.

[27] H. Veeraraghavan and J. Jiang, "Deep learning from small labeled datasets applied to medical image analysis,". in *State of the Art in Neural Networks and their Applications*, pp. 279–291, Elsevier, 2021.

[28] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra *et al.*, "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.

[29] C. Xie, Y. Wu, L. v. d. Maaten, A. L. Yuille, and K. He, "Feature denoising for improving adversarial robustness," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 501–509, 2019.

[30] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2730–2739, 2019.

[31] J. Xiong, D. Yu, S. Liu, L. Shu, X. Wang, and Z. Liu, "A review of plant phenotypic image recognition technology based on deep learning," *Electronics*, vol. 10, no. 1, p. 81, 2021.

[32] H. Zhang and J. Wang, "Defense against adversarial attacks using feature scattering-based adversarial training," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[33] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama, and M. Kankanhalli, "Attacks which do not kill training make adversarial learning stronger," in *International conference on machine learning*, pp. 11278–11287. PMLR, 2020.

[34] R. Zhang, T. Che, Z. Ghahramani, Y. Bengio, and Y. Song, "Metagan: An adversarial approach to few-shot learning," *Advances in neural information processing systems*, vol. 31, 2018.

Biography

You-kang Chang was born in 1994. He is a doctor student at Lanzhou University of Technology. His major research field is adversarial attacks and defense adversarial attacks. E-mail: 2507576651@qq.com.

Hong Zhao was born in 1971. He is a professor and a supervisor of doctor student at Lanzhou University of Technology. His major research field is System modeling and simulation, deep learning, natural language processing. E-mail: zhaoh@lut.edu.cn.

Wei-jie Wang was born in 1994. She is a doctor student at Lanzhou University of Technology.Her major research field is speaker recognition. E-mail: 1132744259@qq.com