

Research on image-oriented adversarial sample attack and defense algorithms

by Hongliang Yu

Submission date: 12-Sep-2023 05:02PM (UTC+0100)

Submission ID: 212493955

File name: 40826_Hongliang_Yu_Research_on_image-oriented_adversarial_sample_attack_and_defense_algorithms_511263_1097506417.docx (11.24M)

Word count: 13511

Character count: 77539



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Research on image-oriented adversarial sample attack and defense algorithms

by
Hongliang Yu

Supervisor: Michael Dacey

September 2023

Table of Contents

Abstract	4
1. Introduction	4
1.1 Research Background	4
1.2 Research Aim	6
1.3 Research Objectives	7
2. Literature Review	7
2.1 Secondary research	7
2.1.1 State of the Art in Counter Attack Research	8
2.1.2 State of the art in adversarial defense research	11
2.1.3 Status of research on other technologies	11
2.2 Other researchers view of topic or concept	12
2.3 Different techniques/algorithms	12
2.3.1 Fast gradient symbolic algorithm	12
2.3.2 Iterative-based algorithm for fast gradient notation	13
2.3.3 Deepfool algorithm	13
2.4 Research methods and test types	14
2.5 Engage with the research promotes collaboration	15
3. Research and Development Methodology	15
3.1 Investigate research method(Research)	15
3.2 Research process (Research)	16
3.2.1 Research structured	16
3.2.2 Focus group size and makeup	16
3.2.3 Validate and verify results	16
3.2.4 Statistical approach to analyse results	16
3.3 Development Methodology (Development)	17
3.3.1 Investigate different development methods	17
3.3.2 Requirements: discuss use cases vs user stories vs numbered list	17
3.3.3 Class diagrams, sequence diagrams	17
3.3.4 Implementation and Testing	17
4. Design and Implementation (development, experiment)	17
4.1 Color space based algorithm for white box adversarial sample generation	17
4.1.1 Requirements analysis	17
4.1.2 Overall framework of color space based white box adversarial sample generation algorithm	18
4.1.3 Color space white box adversarial sample generation algorithm based	19
4.2 Adversarial Sample Defense Algorithm Based on Feature Denoising	23
4.2.1 Requirements analysis	23
4.2.2 Structure of the CRGAN	24
4.2.3 Perturbed dissolved layer	25
4.2.4 Loss function	26
5. Design/ Data collection (survey or review)	27
6. Data Analysis (survey or review)	28
7. Results and Discussion	29

8. Conclusions and Recommendations	38
9. Reflection	38
References	39
Appendix	42

Abstract

In recent years, with the iteration of computer hardware and the emergence of various open-source deep learning frameworks, artificial intelligence technology based on convolutional neural network models has made relatively breakthrough progress in various fields, triggering a new round of industrial revolution. However, with the breakthrough of deep neural network technology, the security problems caused by artificial intelligence have also been revealed. Relevant researchers have found that neural network models are susceptible to false samples. Adding a well-designed perturbation to a clean sample can make the classifier model output false results with high confidence. This imperceptible perturbation is called adversarial perturbation. The emergence of adversarial samples poses a further threat to the implementation of AI systems in finance, transportation, and healthcare, where data security is more stringent. This paper focuses on the above technical background, and conducts an in-depth study on the adversarial sample generation algorithm under different scenario conditions, and the main research work is as follows:

A white-box adversarial sample generation algorithm based on color space is proposed to address the problem of generating more stealthy adversarial samples in white-box scenarios. The algorithm takes into account the crucial role of contour, shape, and other features in both the human visual system and artificial intelligence recognition systems for classification tasks, considering color as only an auxiliary feature. Starting from visual recognition preferences, by altering the relationships between pixel values, one can achieve satisfactory visual effects. Based on this idea, a color domain mapping algorithm is proposed, which segments the values in the pixel domain and maps the original pixel domain to other pixel domains using a non-norm distance strategy through feature space similarity editing. This algorithm aims to find adversarial samples in the color space.

A feature denoising-based adversarial sample defense algorithm is proposed to address the issue of enhancing the transferability of defense models in black-box scenarios. Inspired by Conditional Generative Adversarial Networks (CGAN), we propose a defense model that treats the mask as a condition. This model reconstructs images using the structure of an encoder-decoder and the principles of adversarial learning. Additionally, considering the vulnerability of adversarial perturbations, we employ a noise-dissolving layer that is correlated with the perturbation to eliminate a portion of the disturbance. When designing the loss function, we take into account that slight perturbations can be amplified in the intermediate layers of neural networks. Therefore, we introduce a perceptual loss function to eliminate deeper-level interferences, greatly increasing the transferability of the defense model.

1. Introduction

1.1 Research Background

In recent years, with the development of computer technology and network technology,

artificial intelligence systems based on neural network models have experienced rapid growth. AI systems based on neural networks require a large amount of data to train the models, and the widespread use of the Internet has made data acquisition easier. Furthermore, advancements in hardware technology and improved high-performance computing frameworks have made deep neural network models with millions of parameters increasingly common. Deep neural networks have quickly become a focal point of attention due to their excellent performance. Compared to traditional machine learning methods based on manually designed features, deep neural network models demonstrate superior performance. From AlexNet^[1] to ResNet^[2], the depth and complexity of deep neural network models have increased, leading to improved network performance. Convolutional neural network models have been widely applied in various computer vision tasks, such as image classification^[1,2], image segmentation^[3-5], and object detection^[3,6,7]. Currently, convolutional neural networks have achieved exceptionally high performance in computer vision tasks, surpassing human performance. Meanwhile, deep neural network models are gradually becoming integrated into people's lives. Examples include facial recognition based on facial information, machine translation for foreign languages, and autonomous driving. Deep neural network models are driving more real-world applications and addressing complex problems, becoming an integral part of our lives.

Despite the widespread application and success of deep neural network models in performing various complex tasks, the security of these models has received little attention from the academic community. In 2013, Szegedy, Goodfellow^[8], et al. first pointed out that neural network-based AI models, while exhibiting excellent performance, are also susceptible to security threats and vulnerable to adversarial attacks. Adversarial samples are generated by adding imperceptible perturbations to original samples. Although these small perturbations have no impact on the human visual system, they can cause image classification models to make incorrect predictions, often with high confidence. What's worse is that research has shown that adversarial samples exhibit a certain level of transferability. This means that if an adversarial sample can deceive one machine learning model, it can still successfully deceive other machine learning models with high confidence. This implies that attackers can launch attacks on deep neural networks even without accessing the underlying models. The robustness and reliability of neural networks are overshadowed by adversarial samples. Moreover, this phenomenon is not limited to the field of computer vision^[9-11] but also occurs in text processing^[12-14] and natural language processing^[15-17]. Currently, as neural network systems further develop and become integrated into our lives, deep learning technology has become closely related to us, playing a practical role in industrial and civil domains. For example, assisting drivers in autonomous driving systems^[18], evaluating customers' credit risks in banking supervision systems, diagnosing patients' medical conditions^[19], and detecting malicious software in network security^[20]. In these areas, the robustness and security of neural network systems are crucial. Adversarial samples pose significant challenges to the practical application and development of artificial intelligence, hindering the progress of deep learning. To

promote the secure application of AI technology, numerous scholars and industry professionals are devoted to exploring the robustness of neural networks and the mechanisms of adversarial samples.

In recent years, adversarial samples have become a hot topic of research. Researchers are exploring the security of neural networks from two different aspects. On one hand, they investigate how adversarial samples can better attack deep learning models in various scenarios. On the other hand, they design corresponding algorithms to detect and mitigate adversarial attacks. Researchers have shown great interest in the mechanisms of neural network models from both attack and defense perspectives. They aim to enhance the stealthiness and generalizability of adversarial samples at the attack level, and to make defense models more universal at the defense level. Research in these two aspects mutually reinforces each other, driving the study of robustness in neural network models. The goal is to improve the security of current neural network models, reduce the impact of adversarial samples on real-world applications and daily life. Additionally, this research contributes to researchers' understanding of the mechanisms of deep learning models, providing better explanations and improvements in the robustness of deep learning models. It further advances the field of artificial intelligence and holds significant importance for the development and application of deep learning technology in real-world scenarios.

1.2 Research Aim

The choice of studying adversarial sample attack and defense algorithms for the graduation thesis stems from a personal interest and curiosity in the field of cybersecurity, a recognition of its contemporary relevance, and the aspiration to contribute to the academic and professional domains. By delving into this topic, we aim to satisfy our intellectual curiosity, acquire in-depth knowledge and expertise, enhance analytical and problem-solving skills, and make a meaningful impact on the research community by informing the development of robust defense strategies in the ever-evolving landscape of cybersecurity.

This project focuses on the vulnerability of neural networks to adversarial attacks and the associated security risks. It explores adversarial attack and defense techniques for neural networks, as well as investigates the mechanisms of neural network models. Research in both attack and defense aspects mutually reinforces each other, driving the study of robustness in neural network models. The goal is to enhance the stealthiness and generalizability of adversarial samples at the attack level, and to make defense models more universal at the defense level. This research aims to improve the security of current neural network models and reduce the impact of adversarial samples on real-world applications and daily life. Furthermore, it contributes to researchers' understanding of and insight into the mechanisms of deep learning models, facilitating better explanations and improvements in the robustness of deep learning models. This, in turn, advances the field of artificial intelligence and holds significant importance for the development of deep learning technology in real-world scenarios.

1.3 Research Objectives

The literature review is an essential component of the research project, serving multiple purposes. It allows for the identification of knowledge gaps and research questions, establishes a theoretical foundation, aids in selecting appropriate methodologies, examines empirical evidence and findings, develops a conceptual framework, and identifies key researchers and their contributions. By conducting a thorough review of the existing literature, this research aims to build upon established knowledge while filling gaps and contributing new insights to the field of adversarial sample attack and defense algorithms.

The implementation phase of the research project is crucial as it allows for the practical application and validation of the proposed algorithms and methodologies. Through implementation, empirical evaluations can be conducted to assess the research findings, identify practical challenges and limitations, and provide feasible insights and recommendations. It ensures the reproducibility of the research and narrows the gap between theory and practice. By successfully implementing the research findings, this study aims to facilitate the practical development and deployment of robust adversarial sample attack and defense algorithms, thereby advancing the field and addressing real-world cybersecurity challenges. The research will explore the design of the algorithm generation process to enhance the stealthiness and attack success rate of generated adversarial samples. It will investigate the design of network structures that exhibit universal defense capabilities in large-scale networks and complex data scenarios. Furthermore, it will delve into the design of feature denoising algorithms to remove adversarial features from adversarial samples while preserving the data characteristics consistent with genuine samples, thus improving the robustness of neural networks.

Through this research, the aim is to acquire a comprehensive understanding of adversarial sample attack and defense algorithms, identify vulnerabilities and limitations in existing approaches, develop novel methodologies, evaluate their performance, and provide practical insights and recommendations. By contributing to the advancement of knowledge in the field, this study strives to enhance the security and reliability of machine learning models against adversarial attacks. Ultimately, the goal is to strengthen cybersecurity practices by addressing the challenges posed by adversarial samples.

2. Literature Review

2.1 Secondary research

Since the inception of adversarial samples, they have become a research hotspot for research teams worldwide. The study of adversarial samples mainly consists of two parts: adversarial attacks and defenses. On one hand, adversarial attack techniques aim

to construct adversarial samples from different perspectives to attack robust models and achieve desirable results. On the other hand, adversarial defense techniques aim to detect or reconstruct adversarial samples in a generalizable manner to enhance the robustness of models. Adversarial attacks and defenses engage in a game-like interaction and mutually promote each other. Researching adversarial attack algorithms helps in understanding black-box models of neural networks and explaining the intrinsic characteristics of deep learning. Improving the performance of adversarial sample defense algorithms relies on a diverse set of attack algorithms, and studying adversarial sample defense algorithms contributes to the construction of robust models. Both aspects have witnessed a considerable amount of research, and this subsection provides a brief introduction to various attack and defense algorithms, analyzing the differences among different techniques.

2.1.1 State of the Art in Counter Attack Research

With knowledge of the internal structure, hyperparameters, and other details of the attacked model, the attacker can analyze this information and directly compute adversarial samples. On the other hand, black-box attacks occur when the attacker has limited or no knowledge about the inner workings of the target model. They can only indirectly obtain model information by observing external inputs and outputs, and then use this information to create adversarial samples. In the following section, we will introduce classical attacks in the white-box and black-box domains.

In 2014, researchers from Google Research, including Goodfellow et al. ^[8], proposed a fast and efficient adversarial sample generation algorithm called the Fast Gradient Sign Method (FGSM). It involves calculating the gradient of the classification loss function with respect to clean samples, adding perturbations in the direction of gradient ascent, and generating adversarial samples. This method has gained widespread application due to its ability to achieve good results by computing the loss function gradient only once. However, due to the single gradient computation, its attack capability is limited, leading to the emergence of a series of derivative methods in subsequent research.

The single-step FGSM attack has led to the development of multi-step FGSM attacks. Kurakin et al. ^[21] optimized the principles of the FGSM method and proposed the Iterative FGSM (I-FGSM). I-FGSM performs multiple iterations of the FGSM attack, successfully applying adversarial perturbations in the physical world. In 2017, Madry et al. ^[22] introduced the Projected Gradient Descent (PGD), which is essentially an iterative version of the FGSM algorithm. This method recognizes that a single iteration may not explore the optimal perturbation direction. Compared to regular FGSM, which computes the gradient direction only once, PGD performs multiple iterations, taking small steps in the gradient direction at each iteration while constraining the perturbations obtained in each step to ensure that the pixel values of the adversarial samples remain within a specified threshold.

Although the I-FGSM method enhances the attack strength in adversarial attacks, the

generated adversarial images are prone to overfitting to the source model, resulting in a decrease in the attack success rate when transferring these samples to other models. To address this issue, Dong et al. ^[23] proposed the MI-FGSM attack in 2018, which improves upon the I-FGSM method. The basic idea is to introduce momentum during the perturbation process of the original image to help the algorithm escape from local minima and generate stronger adversarial samples. Specifically, during each iteration, the gradient is computed and signified, and then momentum from previous iterations is added and updated. Finally, perturbations are added to the updated image. The momentum factor records the gradients at each iteration, thereby increasing the stability and robustness of the algorithm and improving the attack success rate. The MI-FGSM method demonstrates superior performance on black-box models compared to other FGSM-like algorithms while achieving high accuracy rates on white-box models.

In order to compute the minimal perturbation that can deceive neural networks, Moosavi-Dezfooli et al. ^[24] proposed an efficient hidden adversarial sample generation algorithm called DeepFool after studying decision boundaries. This method assumes that the input real sample lies within the decision plane of the neural network. It utilizes the linearity property to compute the distance between the input real sample and the nearest decision boundary. The attack is iteratively conducted. The DeepFool algorithm suggests that due to the local linearity of neural networks, if an adversarial sample exists in a space that is very close to the original sample distribution, it can be easily computed using linear algebra to be located at the classification boundary. The non-linear properties of neural networks can be approximated by multiple linear estimations to find adversarial samples located at the classification boundary.

In white-box attacks, optimization attacks are an important attack method. Carlini and Wagner proposed the well-known C&W attack ^[25]. The core of this method for generating adversarial samples is to find the adversarial perturbation that minimizes the objective function under box constraints. It aims to achieve high accuracy while maintaining good stealthiness. Specifically, C&W treats the adversarial sample as a variable and uses optimization algorithms to minimize the distance while causing misclassification in the target neural network model. Experimental results have shown that targeted distillation defense is almost ineffective against these three attacks, making C&W attack one of the most powerful targeted attack algorithms.

The concept of adversarial patches was proposed by Brown et al. ^[26] in 2017. Instead of restricting perturbations to the entire input image, attackers replace a part of the original image with a patch, confining the perturbation within the patch to deceive the classifier. This attack method is often used in real-world scenarios, and the adversarial glasses proposed by Sharif et al. ^[27] in 2016 are an example of adversarial patches. By adding patches to the frames of faces, the wearer can be misidentified by face recognition systems. Adding adversarial patches to traffic signs can pose significant security risks to image recognition systems used in autonomous driving.

Reference ^[28] investigated the vulnerability of neural network classifiers to rotation and translation and proposed a non-norm attack. This method constructs a rotation-translation matrix parameterized by a set of adjustable parameters and optimizes these parameters to perform translations and rotations on the original image. Similarly, Reference ^[29] demonstrated a non-norm attack by introducing an "unrestricted" semantic perturbation. They formulated the problem of generating such images as a constrained optimization problem and proposed an adversarial transformation based on the shape bias characteristics of the human visual system. Effective and realistic adversarial samples are generated by manipulating semantic visual and texture descriptor operators. In this way, the modified images are indistinguishable from clean images in terms of the human visual system. The method adjusts the hue and saturation values in the color space to generate smooth and natural adversarial images.

Estimating the risk level of adversarial samples is crucial for deploying machine learning models in the real world. A popular method for physical-world attacks is the "sticker attack" strategy, but it has limitations, such as difficulty in accessing the target or using effective color printing. Reference ^[30] investigated a novel optical adversarial sample where the perturbation is generated by a very common natural phenomenon: shadows. This approach achieves natural physical-world adversarial attacks by simulating the position of shadows with an appropriate polygon and mapping the image to the LAB color space to adjust the luminance component and generate the shadow. The study reveals that shadows can be deceptive and mislead machine learning-based visual systems to make incorrect decisions.

The most common black-box attack method includes transfer attacks. Single-step gradient-based methods are difficult to apply directly to practical black-box models. Therefore, gradient accumulation methods have emerged to improve transferability by designing new gradient update strategies ^[31]. The MI-FGSM algorithm is a momentum-based iterative gradient method that improves the gradient descent direction during iterations. By adding a momentum factor, it avoids getting stuck in local maxima during optimization, thereby preventing adversarial samples from overfitting to the source model. Lin et al. ^[32] proposed two new methods to enhance the portability of adversarial instances. Inspired by the superiority of the Nesterov accelerated gradient (NAG) method over momentum in traditional optimization, they incorporated it into the iterative attack process. This effectively improves the transferability of adversarial samples and is referred to as the NI-FGSM method. They also proposed a scale-invariant attack method combined with NI-FGSM, constructing a powerful attack.

Utilizing input diversity to transform image samples is also an effective method for transfer attacks. Xie et al. ^[33] proposed a diversity-induced fast gradient sign method that randomly adjusts the size and padding of images during iterations with a certain probability to alleviate overfitting and improve the transferability of adversarial samples. Wu et al. ^[34] introduced a novel adversarial transformation network based on the idea of adversarial training for data augmentation. The adversarial transformation

network primarily suppresses the attack of adversarial samples and trains clean samples so that the generated adversarial samples can withstand the strongest data transformations in defense while retaining their attack effectiveness. Existing data transformation methods can weaken the attack of adversarial perturbations, but the styles of these data transformations are fixed, such as flipping, scaling, and color changes, making them relatively fixed and single in terms of data transformation. The authors trained to find the strongest data transformation method that can effectively suppress the attack of adversarial samples. After obtaining the strongest data transformation function, they trained to generate adversarial samples, resulting in samples with more diversified characteristics.

2.1.2 State of the art in adversarial defense research

To counter the threat of adversarial perturbations, researchers have proposed various defense strategies. These defense strategies can be broadly categorized into two different approaches: passive defense (adversarial attack detection) and active defense (enhancing model robustness).

Researchers believe that traditional denoising techniques can remove a significant portion of adversarial perturbations. However, due to the high complexity and black-box nature of neural networks, some residual noise that is not removed may be amplified when passing through deep neural network models, posing a threat to the input samples. To address this, neural networks can be used for data preprocessing. Liao et al. ^[35] proposed a deep denoising neural network called the High-Level Representation Guided Denoiser (HGD), which aims to eliminate noise in adversarial samples at the feature level. HGD modifies the relevant loss functions to minimize the distance between clean and adversarial samples at intermediate layers, mitigating the error amplification effect in deep neural networks. Similarly, Jia et al. ^[36] designed a compressed reconstruction network called ComDefend to defend against adversarial samples. ComDefend consists of three main defense modules: a noise removal module, a feature reconstruction module, and a model fine-tuning module. Before processing the input adversarial samples, ComDefend applies the noise removal module to remove common perturbations such as salt-and-pepper noise. Then, the feature reconstruction module is applied to the denoised image to remove the perturbations added by adversarial attacks. Finally, the model fine-tuning module is used to fine-tune the target model to improve its performance on adversarial samples.

2.1.3 Status of research on other technologies

There is a growing variety of attacks and defense mechanisms targeting neural network models, and research on the security of neural network models is no longer limited to the means of attack and defense. It is now more focused on studying the data aspect. In this section, we will briefly introduce the latest related research.

Backdoor attacks are a new type of attack paradigm that aims to add triggers to a model during its training process and change the corresponding labels' categories. The attacked model can correctly classify and predict normal data inputs, but when inputs contain triggers, the model outputs the label specified by the attacker. Gu et al. [37] first introduced the concept of backdoor attacks in deep learning models. Traditional backdoor attacks involve selecting a portion of the original data, adding triggers to it, and then incorporating this data into the training set for the model to learn. However, due to the modification of labels, which results in significant loss function values and is easily distinguishable, this approach has limited success rates when the contaminated data proportion is low.

In the future, applications related to deep learning will gradually be implemented. Therefore, research on the robustness of neural networks has become crucial. On one hand, it promotes the development of artificial intelligence in a safer manner. On the other hand, it helps researchers gain a deeper understanding of the black box nature of neural networks and consider the influence of different features on neural networks to varying degrees. This enables them to better grasp new opportunities in the development of artificial intelligence and drive its practical implementation.

2.2 Other researchers view of topic or concept

Researchers have shown a strong interest in the study of adversarial attack and defense algorithms, recognizing the importance of understanding and mitigating the threats posed by adversarial attacks to machine learning models. They acknowledge the evolving nature of attack techniques and emphasize the development of effective defense mechanisms. Researchers actively explore various strategies and approaches to enhance the robustness and security of models against adversarial manipulation. Despite the challenges and open issues that remain, researchers demonstrate a proactive attitude towards advancing the field and safeguarding the reliability of machine learning models in the face of adversarial attacks.

2.3 Different techniques/algorithms

2.3.1 Fast gradient symbolic algorithm

Goodfellow et al proposed an effective algorithm for adversarial sample generation, the Fast Gradient Symbol Method (FGSM). This method calculates the gradient of the loss function after the input of normal samples to the neural network model, obtains the direction that allows the loss function to increase as the direction of the perturbation, and finds the size of the perturbation using the sign function and the paradigm constraints. The main idea of the algorithm is that the neural network model is trained to fit the model in the direction that makes the gradient of the target loss function decrease. The antagonistic perturbation is obtained by finding the direction with the largest change in the gradient, and this perturbation is added to the normal samples to generate the antagonistic samples that make the model misclassified. By adding the

corresponding perturbation globally to the normal samples, it is a fast and efficient algorithm to generate the antagonistic samples, and the perturbation is linearly amplified in the high-dimensional feature space due to the error amplification effect, which is enough to have a significant impact on the discriminative results of the neural network classifier model. Eq. 2-1 shows that adding the perturbation δ to the normal sample can result in an adversarial sample generated by the attacked model:

$$\begin{aligned} x_{adv} &= x + \delta, f(x_{adv}) \neq f(x) \\ \delta &= \varepsilon * \text{sign}(\nabla_x J(\theta, x, y)) \end{aligned} \quad (2-1)$$

Where, δ denotes the added antagonistic perturbation, ε denotes the step size of the perturbation limit, the larger the step size, the more pronounced the perturbation, $J(\cdot, \cdot)$ denotes the loss of the neural network with respect to the input image x and the class label y , ε denotes the step size of the perturbation limit, sign denotes the sign of the gradient, and $\nabla_x J(\cdot, \cdot)$ denotes the partial derivatives of the loss function of the training model with respect to the input samples x is obtained by back-propagating the $f(\cdot)$ denotes the neural network model.

2.3.2 Iterative-based algorithm for fast gradient notation

Ordinary fast gradient symbolic algorithms use only one iteration in generating the adversarial samples, which suffers from insufficient success rate of the attack and imprecise addition of perturbations. Based on this problem, Kurakin et al. proposed an iteration-based fast gradient symbolic algorithm (I-FGSM), to address the shortcomings of the FGSM algorithm, which updates a large step in the direction of the largest change in the gradient, iterates several times in smaller steps, and then re-calculates the direction of the gradient after each iteration. The cropped intermediate quantity is located in the value domain of the original image. The related expression is shown in 2-2:

$$x_0^* = x, x_{t+1}^* = \text{Clip}_{x, \varepsilon} \{x_t^* + \varepsilon \cdot \text{sign}(\nabla_x J(x_{t+1}^*, y_{true}))\} \quad (2-2)$$

where x is the input normal sample, x_t^* denotes the antagonistic sample generated by the t th iteration, and $\text{Clip}_{x, \varepsilon}(\cdot)$ denotes cropping the antagonistic sample into the original image region.

2.3.3 Deepfool algorithm

Gradient-based adversarial sample generation algorithm is a simple and efficient adversarial sample generation algorithm by adding the global perturbation with the largest change in the gradient direction to the normal input sample when generating the adversarial sample, but such methods generate larger perturbations. In order to add smaller perturbations, Deepfool algorithm is proposed. Deepfool algorithm is based on the idea of hyperplane classification, which is an attack method based on decision boundaries, and can accurately calculate the perturbation value. In the case of binary classification, assuming that the decision boundary of the classifier is linear and the two sides of the hyperplane correspond to different classification results, for the sample x

on one side wanting to add a perturbation to make the classifier classify it into another category, then it is only necessary to update the sample x to the other side of the hyperplane, it is only necessary to find the shortest distance of the normal sample x from the hyperplane, so that the sample x , together with the minimum distance r to the hyperplane, can cross the hyperplane. distance r can cross the hyperplane and reach the other side of the hyperplane to find the antagonistic sample. The relevant expression is as follows:

$$\|r\| = \frac{Ax_0 + Bx_1 + C}{\sqrt{A^2 + B^2}} \quad (2-3)$$

Eq. 2-4 demonstrates the extension of a 2D binary classifier to a multidimensional binary classifier:

$$\begin{aligned} \mathbf{r}_*(\mathbf{x}_0) &:= \arg \min \|\mathbf{r}\|_2 \\ &\text{st. } (f(\mathbf{x}_0 + \mathbf{r})) \neq \text{sign}(f(\mathbf{x}_0)) \\ &= -\frac{f(\mathbf{x}_0)}{\|\mathbf{w}\|_2^2} \mathbf{w} \end{aligned} \quad (2-4)$$

where r denotes the minimum perturbation found and w denotes the normal vector of the decision boundary.

In the case of multidimensional binary classification, for high-dimensional neural network models, the decision boundary of the classifier is nonlinear. Since the decision function is highly nonconvex and nonlinear, to compute the shortest distance from the normal input sample x to the nonlinear curve in this case, a heuristic iteration is needed to find the projection point of the normal input sample x on the curve. Segmented linear function is used to approximate the target decision function, in the process of iteration will be obtained in the i the iteration of the antagonistic sample x_i points on the nonlinear classifier using a first-order Taylor expansion can be obtained as a straight line, after a number of iterations to find the antagonistic samples in the domain of the classifier.

2.4 Research methods and test types

Maximum perturbation, attack success rate, and structural similarity are chosen as the final evaluation metrics against attacks and defenses.

To ensure that the generated adversarial examples remain visually similar to the original images, we limit the spatial distance between the adversarial examples and the real examples by specifying a maximum allowable perturbation. This means that the perturbation applied to each pixel of the image must be kept within a certain range.

In this study, we use two different distance metrics, namely the Chebyshev distance and the Euclidean distance, to measure the distance between each adversarial example and its corresponding real example, denoting the maximum allowable perturbation by the symbol ϵ .

The Attack Success Rate (ASR) is a metric used to measure the effectiveness of an

adversarial attack. It is calculated as the ratio of the number of misclassified samples after the attack to the total number of samples. In this study, we assume that there are N different classes of original data samples. We use X_i^a to denote the number of samples in class i that can be correctly identified by the target neural network classification model before the attack, and X_i^b to denote the number of samples in class i that can be correctly identified after the attack. The ASR is then calculated using the following formula:

$$ASR(X^a, X^b) = \frac{\sum_{i=1}^n X_i^b}{\sum_{i=1}^n X_i^a} \times 100\% \quad (2-5)$$

Structural similarity: Structural similarity expresses the similarity of two images. To measure the stealthiness of the adversarial perturbation, we compare the same image classification models under the same adversarial perturbation strength to damage the image structure. If the attack success rates of the two adversarial generation algorithms are comparable, the adversarial algorithm with greater structural similarity is more similar to the original image and the more stealthy the algorithm is, and the value is denoted by $ssim$ in this paper.

2.5 Engage with the research promotes collaboration

Engaging with research is crucial as it broadens perspectives, builds on existing knowledge, facilitates collaboration, helps stay updated with emerging trends, and provides validation and feedback. It allows researchers to gain insights from diverse viewpoints, refine research questions, and identify novel research directions. By actively participating in the scholarly discourse, researchers can establish connections, share ideas, and collaborate on joint projects. Staying informed about the latest advancements ensures that researchers incorporate the most relevant methodologies and approaches into their work. Engaging with research enhances the rigor and quality of the work, contributes to the growth of the field, and strengthens the overall research outcomes.

3. Research and Development Methodology

3.1 Investigate research method(Research)

When selecting a research methodology for the study of adversarial sample attack and defense algorithms, it is critical to consider the research objectives, methodological considerations, and the nature of the research problem. When selecting a methodology, extensive reading of the literature for the relevant research point selects the methodology that is most relevant to the research content of this paper and contributes to the level of knowledge in the field of adversarial sample attack and defense algorithms.

3.2 Research process (Research)

3.2.1 Research structured

The research in this paper is divided into the following sections. The first part is the introduction, which focuses on the significance of the study, the purpose of the study and the importance of the study. The second part is the current state of research, which mainly introduces the main development of adversarial sample attack and defense algorithms, and the advantages and disadvantages of each algorithm. The third part focuses on the overall research design and methodology chosen for this paper. The fourth section focuses on the implementation of the methodology of this paper, followed by a description of the collection of the dataset, including an explanation of the methods and techniques used to collect the data, such as surveys, interviews, or experiments, and the rationale for the choice of these methods and techniques. Finally, the results of the experiments are analyzed and a summary of the full paper is given.

3.2.2 Focus group size and makeup

This paper is mainly completed by me independently. Including the preliminary research, learning the current situation at home and abroad and related technology, learning to use the relevant framework. And complete the experiments afterwards, and analyze the experimental results.

3.2.3 Validate and verify results

The research results obtained in this study on investigating research methods for adversarial sample attack and defense algorithms were subjected to a rigorous process of validation and verification. To ensure their accuracy and reliability, a comprehensive evaluation and comparison approach was employed. In order to improve the feasibility of our experimental results, we compared our algorithm with multiple datasets and multiple models employed, and compared our algorithm with multiple algorithms on multiple evaluation metrics.

3.2.4 Statistical approach to analyse results

In this paper, different algorithms are presented in a table with relevant metrics on different datasets and different models, and the size of the metrics responds to the strengths and weaknesses of the algorithms. And the related curve graph is used to compare the algorithms to react to the advantages and disadvantages of the algorithms.

3.3 Development Methodology (Development)

3.3.1 Investigate different development methods

We will analyze and compare the iterative class-based and optimization-based approaches to choose our attack. We will go from reconstruction and detection to select our defense approach. Discuss how this applies to the development of counter sample attack and defense algorithms. The characteristics of each approach will be discussed in depth to help select the most suitable approach for our project.

3.3.2 Requirements: discuss use cases vs user stories vs numbered list

Deep learning and neural networks have been developing rapidly in recent years, with impressive results in a variety of fields, including computer vision, autonomous driving, and natural language processing. However, neural networks also have a drawback, which is that they can be easily deceived by a kind of false data known as adversarial samples. Research on adversarial samples has focused on their concealment, attack success rate and defense success rate. Our research is motivated by how to go about obtaining adversarial samples that are more stealthy and have stronger attacks, and how to go about obtaining defense methods that have stronger defenses.

3.3.3 Class diagrams, sequence diagrams

We will use structure diagrams to describe the functions of each module, including the specific details of each function, and tables to describe the various steps of the algorithm. Visualization methods are used to better understand the workings and flow of the algorithm.

3.3.4 Implementation and Testing

The experimental equipment and operating environment used for the experiment are CPU Core(TM) i7-10700 CPU, GPU Nvidia Geforce 3090 cuda 11.3, operating system Windows10 or Ubuntu20.04, Pytorch version Pytorch1.16 and CUDA version 11.3.

4. Design and Implementation (development, experiment)

4.1 Color space based algorithm for white box adversarial sample generation

4.1.1 Requirements analysis

In the previous sections of this paper, we introduced classical gradient-based attack methods that are simple and efficient and have obtained good attack rates. These methods, although effective, counter perturbations to the image superposition are only global perturbations to the output of the last layer of the neural network model. The global perturbation brings a large number of inefficient pixels, which in the case of a

large perturbation step leads to a large change in the image structure and will be more easily detected by the human eye visual system. In the traditional method, the imperceptibility is limited by the L_p paradigm of the perturbation in the RGB space, and the generated antagonistic samples constrain the maximum value of the distance of the perturbation to a certain range, after such a limitation, the amplitude of the perturbation will be limited, and the strength of the attack is very dependent on the amplitude of the perturbation, and the limitation of the amplitude of the perturbation makes the performance of the attack greatly reduced.

4.1.2 Overall framework of color space based white box adversarial sample generation algorithm

The whole algorithmic framework is mainly divided into two modules, and the technical means under different parts are summarized as shown in Figure 4-1. In the color mapping stage, the pixel adjustment function is selected to change the pixel values; in the feature similarity editing stage, the features in the feature space layer are separated; in the composite adversarial attack stage, it is proposed to add two different adversarial perturbations to generate an adversarial perturbation that combines the advantages of both adversarial samples and outputs them for the adversarial attack.

The counter-attack algorithm in the white-box counter-attack scenario aims to improve the stealth of the algorithm and still maintain a better attack effect, in Figure 4-1, this paper proposes to introduce two modules such as pixel-domain mapping, feature similarity editing, and so on, which will combine the localized perturbation counter-attack and pixel-transformation unrestricted attack to improve the effectiveness of counter-attacking the samples, and in the following this paper, we will respectively introduce each of the two sub-modules of the individual functions:

(1) "Color Domain Mapping Module": In classification recognition tasks, whether for human eye vision system or artificial intelligence recognition system, features such as contour and shape play a crucial role, while color is only an auxiliary feature. From the perspective of visual recognition preference, the human eye is insensitive to the change of color compared with the change of image structure, and the change of pixel value relationship can also achieve a good observation effect. Therefore, this paper proposes the "Color Domain Mapping Module", which is able to obtain rich transformed samples that meet the vision of the human eye by setting the pixel adjustment function and obtaining the two-by-two mapping relationship between the pixel values, and by drastically modifying the pixel values of the samples.

(2) "Feature Similarity Editing Module": In order to meet the requirements of the "Pixel Domain Mapping Module", we propose the "Feature Similarity Editing Module", which is designed to meet the requirements of the human eye while significantly modifying the pixels. Therefore, this paper proposes the "feature similarity editing module", which extracts the sample image into the feature space. By making full use of the feature representation of the least similar samples in the feature space, narrowing

the feature gap between the antagonistic samples and the least similar samples, and enlarging the feature gap between the antagonistic samples and the normal samples, the normal samples and the antagonistic samples can be distinguished more clearly in the feature space, so that the semantically-aware antagonistic samples can be searched for directly from the feature space in this way.

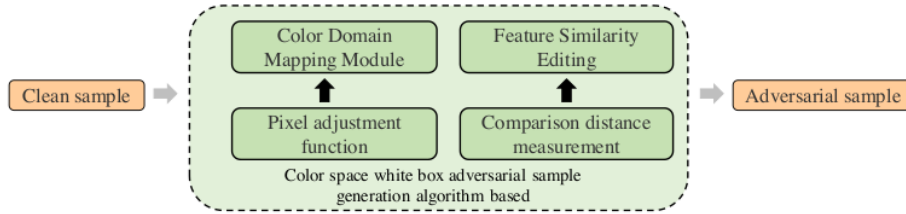


Figure 4-1 Overall framework flow chart of color space attack algorithm

4.1.3 Color space white box adversarial sample generation algorithm based

Deep neural networks have made outstanding achievements in the field of artificial intelligence, and the principle behind their working mechanism originates from the sensory field of the human eye's visual system. In the human eye visual system, light acts on the human eye and activates the corresponding receptor cells which are then transformed to form vision. The outline, color, size and other factors of an object are perceived through the visual system, and among the many factors, the outline plays a crucial role, which is a visual preference. Similarly, with the help of visualization of the feature maps of the middle layer convolutional neural network, it can be seen that the network is more focused on extracting shape information while extracting features. Whereas color is only an auxiliary feature, both humans and neural networks are insensitive to color features, and significant modification of the image pixel values can also produce better visual results. This subsection introduces a microscopic color-domain mapping algorithm in terms of visual recognition preferences, where the hyperparameters in the operator are updated with gradients via backpropagation of the neural network model.

More intuitively, the algorithm utilizes a pixel tuning function, which is a two-dimensional linear segmented function with different hyperparameters controlling the slope of each segmented value, and the value of each pixel point is mapped by the function to other different pixel values. Since the 8bit normal image pixel values ranging from 0-255 are mapped to 0-1 in the neural network model computation, the pixel adjustment function divides the 0-1 values into L intervals of equal length, each of which has a slope determined by the hyperparameter φ_i . Because the range of intervals of pixels before and after the function mapping does not change, the sum of L hyperparameters φ_i is 1. The more the number of segments is, the finer the mapping of the color model is, and it is easier to fit the original image smoothly. As shown in Figs. 4-2, the left side represents the process of pixel single mapping after adjusting the

operator, and the right side is the original and the adjusted image. It can be seen that in different segments controlled by different hyperparameters, the pixels change in the color space.

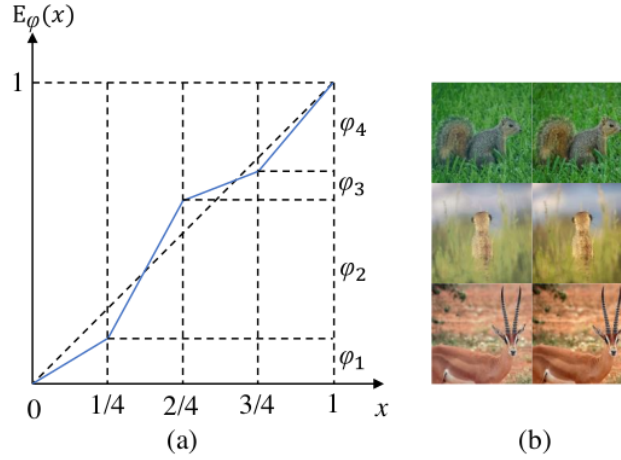


Figure 4-2 Pixel adjustment function map

In order to apply the color domain mapping algorithm in this paper to counter attacks, the functions and steps of the color mapping algorithm are further explained. The algorithm is a mapping algorithm that enables the transformation of pixels in the color space of an image, and establishes a model of the hidden space so that the adversarial samples can be found in this hidden space. Meanwhile, its constituent functions are a series of microscopic hyperparameters, which can be used in conjunction with gradient descent to update the values of the hyperparameters, and then invert the final hyperparameters to find out the final antagonistic samples obtained.

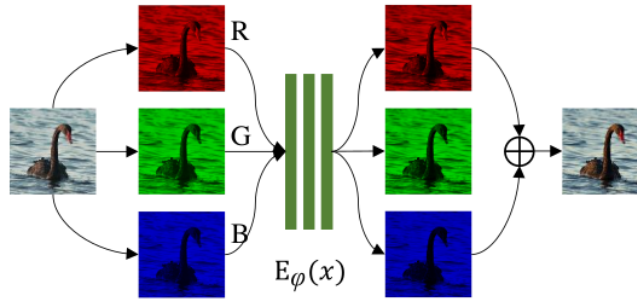


Figure 4-3 Frame diagram of color space semantic attack

As in Figure 4-3, it is illustrated that the algorithm attacks each of the three channels of the RGB space separately to obtain the final target confrontation sample. The steps to obtain the mapping function are as follows: (1) Combine the characteristics of the image, determine the number of intervals to be divided, obtain the expression of the pixel adjustment function, and utilize the expression to realize a single mapping over the

spatial domain. (2) Using the method of gradient descent, the hyperparameters of the constitutive function are updated to obtain the color space hidden space, and the antagonistic samples are searched in the hidden space.

Therefore, the pixel adjustment function with segmentation of 2 can be easily obtained with the equation shown in 4-1:

$$E_{\varphi}(x_1) = k_1 \cdot x_1, E_{\varphi}(x_2) = k_1 \cdot x_1 + k_2 \cdot (x_2 - x_1) \quad (4-1)$$

Recursively, we can easily obtain the relevant expression for $E_{\varphi}(x)$ with segment number L as follows:

$$\begin{aligned} E_{\varphi}(x_k) &= k_1 \cdot x_1 + k_2 \cdot (x_2 - x_1) + \dots + k_k \cdot (x_k - x_{k-1}) \\ &= \sum_{i=1}^{k-1} \varphi_{i-1} + L \cdot \varphi_k \cdot \left(x_k - \frac{k-1}{L}\right) \\ s. t. \quad &0 \leq \varphi_i \leq 1 \text{ and } \sum_i \varphi_i = 1 \end{aligned} \quad (4-2)$$

Where $k_1, k_2, \dots, k_n$ are the slopes of each segment of the function expression in turn,

and $\varphi_1, \varphi_2, \dots, \varphi_n$ denote the pixel-adjusted longitudinal intercept in turn. The pixel adjustment function is proposed mainly to go to find a color space that can be used for optimization and to obtain the color space for generating the antagonistic samples by solving the L microscopic hyperparameters through a certain optimization algorithm. Notice that the more the number of segments, the finer the color single mapping is, and the pixel value of the image remains unchanged when $\varphi_i = 1/L$.

In common classification tasks, the cross-entropy loss function is often used to take the output of the last layer of the neural network model as the optimization objective, and it is clear that using this optimization direction directly is not the most semantically friendly. Optimization at the feature space level uses the intermediate layer features as the optimization objective, increasing the feature distance between the adversarial sample and the clean sample so that the feature similarity between them decreases. Therefore, a loss function aiming at feature similarity change can make the two semantically similar, then the feature space editing scheme in this paper can be converted into the following optimization problem:

$$z'_{i,j} = -\min \text{sim}(f(x'_i), f(x_j)) = -\min \frac{f(x'_i)^T f(x_j)}{\|f(x'_i)\|_2 \|f(x_j)\|_2} \quad (4-3)$$

Where $\text{sim}(\cdot, \cdot)$ denotes the cosine similarity loss function, x'_i denotes the image adjusted by the pixel adjustment function, and x_j denotes the clean image. The cosine similarity is used to determine the spatial similarity of the vectors and takes the value of $[-1, 1]$, with larger values indicating greater similarity. Commonly used in contrast learning, the purpose of contrast learning is to want to train the machine to be able to

recognize similar and or dissimilar images, when performing training to compare the same batch of data, in the feature space, reduce the distance between the same categories and increase the distance between different categories. So that the distance between anchors and positive samples is as small as possible and the distance to negative samples is as large as possible.

During the optimization process, the features of the samples to be optimized are edited so that the feature distance between the samples to be optimized and the clean samples increases. The core point of this section is also to measure the distance between the two using the cosine similarity loss function, but considering that the data input to the neural network model are batches of images with positive and negative samples. The image shown in Figure 4-4 demonstrates the algorithm looking for antagonistic samples in the direction of negative samples in the embedding space. Based on this this paper transforms this optimization problem:

$$L_{tri}(x'_i, x_i, x_j) = -\min \alpha \cdot \text{sim}(f(x'_i), f(x_i)) + \beta \cdot \text{sim}(f(x'_i), f(x_j)) \quad (4-4)$$

In order to further ensure the quality of the generated image, the root mean square error loss function, i.e., L2 loss function is introduced to optimize together, which acts together to find the corresponding adversarial samples. In this paper, the adversarial sample is defined as:

$$x_i^{adv} = \arg \min_{x'_i} \{L_{tri}(x'_i, x_i, x_j) + \gamma \cdot \|E_\phi(x_i) - x_i\|_2\} \quad (4-5)$$

where x_i^{adv} represents the adversarial sample generated from the i the image of this batch of images; x'_i denotes the adversarial sample variable obtained by optimizing the hyperparameters with an initial value of x_i ; x'_i is denoted as the function $\max(0, \cdot)$; t represents the image that is least similar to x_i ; α , β , and γ are the weight parameters, and the cropping function $\text{Clip}_\epsilon\{\}$ is used to make sure that the sample produced within the limit of infinite paradigm. The pseudo-code of the related algorithm is shown in Table 4-1.

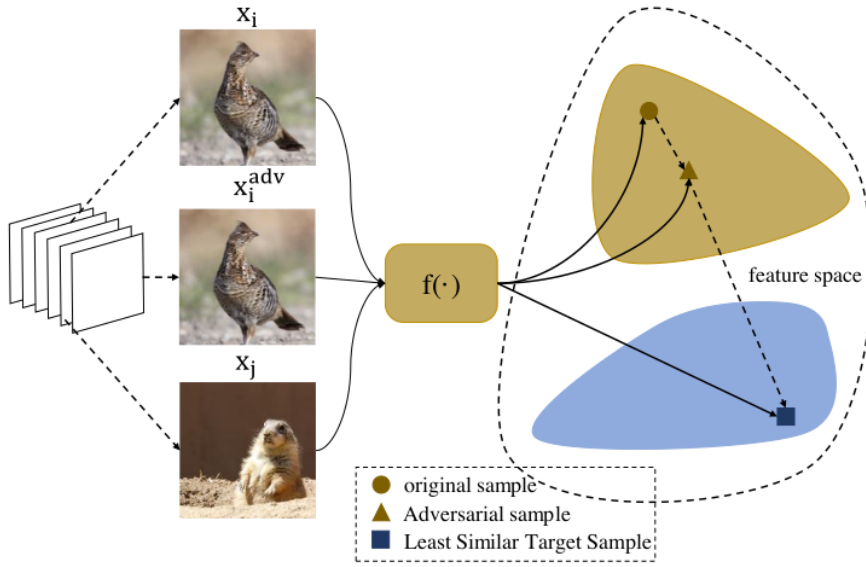


Figure 4-4 Schematic diagram of feature space

Chart 4-1 Pseudo-code for algorithms based on color space transformations

algorithms 4-1: algorithms based on color space transformations

input: The size of a batch of clean data is N , denoted as $[x_1, x_2, \dots, x_N]$;
Cosine similarity loss function $z(\cdot, \cdot)$, loss function $L_{tri}(\cdot, \cdot)$,
iterations T ;
Infinite norm limit ε , learning rate lr ;
pixmap function $E_\varphi(\cdot)$;

output: Adversarial examples $x_{adv}\{E_{\varphi_{t+1}}(x_i)\}_{i=1}^N$;

```

1: for  $t=0$  to  $T-1$  do
2:   Obtain the gradient according to Eq. 4-4;
3:   Optimizing hyperparameters  $\varphi$ , According to Eq. 3-11,  $x_t$ ;
4: end;
5: return  $\text{Clip}_\varepsilon\{x_{adv}\{E_{\varphi_{t+1}}(x_i)\}_{i=1}^N\}$ ;

```

4.2 Adversarial Sample Defense Algorithm Based on Feature Denoising

4.2.1 Requirements analysis

The current neural network defense methods, which are based on adversary training, typically involve using specific attack algorithms to target neural networks. These

methods generate adversary samples, reintroduce these samples into the network, and retrain the network with the correct labels. While this approach is effective in defending against known attack methods, it struggles to provide robust defense against newly emerging attack techniques. Therefore, there is a need to develop a defense method that can enhance the resilience of neural networks. Recent research has revealed a strong connection between the activation caused by adversary perturbations and network parameters. This often results in sparse activation, which significantly differs from the activation patterns seen in real data. Leveraging traditional filtering and denoising techniques can effectively suppress this sparse activation, thus improving the robustness of neural networks. Without relying on specific attack algorithms for adversary training. The key steps of our proposed algorithm are illustrated in the figure below:

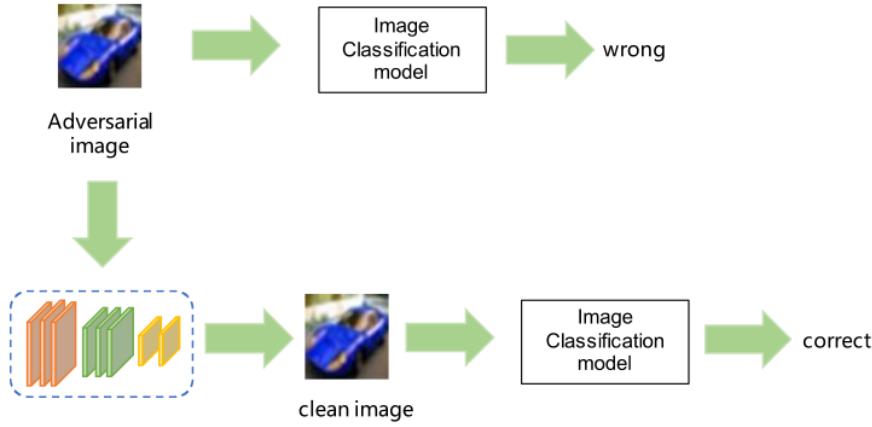


Figure 4-5 Traditional methods can only remove perturbation at pixel level, besides the perturbation between the adversarial image and the original image is amplified during the high-level net. We use CRGAN to remove the perturbation, and to construct the clean image.

4.2.2 Structure of the CRGAN

CRGAN is a defense strategy inspired by the CGANS. The basic idea of CRGAN is illustrated in Figure 4-6. The main structure of CRGAN includes generator and discriminator. The data sent to the generator is the image denoised by the random mask network. The purpose of the G is to learn the relationships between real data and mask data and to further remove noise to generate as accurate as possible normal data. D attempts to determine whether the conditional fake image $G(f(x))$ reconstituted into normal samples. $G(f(x))$ can be correctly identified by the trained model. We define the objective function of CRGAN as

$$\min_{\theta_g} \max_{\theta_d} (E_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x}|f(\mathbf{x}))} [\log D(\mathbf{x}, f(\mathbf{x}))] + E_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x}|f(\mathbf{x}))} [\log(1 - D(G(f(\mathbf{x})), f(\mathbf{x})))]$$

Where G_i represents the parameters of generator's structure and G_d represents the parameters of discriminator's structure.

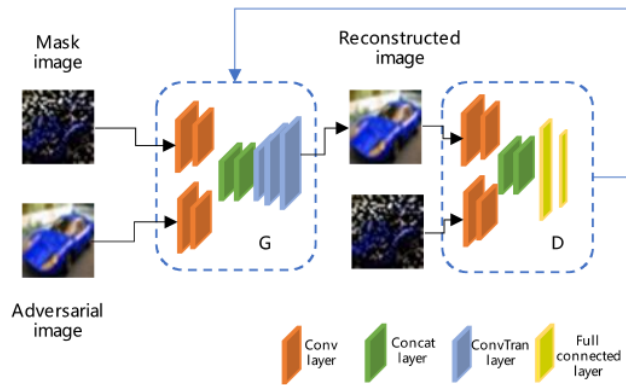


Figure 4-6 The overview of CRGAN. Both G and D utilize concat layer to capture spatial information

As shown in Figure 4-7, we display the details of the generator G and the discriminator D. The generator G consists of 6 weight layers including 3 convolutional layers and 3 deconvolutional layers as encoder and decoder, respectively, which is used to effectively transform the perturbed image into original image. Our method use a concat layer to merge the mask data and the real data as the input in encoder for the purpose of capture the spatial structure of the data. The discriminator D consisting of 4 weight layers utilizing convolutional layers to extract features for classification. We also use the merge data as the input to indicate whether the input data is a correct interpolation.

The GAN is trained by adversarial training, which using the generated adversarial samples and inputting them into the neural network again, the adversarial samples are extracted through the intermediate layer features of the neural network. The model mainly makes use of the remap method to reconstruct the output image. It is obvious that the GAN can be used as a defense to any attack, and can leverage the generative power of GANs to reconstruct adversarial examples.

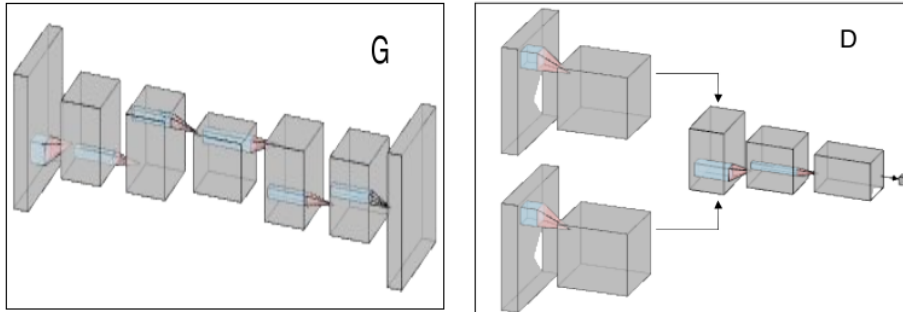


Figure 4-7 Detailed information in G and D.

4.2.3 Perturbed dissolved layer

As regarded above, adversarial perturbations are more destructive in deep learning. Considering that the vulnerability to adversarial examples, this paper uses a random mask network to destroy the adversarial perturbation, while constructing an interpolation scene as a condition. Specifically, we use the masked images as the only constraint for the generator rather than random data samples. We let the data space be P , where W and H represents the width and height of the data and C is the number of its channels. A real spatial input is defined as $x \in V$. Formally, We can define a random sampling function $F()$ to process the input x .

$$f = [(a_1, b_1), (a_2, b_2), \dots, (a_n, b_n)] \in \mathbb{R}^{2n}$$

Where (a_k, b_k) denotes the coordinate of the k th point, we can formalize the sampled spatial image $f(x) \in V$ as:

$$f(x)^{(i,j)} = \begin{cases} x(\cdot, i, j) & \text{if } (i, j) \in f, \\ N & \text{otherwise} \end{cases}$$

Compared with CGANs, the above method contains a spatial conditional data $f(x)$ from the training data instead of some additional label information y . In addition, the input to the proposed model is an image rather than the prior noise vector z . In this way, the model can defend the adversarial examples, and can also build a conditional interpolation scene. Then, we feed the mask data to the generator to transform the adversarial examples into clean examples.

4.2.4 Loss function

The loss function provides us with the critical part in image reconstruction mission. The pixel-level mean squared error (MSE) is one of the most widely used loss functions, but with the deepening of the network, some of the noise that has not been removed will have a large difference in the features of the middle layer due to the error amplification effect. The lack of semantic information cannot guarantee the quality of image restoration. We hope that the denoised image is similar to the original image in terms of visual effect, so we choose to use both L_1 and L_{feat} to measure gap between features.

The feature reconstruction loss is the(squared, normalized) Euclidean distance between feature representations:

$$\ell_{feat}^{\phi, j}(\hat{y}, y) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y)\|_2^2$$

Where $\phi_j(x)$ be the activations of the j th layer of the network ϕ when processing the image x ; $C_j \times H_j \times W_j$ represents the size of the j th layer convolutional map $\phi_j(x)$.

To further constrain adversarial examples in the latent space, we use triplet loss to provide more effective and stronger constraints in adversarial training. Given an original clean image x_s , we first utilize the targeted attack method to generate the targeted adversarial example $x_s^{y_c}$ towards its most confusing class. Then, Let $x_s^{y_c}$ be

a sample from the most confusing class, and generate its adversarial example $x_s^{y_c}$ that the label is the same as the original label s . Let x_s , $x_{y_c}^s$, $x_s^{y_c}$ be positive, anchor and negative sample, respectively. The triplet loss is defined as:

$$\begin{aligned} \mathcal{L}_{tr}(x_a, x_p, x_n) \\ = \left[d(E(x_a), E(x_p)) - d(E(x_a), E(x_n)) + m \right]_+ \end{aligned}$$

where x_a , x_p , x_n denote to positive, anchor and negative samples, respectively; We can define the distance as:

$$1 - \frac{|a \cdot b|}{\|a\|_2 \|b\|_2}$$

By shorting the distance between the original image and the most easily confused adversarial example, it can effectively constrain the representation of different categories in the feature space, while expanding the distance between the forward adversarial examples and the backward adversarial examples. The total loss function of the generative network is as follows:

$$L = L_1 + \alpha L_{\text{feat}} + \beta L_{tr}$$

where α , β indicate the balance factor.

5. Design/ Data collection (survey or review)

Our research aims to explore the image classification performance on CIFAR-10 and GTSRB datasets. Our main research question is to determine the classification performance of different models on different datasets. We adopted a quantitative research methodology in order to evaluate the model performance through accurate numerical results.

We downloaded the CIFAR-10 dataset containing 60,000 color images from publicly available data repositories and normalized the images. For GTSRB, we downloaded the traffic sign images from the official website and pre-processed them, including resizing and label encoding.

The CIFAR-10 dataset was created by the Canadian Computer Vision Research Group for image classification research and released in 2010. GTSRB was created by the German Ministry of Transportation for traffic sign recognition research and released in 2011. We divided the CIFAR-10 dataset into training, validation, and test sets to ensure randomization and reproducibility of the data. In GTSRB, we removed some poor quality images and resized the images to the same size to ensure input consistency. We plan to use metrics such as accuracy, ssim, etc. to evaluate the algorithm performance.

6. Data Analysis (survey or review)

This experiment is verified experimentally on CIFAR10 and GTSRB . The CIFAR10 dataset is a small color image dataset that is often used for classification tasks in the field of computer vision. It contains 10 different categories, such as airplane, car, and truck, etc., and each category has 6000 images, of which 5000 images are used as training set and 1000 images are used as test set. As shown in Figure 6-1, some images of the CIFAR10 dataset are shown.

The GTSRB traffic dataset is a dataset for traffic sign recognition. The original dataset consists of 43 different categories, which are saved in 43 folders. The data set contains 39,209 training images and 12,630 test images. The size of each sample ranges from 15×15 to 250×250 . During the training process, the image size needs to be adjusted to the same size. Subsequent experiments The size of the image is adjusted to 32×32 by means of bilinear interpolation. The image samples in the GTSRB traffic data set are affected by the external environment such as illumination and occlusion during the collection process. Data images are rich in features, with varying lighting conditions and rich and varied backgrounds. Figure 6-2 shows some images from the GTSRB traffic dataset.

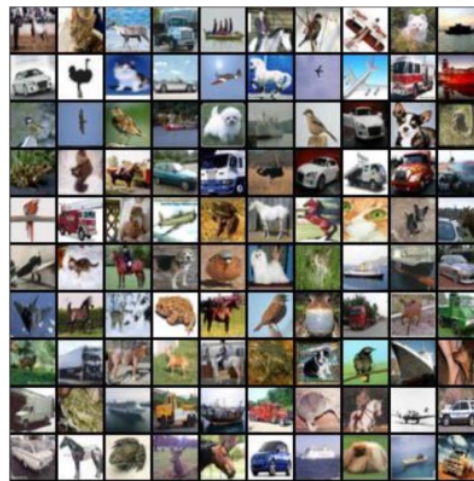


Figure 6-1 CIFAR10 dataset sample map

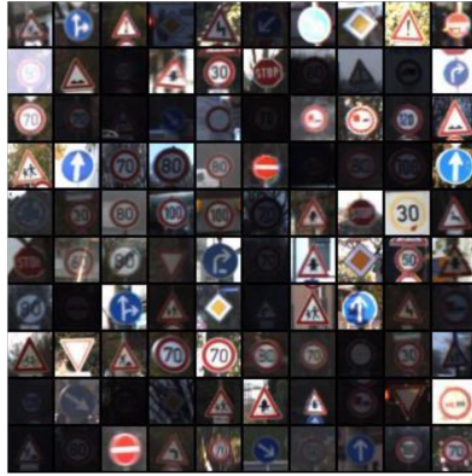


Figure 6-2 GTSRB dataset sample map

We used some preprocessing schemes for both datasets, for CIFAR-10, we normalized the images and scaled the pixel values to between 0 and 1. For GTSRB, we resized the images and coded the classes of traffic signs using solo thermal coding.

7. Results and Discussion

In the attack algorithm, we use the following metrics: maximum perturbation, attack success rate, and structural similarity. In this paper, we use the adversarial attack algorithms proposed in this chapter on four common neural network classification models, ResNet18, VGG16, AlexNet, and Inception-v3, to conduct experiments for an infinite number of paradigm distances with different number of algorithmic iterations, T , and different perturbation ranges, ϵ , settings, to generate the corresponding adversarial samples, and then use different image classification models for classification. The success rate of the attack on the image classification model and the image distortion of the confrontation samples are recorded for different T and epsilon settings to compare the efficiency of each algorithm. In this experiment, the segmentation of the pixel adjustment function is adjusted to $L=64$, the perturbation ratio is limited to 0.5, and the hyperparameters of the loss function, α , are set to 1.2 and β is set to 0.8. The experiments in this chapter validate the effect of the attack algorithm based on the color space, and prove the effectiveness of the algorithm.

Tables 7-1, 7-2, and 7-3 show the attack success rate, image structure similarity, and L2 distance of the adversarial samples generated by the traditional attack method and this paper's method on the four network models with the same infinite-paradigm distance constraints ϵ (0.03, 0.06), and the same number of iterations for the CIFAR10 and GTSRB datasets, respectively.

7-1 Experimental results of attack success rate

Datasets	Model	PGD	I-FGSM	MI-FGSM	Ours
CIFAR10	ResNet18	0.918/0.996	0.877/0.897	0.965/0.994	0.992/0.996
	VGG16	0.855/0.996	0.869/0.887	0.899/0.992	0.920/0.999
	AlexNet	0.912/0.987	0.923/0.958	0.966/0.999	0.955/ 0.999
	Inceptionv3	0.895/0.956	0.884/0.936	0.908/0.957	0.908/ 0.959
GTSRB	ResNet18	0.539/0.853	0.755/0.883	0.925/0.985	0.896/0.975
	VGG16	0.593/0.875	0.750/0.836	0.901/0.974	0.887/0.971
	AlexNet	0.620/0.872	0.857/0.928	0.913/0.939	0.896/0.933
	Inceptionv3	0.429/0.720	0.572/0.755	0.737/0.891	0.737/0.822

From Tables 7-1, it can be seen that for different datasets and models, the methods proposed in this paper can all achieve better attack results. Under the same perturbation intensity, the achieved attack performance is better than that of PGD and I-FGSM which are also based on gradient optimization. The method in this paper is improved from MI-FGSM, and due to the sacrifice of the perturbation range, it will lead to a less successful attack rate than that of MI-FGSM on some models. The attack performance of the various attack algorithms on the CIFAR10 dataset is significantly better than that on the GTSRB dataset, and the reason is that although the GTSRB dataset is not the most popular dataset, it is not the most popular dataset in the world. It is because the GTSRB dataset is a color image but has simpler data categories, and the neural network models are sufficient to fit distinct decision boundaries by learning the low-frequency structural information, resulting in a less effective attack than the CIFAR10 dataset.

7-2 L2 distance experiment results

Dataset	Model	PGD	I-FGSM	MI-FGSM	Ours
CIFAR10	ResNet18	1.008/3.855	1.633/5.548	2.718/10.77	0.766/3.129
	VGG16	1.006/3.856	1.616/5.594	2.718/10.77	0.753/3.092
	AlexNet	1.051/4.085	2.048/7.394	2.719/10.77	1.064/3.847
	Inceptionv3	0.999/3.751	1.757/6.118	2.718/10.76	0.983/3.361
GTSRB	ResNet18	1.021/3.947	1.739/6.137	2.701/10.67	0.930/3.327
	VGG16	1.008/3.891	1.627/5.838	2.701/10.67	0.864/3.084
	AlexNet	1.029/3.980	1.871/6.506	2.702/10.67	0.981/3.546
	Inceptionv3	0.991/3.809	1.575/5.486	2.701/10.67	0.904/3.124

Table 7-2 demonstrates the L2 distances of the four algorithms. The smaller the L2 distance, the less distorted the image. From the table, it can be seen that the MI-FGSM algorithm has the best attack success rate, but at the same time, the L2 distance is also the largest. The L2 distance of MI-FGSM is three times of the proposed method in this paper. The MI-LPFGSM algorithm is able to achieve the effect of the similar attack success rate to obtain the smallest L2 distance. Its attack success rate and L2 distance are better than PGD and I-FGSM algorithms under the same perturbation intensity. The

L2 distance is only half of that of I-FGSM with ten percentage points higher attack success rate than I-FGSM.

In addition to visually comparing the image distortions of different methods using L2 paradigm distance, Structural Similarity of Images (SSIM) is introduced in this experiment to evaluate the image loss of generating adversarial samples. As can be seen from Table 7-3, the algorithm MI-LPFGSM proposed in this paper performs better than others on both datasets with higher image structural similarity. When $\epsilon=0.06$, it is able to improve the SSIM by about 0.16 while ensuring the same attack rate.

7-3 Experimental results on image structure similarity

Dataset	Model	PGD	I-FGSM	MI-FGSM	Ours
CIFAR10	ResNet18	0.980/0.905	0.977/0.799	0.962/0.776	0.987/0.933
	VGG16	0.980/0.904	0.977/0.799	0.962/0.774	0.987/0.933
	AlexNet	0.980/0.900	0.977/0.799	0.962/0.776	0.987/0.933
	Inceptionv3	0.980/0.903	0.928/0.800	0.963/0.776	0.970/ 0.933
GTSRB	ResNet18	0.940/0.800	0.878/0.758	0.855/0.654	0.957/0.853
	VGG16	0.941/0.801	0.878/0.757	0.905/0.675	0.957/0.850
	AlexNet	0.940/0.800	0.878/0.758	0.905/0.676	0.957/0.852
	Inceptionv3	0.941/0.803	0.878/0.755	0.905/0.671	0.957/0.852

Figures 7-1 and 7-2 show the experimental results in the CIFAR10 and GTSRB datasets, respectively. In these two datasets, the same experimental parameters including the same models and algorithms are used, and different algorithms have similar performance in the two datasets with the same experimental parameters. With the same number of iterations, the classification accuracy, SSIM decreases sequentially as the attack intensity increases. Meanwhile, the L2 distance keeps increasing. With the same attack intensity, the classification accuracy decreases sequentially as the number of iterations increases. The attack performance of our proposed algorithm is better than that of FGSM and I-FGSM. When the L_∞ distance is small, the attack performance of MI-FGSM algorithm is slightly higher than that of this paper's algorithm, and with the increasing L_∞ distance, the attack performance of each algorithm rises rapidly, and finally tends to the constant value. On the CIFAR10 dataset, the attacks of all algorithms except the I-FGSM algorithm reach close to 100 percent success rate when the number of iterations is 20. Similarly, on the GTSRB dataset, the effect is achieved at a number of iterations of 40. As the number of iterations increases, the attack performance of this paper's algorithm tends to be consistent with MI-FGSM. The L2 distance reacts to the similarity between two images, and it can be seen from the L2 distance index that the L2 distance of each method increases sequentially with the increase of the L_∞ paradigm distance. The MI-FGSM increases rapidly, while the algorithm proposed in this paper increases slowly. When L_∞ is greater than or equal to 0.06, all the algorithms except the I-FGSM algorithm can achieve close to 100% success rate, and at this time, the SSIM of this paper's method is 0.2 higher than that of MI-FGSM, and

0.13 higher than that of I-FGSM, and the corresponding average L2 distance is only about 1/2 to 1/5 of that of MI-FGSM, and about 1/2 of that of I-FGSM. As the L_∞ paradigm distance continues to increase, the L2 distance for each method will increase in turn. L_∞ paradigm distance continues to increase, when L_∞ is equal to 0.1, the performance of the average L2 distances for the antagonistic and clean samples at the corresponding number of iterations is 2-4 times that of the method in this paper for the I-FGSM and about 6 times that of the method in this paper for the MI-FGSM.

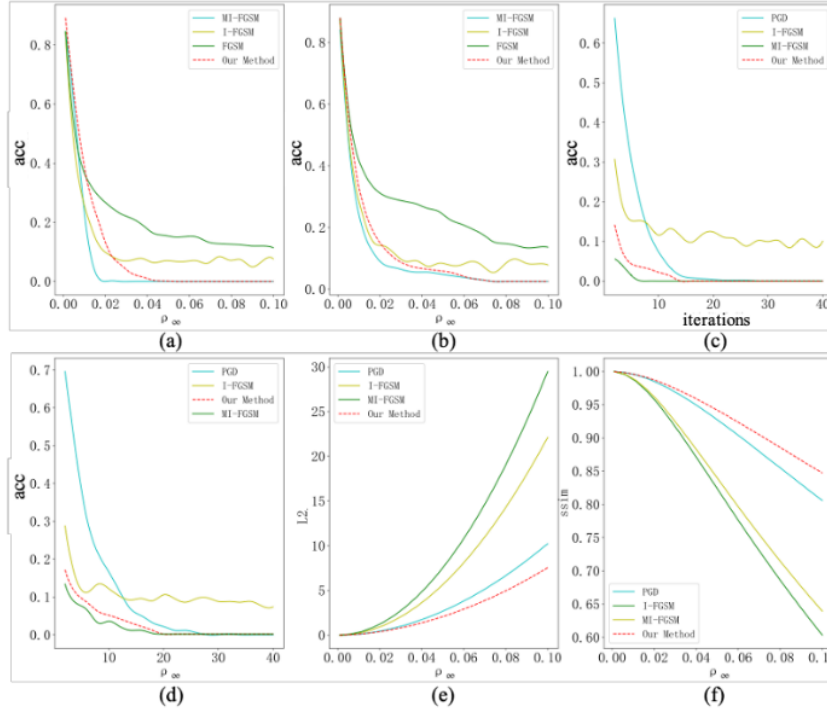


Figure 7-1 Plot of experimental results on the CIFAR10 dataset. (a) L_∞ attack strength vs. classification accuracy on ResNet model (b) L_∞ attack strength vs. classification accuracy on VGG16 model (c) Number of iterations vs. classification accuracy on ResNet model (d) Number of iterations vs. classification accuracy on VGG model (e) L_∞ attack strength vs. image similarity on ResNet model (f) L_∞ attack strength vs. image similarity on VGG model. infity attack strength versus image similarity(f) L_∞ attack strength versus image similarity on the VGG model.

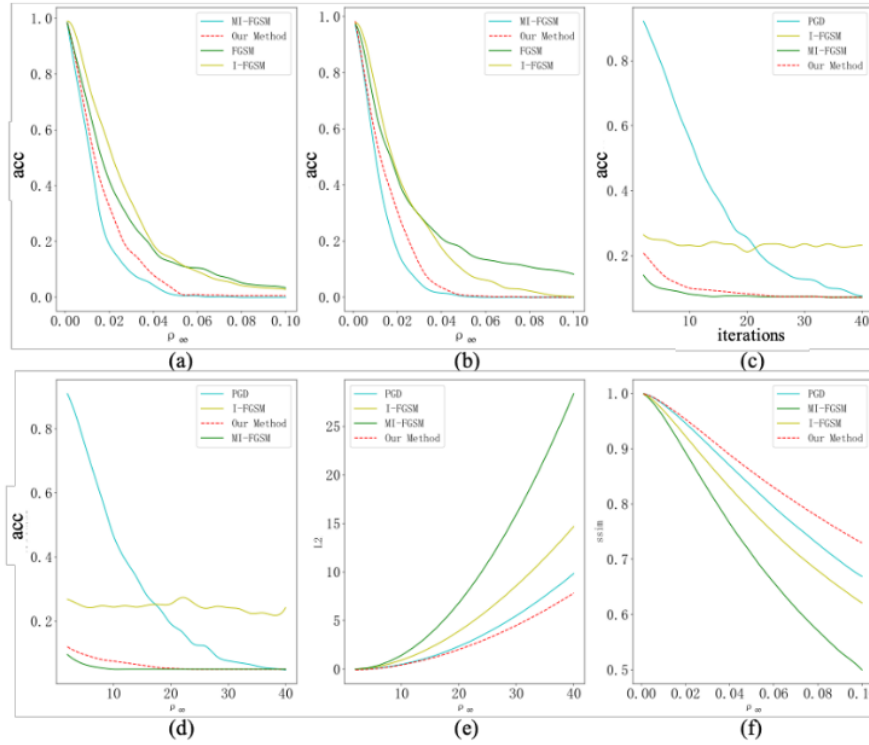


Figure 7-2 Plot of experimental results on the GTSRB dataset. (a) L_∞ attack strength vs. classification accuracy on ResNet model (b) L_∞ attack strength vs. classification accuracy on VGG16 model (c) Iteration number vs. classification accuracy on ResNet model (d) Iteration number vs. classification accuracy on VGG model (e) L_∞ attack strength vs. image similarity on ResNet model (f) L_∞ attack strength vs. image similarity on VGG model (g) L_∞ attack strength vs. image similarity on ResNet model (h) L_∞ attack strength vs. infty attack strength versus image similarity (f) L_∞ attack strength versus image similarity on the VGG model.

Similarly, there is a similar gap in the adversarial and clean sample SSIM metrics, with MI-LPFGSM taking the best results at all attack intensities. For the same number of iterations, although MI-FGSM has the highest attack success rate, it generates more distorted adversarial images. And as the number of iterations increases, its attack effect with MI-LPFGSM tends to be similar, so as long as the number of iterations is high enough, the MI-LPFGSM algorithm achieves better image similarity while achieving a higher success rate.

Figure 7-3 demonstrates the attack success rate of the white-box attack algorithm

proposed in this paper on the defense model (a neural network classification model obtained by adversarial training of the PGD attack algorithm). In the figure, it can be seen that in the case of the same number of iterations, the attack success rate is highest when the segmentation $L=64$, and can be higher than the other segmentation results in 10% attack success rate. Moreover, when the segmentation is fixed, with the increase of iteration number, the attack success rate under each segmentation increases sequentially and tends to stabilize, in which the attack success rate is the highest when the segmentation $L=64$.

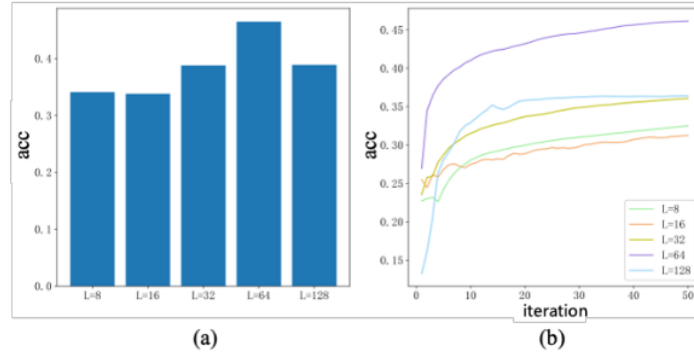


Figure 7-3 Effects corresponding to different segments of the pixel adjustment function. (a) Attack success rate for different segments (b) Number of iterations vs. segments.

7-4 Attack performance of different algorithms on defense systems

Defense Model	PGD	FGSM	Deepfool	Ours
Normal	0.928	0.819	0.908	0.953
AT	0.225	0.231	0.246	0.484
ComDefend	0.755	0.632	0.109	0.802
Feature Squeezing	0.966	0.734	0.199	0.990
DipDefend	0.447	0.506	0.098	0.568

Table 7-4 shows the results of the attack accuracy comparison of the related algorithms on the CIFAR10 dataset against each defense model. As can be seen from the table, on the non-defense model, the classification accuracies between the algorithms are similar. However, on the defense model, AT is trained by the PGD algorithm, which robustly fits the adversarial distribution space generated by the gradient attack algorithm, and has similar defense effects against several attack algorithm models. For the two compression defense strategies ComDefend, Feature Squeezing lose more image information during image compression, so they also do not produce a better defense effect. DipDefend can produce the best defense effect. Compared with PGD and FGSM

algorithms, this paper's algorithm has close attack strength on the original model, which is ten percentage points higher than other attack methods, and obtains the best attack effect on the defense model.

Figure 7-4 shows the attack effect of MI-FGSM with our proposed method on some image datasets. In each set of similar images, it can be seen that after the color space transformation, the adversarial perturbation increases, but the image similarity is approximate, indicating that the proposed algorithm can explore a broader adversarial space and obtain better adversarial samples.

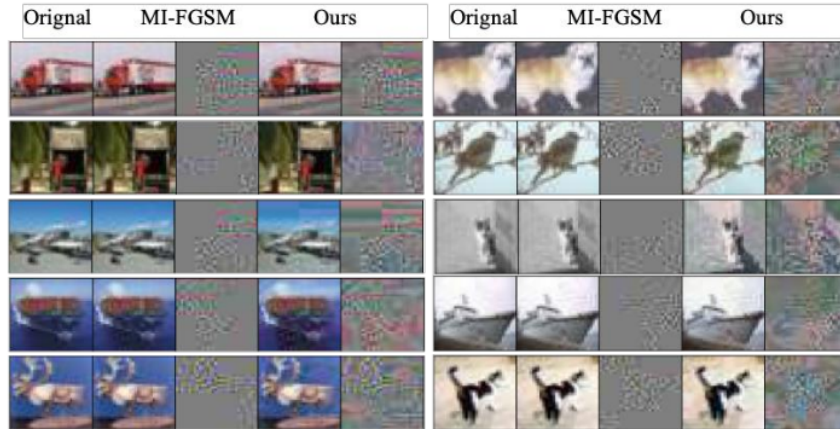


Figure 7-4 Anti-effect diagram. The first column is the original image, the second and fourth columns are the adversarial samples generated by MI-FGSM and ours attacking the ResNet model respectively, and the third and fifth columns are the noise diagrams of the adversarial samples generated by it

In defense algorithms, we used defense success rate to evaluate the performance of defense algorithms. We evaluated the performance of defense algorithms in an generic model Resnet50. In this paper, we first proposed a CRGAN model, in which uses GAN to eliminate adversarial perturbation. To verify the effective of CRGAN model, we compared CRGAN with some commonly used models, such as Adversarial Training, Feature Squeezing and ComDefend. The test images consist of clean data and adversarial data. The data are masked by the perturbed dissolved layer, and then they are transformed by CRGAN, finally they are fed to the well-trained classifier. Specifically, we use Batch normalization (BN) for all layers and LeakyReLU as a non-linearity. Moreover, we use convolution operation for downsampling and upsampling, respectively. we train our method for 100 epochs using a batch size of 512. For the feature extraction network, we choose the output feature map of the 4th convolutional layer before the 5th max-pooling layer in the VGG-16 classification network to calculate the perceptual loss.

7-5 Experimental results on cifar10

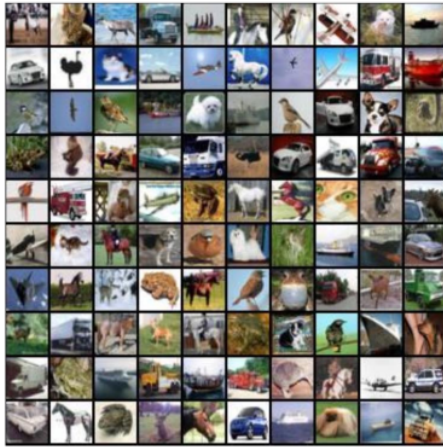
Defense Model	Clean	FGSM	PGD	Deepfool	C&W
Normal	0.93	0.18/0.13	0.00/0.00	0.09/0.09	0.00/0.00
AT	0.76	0.26/0.16	0.03/0.03	0.80/0.79	0.77/0.68
ComDefend	0.88	0.59/0.48	0.80/0.79	0.80/0.80	0.61/0.37
Feature Squeezing	0.83	0.26/0.16	0.03/0.01	0.80/0.79	0.45/0.33
Ours	0.93	0.88/0.70	0.93/0.91	0.93/0.93	0.93/0.93

7-6 Experimental results on GTSRB

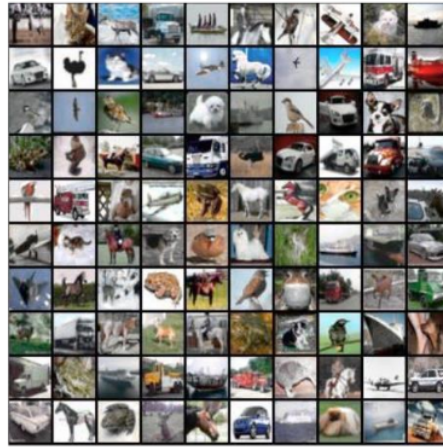
Defense Model	Clean	FGSM	PGD	Deepfool	C&W
Normal	0.98	0.28/0.10	0.69/0.36	0.11/0.00	0.98/0.94
AT	0.95	0.88/0.71	0.90/0.89	0.95/0.93	0.95/0.95
ComDefend	0.95	0.88/0.86	0.89/0.88	0.95/0.95	0.96/0.96
Feature Squeezing	0.51	0.42/0.32	0.50/0.44	0.47/0.39	0.51/0.51
Ours	0.98	0.91/0.69	0.97/0.96	0.98/0.98	0.98/0.98

Table 7-5、7-6 shows the results of our experiments. Comparing with other deep learning based models, we can find that the proposed method has an excellent performance. The CRGAN can achieve nearly 80% accuracy under various attack types and strengths. In addition, the performance of is CRGAN is better than other preprocessor based defense methods such as Feature Squeezing ComDefend. By contrast, our model can better reconstruct the important features of the image and has better robustness. The experiment shows that when the clean sample is passed through the reconstruct model CRGAN, the classification accuracy did not decrease too much, cifar-10 remained at 89.9%, and GTSRB remained at 98.0%.

Figures 7-5 and 7-6 show the experimental effect graphs after the defense algorithm, it can be seen that after our defense algorithm the original image will have part of the saturation degradation, and the denoising algorithm removes part of the original information.



Original



Ours

Figure 7-5 The left image is the original image and the right image is the image after the defense algorithm.



Original



Ours

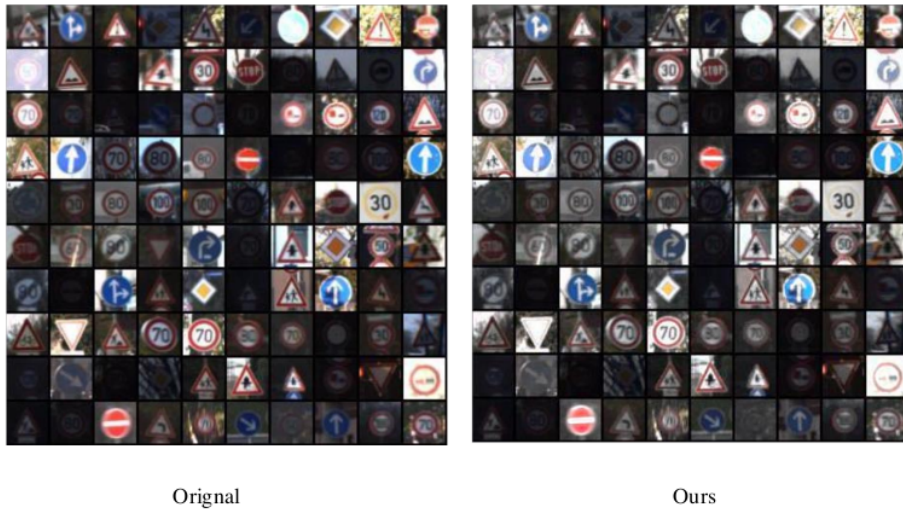


Figure 7-6 The left image is the original image and the right image is the image after the defense algorithm.

8. Conclusions and Recommendations

In this paper, a color domain mapping algorithm is proposed to represent the individual values in the pixel domain in segments and use feature space similarity editing to map the original pixel domain into other pixel domains with a non-paradigmatic distance strategy, and finally experiments are conducted on two commonly used datasets, CIFAR10 and GTSRB, to validate the effectiveness of the attack algorithm. In addition, this paper proposed and evaluated a new approach on basis of GAN model, which is based on the idea of image preprocessing, and converts adversarial sample defense into an image denoising task. Especially CRGAN model can be effectively applied to defend the adversarial examples. Our model builds the mapping space by using GAN model. It was noticed that our model has greater result than other models such as Adversarial Training, Feature squeezing. CRGAN is only trained for adversarial examples of one attack, but has the ability to defend against adversarial examples of other attack algorithms, which indicates that the model has a certain generalization ability. The advantage of this method lies in its remarkable defense effect and great transferability.

9. Reflection

The algorithms proposed in this paper aim to address the problem of defending against adversarial sample attacks in two different scenarios based on deep learning security. However, there are still many research directions in the field of neural network security that are worth exploring in depth. From the experimental results, certain results have been achieved, but future research can be continued in the following areas:

In this paper, an encoder is trained to map the original sample space to the hidden space of the image, which is combined with a GAN to continuously update the network during

the training process, which takes a long time and is complicated to train, so other compression techniques that do not require large-scale training while being able to map the image to the corresponding hidden space will be explored in the future.

More practical black-box counter attack methods. In practical application scenarios, AI service providers do not disclose the parameters and datasets for model training, it is difficult to obtain pre-trained models and the same datasets with similar structural variability as the target model, how to obtain an effective alternative model through pre-training is an important test, to address these issues, we can start from the high-dimensional spatial boundaries of the data and the intrinsic mechanism of the model, and seek to approximate the practical cross-domain methods for attack problems in scenarios.

References

- [1] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

- [2] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
- [3] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).
- [4] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).
- [5] Zhang, Q., Zhao, S., Luo, Y., Zhang, D., Huang, N., & Han, J. (2021). ABMDRNet: Adaptive-weighted bi-directional modality difference reduction network for RGB-T semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 2633-2642).
- [6] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).
- [7] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
- [8] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
- [9] Lin, Z., Peng, C., Tan, W., & He, X. (2023). Image Adversarial Example Generation Method Based on Adaptive Parameter Adjustable Differential Evolution. *Entropy*, 25(3), 487.
- [10] Zhang, Y., Liu, Y., Liu, J., Miao, J., Argyriou, A., Wang, L., & Xu, Z. (2022). 360-attack: Distortion-aware perturbations from perspective-views. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 15035-15044).
- [11] Hu, Z., Huang, S., Zhu, X., Sun, F., Zhang, B., & Hu, X. (2022). Adversarial texture for fooling person detectors in the physical world. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 13307-13316).
- [12] Li, L., Ma, R., Guo, Q., Xue, X., & Qiu, X. (2020). Bert-attack: Adversarial attack against bert using bert. arXiv preprint arXiv:2004.09984.
- [13] Belinkov, Y., & Glass, J. (2019). Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7, 49-72.
- [14] Seung-Hoon Na & Yehe Lee.(2015).Deep learning for Natural language processing. *Communications of the Korean Institute of Information Scientists and Engineers*(10).
- [15] Ebrahimi, J., Rao, A., Lowd, D., & Dou, D. (2017). Hotflip: White-box adversarial examples for text classification. arXiv preprint arXiv:1712.06751.

- [16] Zhang, H., Zhou, H., Miao, N., & Li, L. (2020). Generating fluent adversarial examples for natural languages. arXiv preprint arXiv:2007.06174.
- [17] Alzantot, M., Sharma, Y., Elgohary, A., Ho, B. J., Srivastava, M., & Chang, K. W. (2018). Generating natural language adversarial examples. arXiv preprint arXiv:1804.07998.
- [18] Dar, S. A., Palanivel, S., & Geetha, M. K. (2022). Autonomous Taxi Driving Environment Using Reinforcement Learning Algorithms. *International Journal of Modern Education & Computer Science*, 14(3).
- [19] Smmarwar, S. K., Gupta, G. P., & Kumar, S. (2023). AI-empowered malware detection system for industrial internet of things. *Computers and Electrical Engineering*, 108, 108731.
- [20] Shaukat, K., Luo, S., & Varadharajan, V. (2022). A novel method for improving the robustness of deep learning-based malware detectors against adversarial attacks. *Engineering Applications of Artificial Intelligence*, 116, 105461.
- [21] Wang, J. (2021, August). Adversarial Examples in Physical World. In *IJCAI* (pp. 4925-4926).
- [22] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
- [23] Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., & Li, J. (2018). Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9185-9193).
- [24] Moosavi-Dezfooli, S. M., Fawzi, A., & Frossard, P. (2016). 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).
- [25] Carlini, N., & Wagner, D. (2017, May). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)* (pp. 39-57). Ieee.
- [26] Brown, T. B., Mané, D., Roy, A., Abadi, M., & Gilmer, J. (2017). Adversarial patch. arXiv preprint arXiv:1712.09665.
- [27] Liu, X., Yang, H., Liu, Z., Song, L., Chen, Y., & Li, H. D. (2019, January). An Adversarial Patch Attack on Object Detectors. In *Proceedings of the Workshop on Artificial Intelligence Safety*.
- [28] Xiao C, Zhu J-Y, Li B, et al. Spatially transformed adversarial examples[J]. arXiv preprint arXiv:1801.02612, 2018.
- [29] Laidlaw C, Feizi S. Functional adversarial attacks[J]. *Advances in neural information processing systems*, 2019, 32: 1-11.
- [30] Zhong, Y., Liu, X., Zhai, D., Jiang, J., & Ji, X. (2022). Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 15345-15354).

- [31] Wang, X., & He, K. (2021). Enhancing the transferability of adversarial attacks through variance tuning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1924-1933).
- [32] Lin, J., Song, C., He, K., Wang, L., & Hopcroft, J. E. (2019). Nesterov accelerated gradient and scale invariance for adversarial attacks. arXiv preprint arXiv:1908.06281.
- [33] Xie, C., Zhang, Z., Zhou, Y., Bai, S., Wang, J., Ren, Z., & Yuille, A. L. (2019). Improving transferability of adversarial examples with input diversity. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2730-2739).
- [34] Wu, W., Su, Y., Lyu, M. R., & King, I. (2021). Improving the transferability of adversarial samples with adversarial transformations. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 9024-9033).
- [35] Liao, F., Liang, M., Dong, Y., Pang, T., Hu, X., & Zhu, J. (2018). Defense against adversarial attacks using high-level representation guided denoiser. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1778-1787).
- [36] Jia, X., Wei, X., Cao, X., & Foroosh, H. (2019). Comdefend: An efficient image compression model to defend adversarial examples. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6084-6092).
- [37] Gu, T., Dolan-Gavitt, B., & Garg, S. (2017). Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv preprint arXiv:1708.06733.

Appendix

- Appendix1: PG2-E1-Ethics-Form-08-2019 - annotated
- Appendix2: Declaration and Form of Consent
- Appendix3: Research on image-oriented adversarial sample attack and defense algorithms PPT
- Appendix4: Project Code

Research on image-oriented adversarial sample attack and defense algorithms

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42