remote sensing

MDPI

Article

A Gradual Adversarial Training Method for Semantic Segmentation

Yinkai Zan [1,2], Pingping Lu [1,2,*] and Tingyu Meng [1]

[1] National Key Laboratory of Microwave Imaging, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China; zanyinkai20@mails.ucas.ac.cn (Y.Z.); mengty@radi.ac.cn (T.M.)
[2] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: lupp@aircas.ac.cn

Abstract: Deep neural networks (DNNs) have achieved great success in various computer vision tasks. However, they are susceptible to artificially designed adversarial perturbations, which limit their deployment in security-critical applications. In this paper, we propose a gradual adversarial training (GAT) method for remote sensing image segmentation. Our method incorporates a domain-adaptive mechanism that dynamically modulates input data, effectively reducing adversarial perturbations. GAT not only improves segmentation accuracy on clean images but also significantly enhances robustness against adversarial attacks, all without necessitating changes to the network architecture. The experimental results demonstrate that GAT consistently outperforms conventional standard adversarial training (SAT), showing increased resilience to adversarial attacks of varying intensities on both optical and Synthetic Aperture Radar (SAR) images. Compared to the SAT defense method, GAT achieves a notable defense performance improvement of 1% to 12%.

Keywords: adversarial examples; adversarial training; deep neural network

Citation: Zan, Y.; Lu, P.; Meng, T. A Gradual Adversarial Training Method for Semantic Segmentation. *Remote Sens.* **2024**, *16*, 4277. https://doi.org/10.3390/rs16224277

Academic Editors: Haipeng Wang, Gang Xu and Lan Du

Received: 9 October 2024
Revised: 13 November 2024
Accepted: 14 November 2024
Published: 16 November 2024

1. Introduction

Deep neural networks have demonstrated their performance in most computer vision tasks, such as image classification, object detection, and semantic segmentation. Among them, semantic segmentation plays an important role in urban planning, vegetation coverage detection, and resource detection, which require their robustness in such situations. However, recent studies have shown the vulnerability of DNNs to adversarial examples, which refer to images with adversarial perturbations that are imperceptible to humans [1–3]. The concept of adversarial examples was first introduced by Szegedy et al. in 2013 [4]. They demonstrated that small, imperceptible perturbations added to the input image can lead to misclassification for most state-of-the-art neural network models. Adversarial perturbations in practical applications have been achieved through a range of methods, including the partial coating of objects with radar-absorbent materials [5], the use of camouflage grass for target masking [6], and the application of electromagnetic camouflage using a metasurface skin [7].

In order to improve the reliability of DNNs, a lot of research has been conducted to design defense mechanisms against these vulnerabilities [8]. In 2019, the GARD (Guaranteeing AI Robustness against Deception) program was established by the Defense Advanced Research Projects Agency (DARPA) [9]. The GARD program aims to develop a new generation of artificial intelligence adversarial defense systems that can deal with a broad class of adversarial attacks, rather than defense methods for highly specific adversarial attacks [10]. The GARD project provides a series of toolkits, such as the Armory virtual platform [9] and the Adversarial Robustness Toolbox [11], which, respectively, include a large number of defense [12–14] and attack [15–17] methods. These allow all members

participating in the project to submit their methods for inclusion and to conduct defense and attack performance evaluation.

At present, according to different defense strategies, adversarial defense mechanisms can be roughly divided into two categories [18,19]: passive defense (PD) and active defense (AD). Passive defense mechanisms mitigate the impact of adversarial attacks in the image domain by applying various transformation methods, including input gradient regularization [20], adversarial sample detection [21], and region-based classification [22]. However, passive defense mechanisms are highly dependent on expert knowledge and often can only target specific adversarial attack algorithms, which causes information loss [23].

As shown in Figure 1, compared with passive defense, active defense mechanisms improve the model's adversarial robustness by designing different DNN models or changing the neural network structure, with almost no loss of DNN detection performance. Active defense methods can be divided into three categories: adversarial training [24], creating gradient masks [25], and model modification [26]. Adversarial training improves the model's robustness by including adversarial perturbations in the input data, gradient mask defense protects the most critical weights (making it impossible for an adversary to launch an effective attack), and defense methods based on model modification ensure robust output, even in the presence of small perturbations [27].

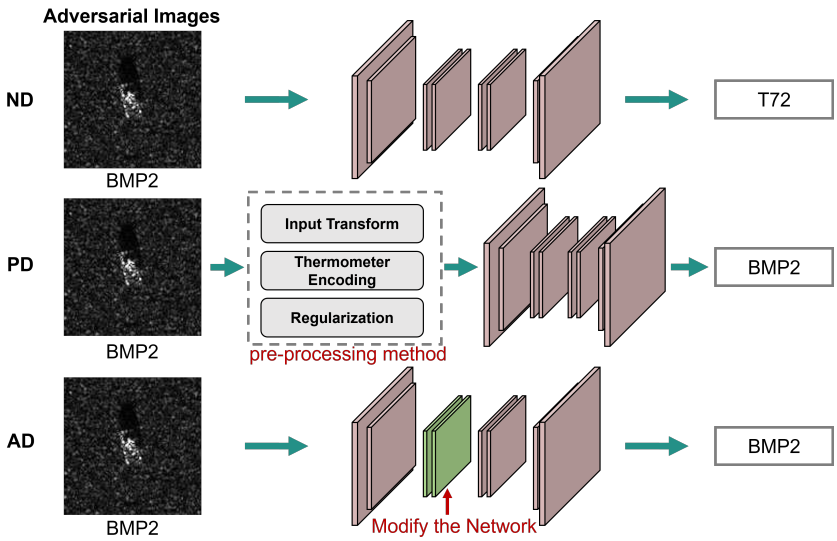


Figure 1. Comparison of no defense (ND), active defense (AD), and passive defense (PD). Active defense is robust by adjusting the network, while passive defense is defended by preprocessing operations outside the network.

Although the active defense mechanism has certain advantages, some of its shortcomings cannot be ignored. Adversarial training can fundamentally improve the robustness of the model by retraining the network with adversarial examples, but its training efficiency is low and difficult to scale to complex tasks and large datasets [28]. Gradient masking aims to hide model gradients from potential attackers, which can effectively block most attack methods, but it has been proven to be ineffective because attackers can still compute gradients using different methods [29]. Model modifying is feasible for a specific adversarial attack algorithm, but it is difficult to scale to other tasks or other adversarial attack algorithms [27].

As the most classic active adversarial defense method, standard adversarial training [28] mixes adversarial examples with clean images and inputs them into the neural network model for training to increase the robustness of the model. The stochastic activation pruning (SAP) [30] method randomly discards the parameters of each layer in the neural network, thereby giving the network higher robustness. However, as one of the gradient

masking defense methods, SAP has been proven to be ineffective because attackers can still use different methods [31,32] to calculate the gradient. The defensive distillation (DD) method [33] uses the predicted probability output by the first trained model as the ground truth to train a new model, thereby enhancing the robustness of the second trained model. However, the training process of this method is time-consuming and its adversarial robustness is uncertain. In addition, some new adversarial defense methods have emerged in recent years, such as attack-invariant attention features (AIAF-Defense) [34], Jacobian norm with selective input gradient regularization (JSIGR) [35], adaptive batch normalization (ABNN) [36], adversarial training through adaptive knowledge fusion (AT-AKA) [37], and debiased high-confidence logit alignment (DHAT) [38]. The AIAF-Defense method reconstructs the input data through an additional neural network to eliminate the impact of adversarial perturbations, but this method reduces the efficiency of the inference stage and requires a lot of time to train additional neural networks. The JSIGR method combines the regularization of the input data gradient with the saliency map, which also reduces the efficiency of the inference stage. The ABNN method uses a network pre-trained on large-scale clean data to adjust the BN statistical characteristics of the input data, but it is difficult to obtain such pre-trained models for fields such as remote sensing. AT-AKA uses at least three neural networks for parallel training to increase adversarial robustness, but the time and hardware costs are too high. The DHAT method uses reverse adversarial examples to train the model, which is essentially standard adversarial training.

In this paper, a general training method named gradual adversarial training (GAT) is proposed for DNNs. This is an active defense mechanism that affects the network weights during the neural network training process. It can strengthen the network's attention to salient features and suppress the network's attention to adversarial features without modifying the network structure and input data. Studies show that adding adversarial perturbations leads to considerable domain gaps in the distribution of clean images and adversarial examples in high-dimensional space [39]. In order to narrow the domain gap between clean images and adversarial examples, GAT creates multiple intermediate domains according to the input data during the training process, so that the neural network gradually adapts to the influence of high-dimensional feature representations that are unique to the adversarial domain. Therefore, by using adversarial training based on the theory of multiple intermediate domains to suppress the unique high-dimensional feature representation of the adversarial domain in the network, the neural network is not easily fooled by existing adversaries [40]. Furthermore, it can provide meaningful data augmentation, even if the removal of adversarial features is incomplete.

Our overall contributions can be summarized as follows:

- A novel adversarial defense method is proposed to learn robust feature representations based on the domain generalization theory;
- A gradual adversarial training method is proposed to enhance the robustness of the network without requiring specific information about the target network's architecture. By controlling the input data flow based on model weights external to the network structure, our proposed method can adaptively achieve model robustness enhancement;
- The proposed method is verified on both SAR and optical images, which proves that the proposed method is suitable for image segmentation with various attack intensities. The results show that, in SAR images, for commonly used methods (FGSM, PGD, etc.), when the attack intensity varies from 0.001 to 0.010, the accuracy of GAT improves by 0.5% to 4.23%, with an average of 1.57%. F1 improves by 1.31–5.02%, with an average of 3.24%. In optical images, when the attack intensity changes from 0.0039 to 0.196, the accuracy of the GAT method increases by 4.94% to 12.13%, with an average of 7.94%. F1 improves by 5.2–14.28%, with an average of 8.86%.

2. Background

In this section, the background of adversarial defense for semantic segmentation is presented. This includes a discussion of adversarial attack and defense theory, highlighting the importance of adversarial defense research. The key attack methods, such as Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), and segPGD, and the key defense methods, such as standard adversarial training (SAT), are introduced.

2.1. Adversarial Examples

Since adversarial examples were first proposed in 2013 [4], researchers have developed a variety of attack algorithms to generate them. In computer vision, classification is a focal point of adversarial attack research [41–43], but only a few works have investigated methods that are suitable for more intensive prediction tasks, such as semantic segmentation tasks. However, as the academic research on adversarial examples in the field of image classification has reached a high level, some researchers have turned their attention to the exploration of adversarial examples related to semantic segmentation [44–46]. Semantic segmentation, while viewed as a per-pixel classification problem, presents unique challenges in designing adversarial attacks [47]. In classification, generating the smallest adversarial sample involves a non-convex constrained problem with a single constraint. However, in segmentation, this optimization problem introduces multiple constraints because each pixel of the image must satisfy at least one constraint. As a result, most attacks originally designed for classification cannot be directly extended to segmentation. Listed below are several adversarial attack algorithms that are considered suitable for the field of semantic segmentation.

2.1.1. Fast Gradient Sign Method

Fast Gradient Sign Method (FGSM) is one of the earliest adversarial attack methods, introduced by Goodfellow et al. [48]. It is a single-step attack which aims to find the adversarial perturbations by moving in the opposite direction to the gradient of the loss function:

$$x_{adv} = x_c + \varepsilon \cdot sign(\nabla(L)) \tag{1}$$

where x_c are the clean data, x_{adv} are the adversarial sample data, $sign()$ is the sign function, $\nabla()$ is the gradient function, L is the loss function, and ε is the step size.

2.1.2. Dense Adversarial Generation

Dense Adversarial Generation (DAG) is a simple and effective algorithm proposed by Xie et al., which has made significant contributions to the field of segmentation adversarial attacks [49]. The algorithm generates adversarial perturbations for dense prediction tasks, including object detection and segmentation. DAG iteratively adds rescaled gradients of loss relative to the input to the current perturbation until a stopping criterion is reached, typically when a certain percentage of pixels become adversarial. The total loss per iteration is the sum of the losses of the non-adversarial pixels, similar to a greedy algorithm. Excluding the consideration of iteration, the equation of the DAG algorithm for generating disturbance is as follows:

$$x_{adv} = x_c + \sum_{i=1}^{N} (\nabla(L_i) - \nabla(L)) \tag{2}$$

where x_c are the clean data, x_{adv} are the adversarial sample data, N is a hyperparameter that means that N attack targets are generated, $\nabla()$ is the gradient function, and L represents the loss function for the prediction of the i-th target. Although DAG is effective in practice, it accumulates gradients up to a stopping criterion without explicitly minimizing the considered norm.

2.1.3. Projected Gradient Descent

Projected Gradient Descent (PGD) attempts to find the perturbation that can cause the greatest DNN loss on a particular input, while keeping the size of the perturbation smaller than a specified amount [50]. This is accomplished by using a multi-step perturbation, and the iterative equation for the data is

$$x^{t+1} = \prod(x^t + \alpha \cdot (\nabla(L))) \tag{3}$$

where x^t are the input data of t iterations, and x^{t+1} are the output data after t iterations and also the input data to the $(t + 1)$ iteration.

2.1.4. segPGD

On the basis of PGD, the segPGD algorithm takes into account the density of pixel classification in the iterative process, and, by adding a coefficient to the correct and wrong pixel classification loss function, it makes the attack more effective [51]. The loss function of segPGD is

$$L = \frac{1-\lambda}{H \times W} L_{correct} + \frac{\lambda}{H \times W} L_{wrong} \tag{4}$$

where L is the final loss function and $L_{correct}$ and L_{wrong} are the loss functions calculated from correctly classified pixels and incorrectly classified pixels, respectively. H and W correspond to the height and width of the input image. λ is a coefficient between 0 and 1.

2.2. Adversarial Defense

While generating adversarial examples has been extensively studied, there are also efforts to reduce the impact of adversarial examples, which is known as adversarial defense. The purpose of adversarial defense is to reduce the effect of adversarial disturbance, so that the predicted results are restored to the correct values.

As mentioned above, adversarial defense methods are divided into active defense methods and passive defense methods according to their implementation. Passive defense methods aim to eliminate adversarial perturbations before they are input into the model, while active defense methods enhance defense by directly or indirectly improving the robustness of the model. Existing studies [19] have shown that active defense methods usually do not increase the time cost in the inference phase, while passive defense methods will significantly increase time cost. This is because active defense methods enhance the robustness of the model during training, while passive defense methods require the preprocessing of inputs, even in the inference stage. This study is committed to improving the robustness of the neural network itself, so the active defense method was selected. The principles of some active adversarial defense methods are listed below.

2.2.1. Standard Adversarial Training

Standard adversarial training (SAT) improves robustness by retraining the network on adversarial example datasets, which can be formulated as a min–max optimization problem as follows [28,52]:

$$\underset{w}{\operatorname{argmin}}\, E_{(x_c,y)}[\max_r L(f_w(x_c + r), y)] \tag{5}$$

where r represents the target disturbance, x_c represents the clean image, y represents the corresponding label, $f(\cdot)$ is the neural network model, and w is the weight of the model. To summarize, the standard adversarial training process is as follows: first, train on a clean dataset to obtain a trained network model; then, conduct adversarial attacks on the model to obtain an adversarial sample dataset; finally, use the adversarial sample dataset to retrain the network model to obtain a robust neural network model. However, it may not be easy to scale to complex tasks or large datasets due to the complexity of the training step.

2.2.2. Stochastic Activation Pruning

Stochastic activation pruning (SAP) incorporates randomness into neural networks [30]. The method drops out the parameters of each layer in the neural network in a non-uniform manner with a probability proportional to the absolute value of the current weight. Its principle can be expressed in the following equation:

$$\underset{w}{\operatorname{argmin}} \max_{r} E_{(x_c,y)}[L(f_w(M(|w|), x_c + r), y)] \tag{6}$$

where $M(|w|)$ represents the loss of w, and its probability is proportional to the absolute value of w. The introduction of randomness enables the neural network to pay attention to more features, thereby endowing the network with higher robustness. However, as one of the gradient masking defense methods, SAP has been shown to be ineffective because attackers can still compute gradients using a different model [31,32].

2.2.3. Defensive Distillation

Defensive distillation (DD) is a defensive method that aims to smooth gradient changes and is trained using knowledge extracted from DNNs so that it is less affected when facing counterattacks [33]. Specifically, defensive distillation first completes model training, and then uses the predicted probabilities of different categories obtained during training as true values to train a new model to obtain a more robust model:

$$\underset{w}{\operatorname{argmin}} \; -E_{(x_c,y)}[\sum_{i}^{N} \vec{y}_i \cdot log f_i(x_c)] \tag{7}$$

where w is the weight of the model, x_c represents the clean input data, $\vec{y}$ represents the label corresponding to x_c, y_i is a vector representing the i-th class, and $f_i(x_c)$ represents the probability that x_c is predicted to be the i-th class.

2.2.4. Summary

The above methods have different scopes of application and each has its own advantages and disadvantages. Among them, adversarial training has always been considered the most effective method to enhance robustness. Standard adversarial training enhances model robustness by generating adversarial examples and injecting them into training data. The method proposed in this article is an improvement on adversarial training.

3. Methodology

In this section, we first introduce the proposed gradual adversarial training method. Then, we describe the evaluation metrics used to evaluate DNN models.

3.1. Gradual Adversarial Training Method

The proposed method is based on the manifold hypothesis theory and domain generalization. As shown in Figure 2, the manifold hypothesis assumes that natural images exist on a low-dimensional manifold, while, due to the high-dimensional nature of deep neural networks, adversarial images deviate from the low-dimensional manifold of natural images [53]. Previous studies have shown that binary classifiers can effectively distinguish adversarial images from clean images, thus establishing the distinction between visually identical but substantially different adversarial and clean data [54], which further confirms that there is a feature shift between the natural and adversarial images that can be detected by deep neural networks.

Domain generalization is an important research direction of the domain adaptive mechanism in deep learning, which is devoted to solving the problem that the distribution of training data (source domain) is different from that of test data (target domain). According to the manifold hypothesis theory, clean images and adversarial images have different distributions in high dimensions, so it is feasible to use the domain generalization method

to solve the adversarial defense problem. The basic domain-adaptive mechanism takes the source domain (S) and the target domain (T) as input and generates intermediate domain data (D) that contain information from both domains. This is represented as

$$D = \{f(x^s, x^t), y^s\} \tag{8}$$

where x and y represent the training data and the corresponding labels, s represents the source domain data in the form of $\{(x^s, y^s)\}$, and t represents the target domain data as $\{(x^t)\}$. The function $f(.)$ captures the relationship between the two domains and produces intermediate domain data with corresponding labels.

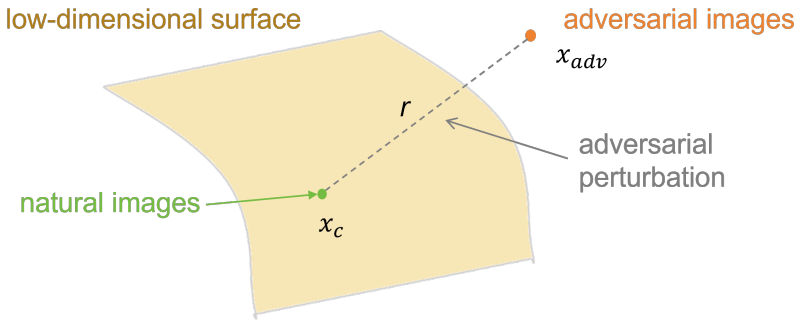


Figure 2. Schematic diagram of the manifold hypothesis. Natural images lie on a low-dimensional manifold, while images with adversarial perturbations added to them lie outside the low-dimensional manifold.

Domain generalization uses multiple source domain data with similar characteristics to train deep neural networks and can achieve satisfactory detection performance on target domain data. Inspired by this, this study proposes an improved adversarial training method, GAT, whose flowchart is shown in Figure 3.

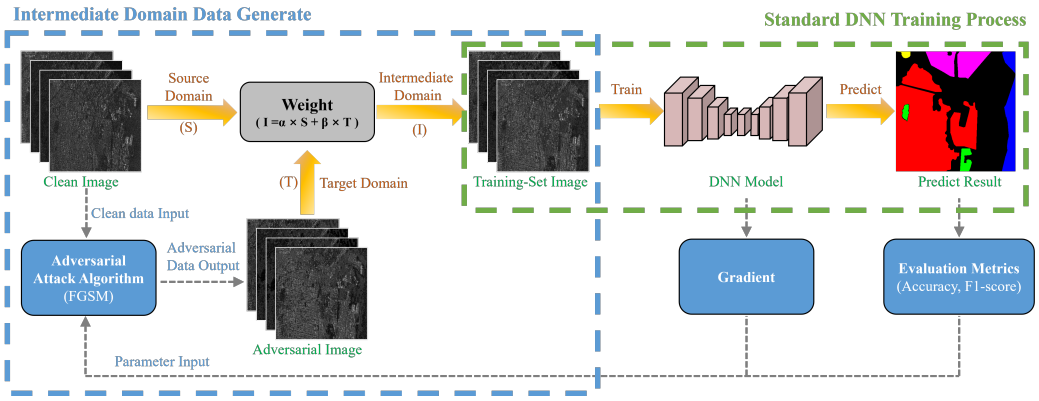


Figure 3. GAT training flowchart. The GAT method proposed in this paper can be divided into two modules: intermediate domain data generation and standard DNN training process. The intermediate domain data generation module generates intermediate domain data based on clean images and uses them as input to the latter. The standard DNN training process trains the model based on the input data and provides the former with parameters for the generation of adversarial perturbations.

This advanced adversarial training method considers the clean data as the source domain, regards the data with added adversarial perturbation as the target domain, and generates additional intermediate domain data based on both source and target domains. The i-th intermediate domain, denoted as $G(i)$, can be expressed as

$$G(i) = \alpha \times S + \beta \times T(i) \tag{9}$$

where α and β are the domain factors controlling the influence of the clean domain (S) and the current adversarial domain data ($T(i)$), respectively; the sum of α and β is 1.

FGSM is employed to generate the data in the i-th adversarial domain. Because the direction of the input which can be perturbed to deceive the model corresponds to that where the loss function increases fastest, given an input sample x and its corresponding true label y, the perturbation is computed as

$$x_i^t = x^s + \varepsilon \times \text{sign}(\nabla_x(L(x, y))) \tag{10}$$

where x^s represents the source data, x_i^t represents the i-th generated adversarial example, ε is the attack intensity (which refers to the maximum change in the magnitude of each pixel), and $\nabla_x(L(x, y))$ represents the gradient of the loss function, which measures the difference between the predicted output of the model and the true label y.

3.2. Framework

The algorithm of the proposed method, GAT, belongs to the active defense method, as described in Algorithm 1. The GAT method is applied to each epoch of the training process: for the i-th epoch, the input clean data are first fed into the attack algorithm to generate adversarial data; then, the clean data and perturbed data are weighted and superimposed according to Equation (9) to obtain the intermediate domain data. Next, the intermediate domain data are fed into the model for training. Finally, the updated parameters are provided to the attack algorithm for the next epoch. Steps 5–8 are parameter updates for the adversarial data generation method. In step 3, if the accuracy index does not improve, this means that the neural network has not fully learned the high-dimensional perturbations added in the previous epoch, so it is necessary to skip the parameter update.

Algorithm 1. Gradual Adversarial Training (GAT) Method

Input: Clean image x_c and ground truth y
Output: Robust model f

1. Train the model f using data x_c and labels y
2. Generate output results and evaluation metrics
3. If the evaluation metrics do not improve, repeat steps 1–2
4. Let $x = x_c$
5. Generate adversarial images x_{adv} according to Equation (10)
6. Generate intermediate data x_{inter} according to Equation (9)
7. Train the model f using data x_{inter} and labels y
8. Repeat steps 5–7 until the model converges
9. Output model f

Compared to SAT, which directly informs the distribution of data with specific perturbations, the proposed method dynamically adjusts the input data to force the model to pay less attention to the effect of high-dimensional features, which can be easily exploited by adversarial attack algorithms. In addition, the proposed method can simultaneously improve the segmentation accuracy on non-adversarial images. The code for this paper will be accessible at https://github.com/ykliming/GAT, accessed on 13 November 2024.

3.3. Evaluation Metrics

In this paper, accuracy and F1 score are used to exhibit the effectiveness of defense methods that are commonly used to evaluate segmentation performance.

Accuracy measures the overall pixel-level classification accuracy of the model by calculating the ratio of correctly classified pixels to the total number of pixels of the image. It provides an overall indication of the extent to which the model is able to classify pixels into their respective semantic categories. The accuracy rate can be expressed as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where TP represents true positives (the number of correct positive predictions), TN represents true negatives (the number of correct negative predictions), FP represents false positives (the number of incorrect positive predictions), and FN represents false negatives (the number of incorrect negative predictions).

$F1$ score is a metric that considers both *precision* and *recall* to evaluate the performance of the model. It provides a balanced measure of the model's ability to correctly classify foreground and background pixels. $F1$ score is calculated as the harmonic mean of *precision* and *recall*, as follows:

$$F1score = 2 \times \frac{precision \times recall}{precision + recall} \tag{12}$$

where *precision* is the ratio of true positives to the total number of predicted positives, and *recall* is the ratio of true positives to the total number of actual positives.

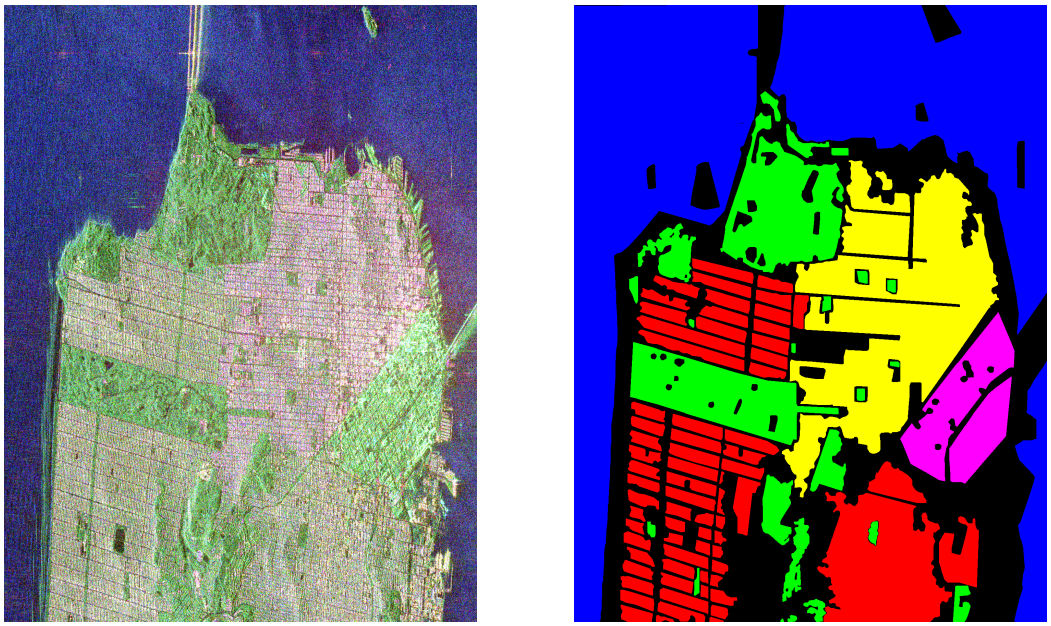
4. Experiments

In the experiment, SAR images of the San Francisco area and optical datasets of the Vaihingen area were used for semantic segmentation. FGSM, DAG, PGD, and segPGD were used to attack the same model to test the defense effect of a defenseless model, SAT [52], and GAT. Several groups of attack intensity were selected as comparative experiments.

4.1. Models and Datasets

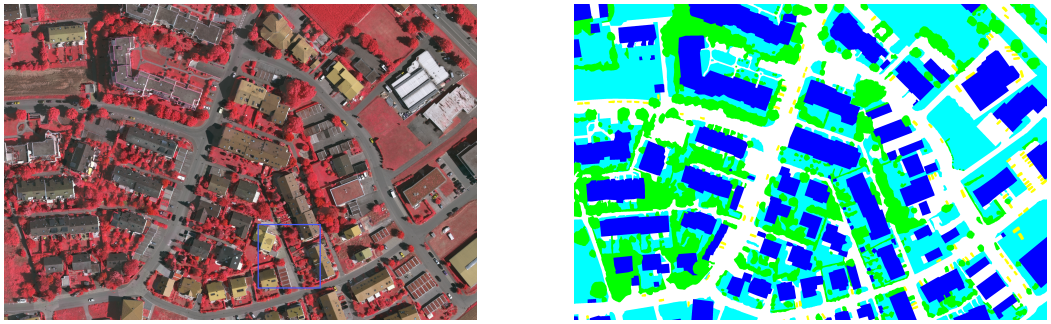
In our experiments, we used UNet with randomly initialized weights and trained the model until convergence. Compared with the codec structure, Unet can simultaneously extract the pixel-level features and semantic-level features of images. Compared with other neural networks, UNet has the advantage of a simple structure. The evaluation was performed on two datasets, the first one being the SF-RS2 dataset [55], containing a fully polarized SAR image from Radarsat-2. The image covers an area of size 1380 × 1800 pixels centered on San Francisco with a spatial resolution of 8m and 5 classes, namely, water, vegetation, high-density urban, low-density urban, and development areas. In this experiment, the SAR image was cropped into 446 slice samples with a size of 256 × 256, and the training set and test set were randomly divided with a division ratio of 8:2. The SAR image used as input data is a fully polarized amplitude image, and the amplitude of the image has been quantized to [0, 1]. The Pauli decomposition diagram and label truth value of the San Francisco data are shown in Figure 4. In the subsequent demonstration of experimental results, we chose the middle area as the display because it contains three types of ground objects, making it the most representative area in the entire image.

The second dataset, the ISPRS-Vaihingen dataset, consists of 33 very fine spatial resolution optical image tiles with an average size of 2494 × 2064 pixels. The dataset includes five foreground classes (impermeable surfaces, buildings, low vegetation, trees, and cars) and one background class (clutter). We used IDs 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13, 14, 15, 16, 17, and 20 for training and the remaining 16 images for testing. Image blocks were cropped into 512 × 512 px patches. The dataset contains several RGB three-channel optical images, each of which is quantized to [0, 255]. Since there are 33 images in the ISPRS-Vaihingen dataset, 1 image (Area 38) and its label were selected to be shown in Figure 5.



(**a**) (**b**)

Figure 4. A presentation of data from San Francisco. (**a**) Pauli decomposition result and (**b**) Ground truth. Blue is water, green is vegetation, red is high-density urban, yellow is low-density urban, and purple is development areas, black is unlabeled background.



(**a**) (**b**)

Figure 5. A presentation of data from Vaihingen. The blue box area in the left figure is the selected typical area with 4 different types of ground objects, which are used for analysis in the subsequent presentation of the experimental results. (**a**) ISPRS-Vaihingen dataset example and (**b**) Ground truth. Blue is buildings, light blue is low vegetation, green is trees, yellow is cars, red is the background, and white is impervious surfaces.

4.2. Experimental Settings and Implementation Details

All the models in the experiments were implemented with the PyTorch framework (version 1.13) on a single NVIDIA GTX 3090 GPU. For fast convergence, we deployed the Adam optimizer to train all models. The base learning rate was set to 1×10^{-5} for the SF-RS2 dataset and 1×10^{-4} for the ISPRS-Vaihingen dataset. The learning rate decay strategy was employed to adjust the learning rate.

In the experiments, the hyperparameters β and α of the GAT method were set to be 0.4 and 0.6, respectively, which were the optimal ratios obtained through extensive experiments. Due to the limitations of the dataset size and the learning ability of the neural network, the optimal values of β and α may vary depending on the dataset and network architecture, so these parameters need to be tested and adjusted according to the specific situation. But, in general, when α increases, the accuracy on clean images and the accuracy on adversarial perturbations will both increase. However, α should not be too large, as too large an α may lead to a decrease in accuracy. In this experiment, when α

was 0.6, the accuracy of clean images and that of adversarial images both reached a better value, so α was taken as 0.6 and β was taken as 0.4. In data preprocessing, images on both the SF-RS2 dataset and the ISPRS-Vaihingen dataset were normalized to the interval [0, 1]. In the domain of machine learning and artificial intelligence, particularly in the context of adversarial machine learning, the term "attack intensity" typically denotes the magnitude of perturbation that an adversary applies to deceive or mislead the model. This perturbation is introduced into the input data to induce false predictions from the model, while striving to maintain imperceptibility to humans or at least to minimize its visibility so as not to be easily detected. In this paper, attack intensity was defined as the ratio between the size of perturbed pixel value and maximum pixel value, which is between 0 and 1. For the selection of the maximum attack intensity, this study took perturbation as the standard that can be detected by human eyes. Therefore, the maximum perturbation of the SF-RS2 dataset was selected as 0.010, and the maximum perturbation of the ISPRS-Vaihingen dataset was 0.0196. The SF-RS2 dataset images underwent perturbation in 10 stages with an attack step of 0.001 per stage. On the other hand, the ISPRS-Vaihingen dataset's optical image data were integer with an amplitude change of at least 1, so they were perturbed in 5 stages with a quantitative attack step of 0.0039 per stage, which corresponds to 1/255.

4.3. Result Analysis on SF-RS2 Dataset

The results of the experiments on the SF-RS2 dataset are shown in Table 1. We first evaluated the segmentation effects after applying different defense methods under non-adversarial conditions, and then sequentially evaluated the segmentation effects in the face of four adversarial attacks: FGSM, DAG, PGD, and segPGD.

Table 1. Segmentation results of the SF-RS2 dataset in the face of different adversarial attacks.

	No Attack		**FGSM [48]**		**DAG [49]**		**PGD [50]**		**segPGD [51]**	
	Acc (%)	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**
No Defense	95.7	94.95	56.89	49.99	67.47	64.27	44.85	37.14	49.63	41.72
SAT [52]	96.43	96.13	63.26	51.97	64.39	61.31	49.73	39.97	53.86	39.97
GAT	**97.54**	**97.07**	**64.08**	**54.39**	**68.62**	**65.52**	**50.34**	**41.28**	**54.36**	**44.99**

To assess the robustness and generalizability of the GAT method, we conducted comparative experiments using no defense training and standard adversarial training methods. The segmentation results of the model without adversarial training had an accuracy of 95.7%, which is comparable to previous studies [56]. The perturbation intensity of the adversarial attack algorithm in Table 1 was set to 0.01, and the bold parts are the defense methods that perform better when facing the same attack. It is evident that, for both no attack and data with different adversarial attack algorithms, the GAT method outperformed SAT and the no defense method in all cases. The accuracy of the segmentation results by the GAT method was 97.54%, while F1 score was 97.07%, with an improvement of about 2% for both metrics compared to the no defense method.

To investigate the effectiveness of the proposed method under varying attack intensities, controlled experiments were conducted under different attack intensities ranging from 0.001 to 0.01. The metric curves of the segmentation results are shown in Figure 6, with the columns denoting the various adversarial attack methods and the two rows denoting accuracy and F1 score, respectively. The blue lines in Figure 6 represent the curves of no defense, the orange lines are the curves of standard adversarial training, and the green lines are the curves of gradual adversarial training.

It can be seen from Figure 7 that the accuracy of the three methods decreased with an increase in attack intensity, while the method proposed in this paper was the highest in both accuracy and F1 score under different levels of attack intensity. Compared with the no defense method, when the attack intensity was 0.01, the accuracy of GAT in the face of FGSM, DAG, PGD, and segPGD improved by 7.19%, 1.15%, 5.49%, and 4.73%, respectively.

Compared with the SAT method, when the attack intensity was 0.01, the accuracy of GAT in the face of FGSM, DAG, PGD, and segPGD improved by 0.82%, 4.23%, 0.61%, and 0.50%, respectively. In the face of DAG attacks, the detection accuracy of SAT was even worse than that of the defenseless model. This may be due to the fact that only FGSM attack examples were provided when training SAT, while a DAG attack is an iteratively optimized attack based on loss function, and the principle of the FGSM attack is different.

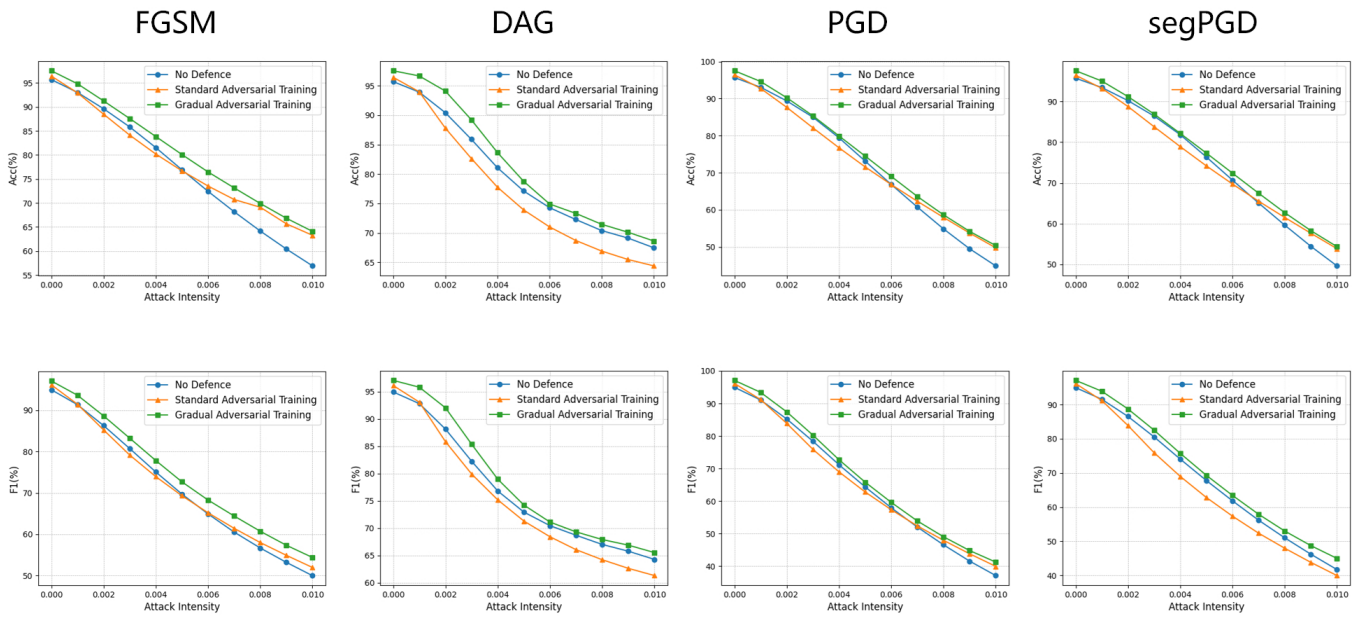


Figure 6. Metric curves of segmentation results on the SF-RS2 dataset facing adversarial attacks with different attack intensities. The first row shows the Acc evaluation index curve, and the second row shows the F1 score evaluation index curve. From the first column to the fourth column, the attack algorithms using FGSM, DAG, PGD, and segPGD are shown. The horizontal axis of each graph is the attack intensity, which ranges from 0.00 to 0.01, and the vertical axis is the evaluation index.

4.4. Result Analysis on ISPRS-Vaihingen Dataset

Table 2 shows the experimental results on the ISPRS-Vaihingen dataset. We first evaluated the segmentation performance after applying different defense methods under non-adversarial conditions and then sequentially evaluated the segmentation performance in the face of four adversarial attacks: FGSM, DAG, PGD, and segPGD.

Table 2. Segmentation results of the ISPRS-Vaihingen dataset in the face of different adversarial attacks.

	No Attack		**FGSM**		**DAG**		**PGD**		**segPGD**	
	Acc (%)	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**	**Acc (%)**	**F1 (%)**
No Defense	75.6	71.39	48.2	40.78	66.11	66.62	31.09	28.22	24.49	24.49
SAT	**77.18**	**78.41**	44.27	40.58	60.95	59.18	33.60	27.98	21.22	18.84
GAT	77.03	77.53	**49.21**	**45.78**	**70.18**	**73.46**	**39.05**	**33.42**	**33.35**	**29.35**

In order to evaluate the robustness and generalization of the GAT method, we used no defense training and standard adversarial training methods as comparative experiments. The segmentation results of the model without adversarial training had an accuracy of 75.6%, which is comparable to previous studies [57]. The attack intensity of the adversarial attack algorithm in Table 2 was set as 0.0196, and the bold parts are the defense methods that perform better when facing the same attack. It is evident that, for data with different adversarial attack algorithms, the GAT method outperformed SAT and the no defense

method in all cases. The accuracy of the segmentation results by the GAT method was 77.03%, while F1 score was 77.53%, with an improvement of about 1.43% for accuracy and 5.14% for F1 score compared to the no defense method.

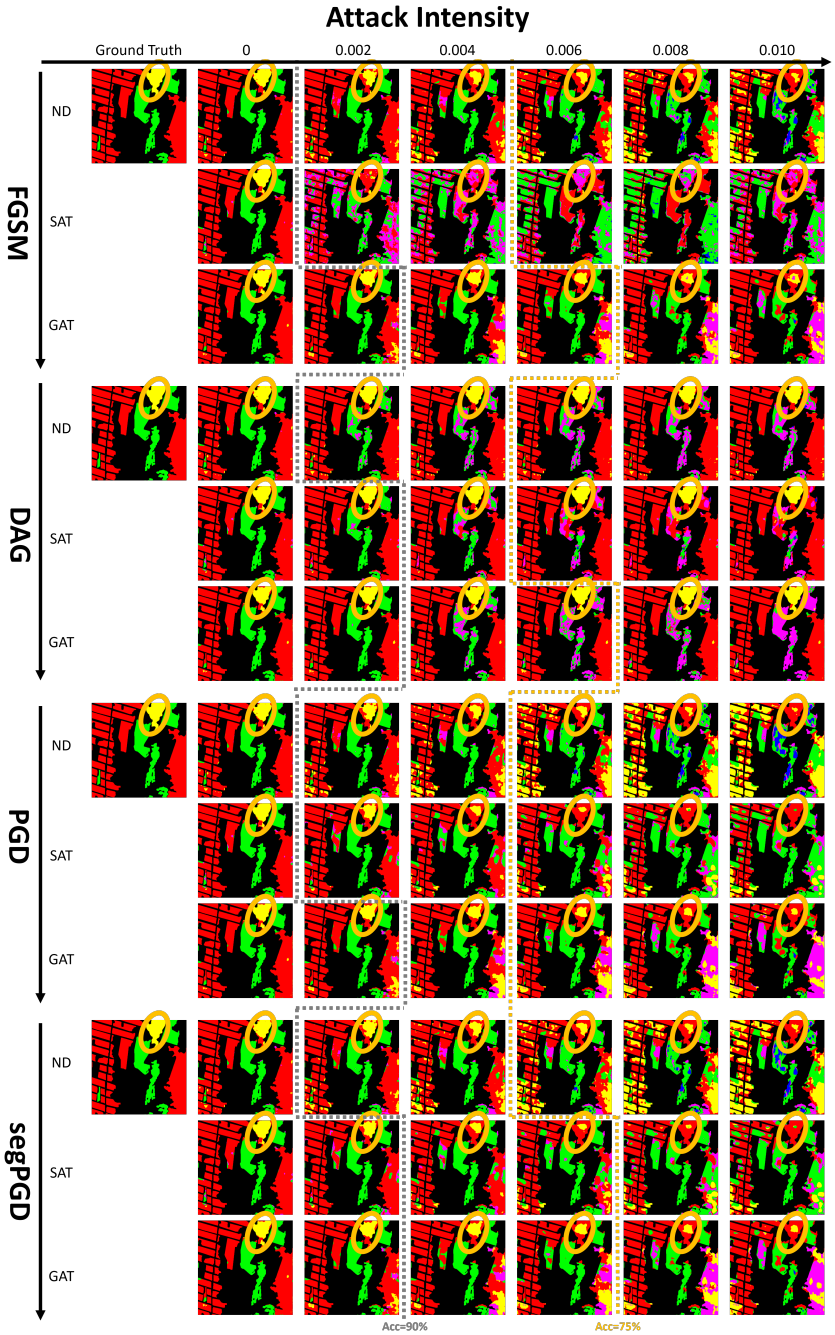


Figure 7. SF-RS2 dataset segmentation results in the face of FGSM, DAG, PGD, and segPGD attacks. The attack intensity ranges from 0 to 0.01. For each adversarial attack algorithm, the segmentation results of no defense, SAT, and GAT are compared in turn. Taking the yellow circle as an example, the feature type of the area is high-density city, and it can clearly be seen that the segmentation accuracy of GAT is better than that of SAT and no defense. The dotted gray lines correspond to 90% accuracy and the dotted yellow lines correspond to 75% accuracy. When accuracy is reduced to the same level, the GAT method can withstand a stronger attack intensity.

Controlled experiments were conducted in this study to investigate the effectiveness of the proposed method under different attack intensities. The attack intensity varied from 0 to 0.0157 with an increment of 0.0039. The metrics curves of the segmentation results are shown in Figure 8, with the columns denoting various adversarial attack methods and the two rows denoting accuracy and F1 score. The blue lines in Figure 8 are the curves of no defense, the orange lines are those of standard adversarial training, and the green lines are those of gradual adversarial training.

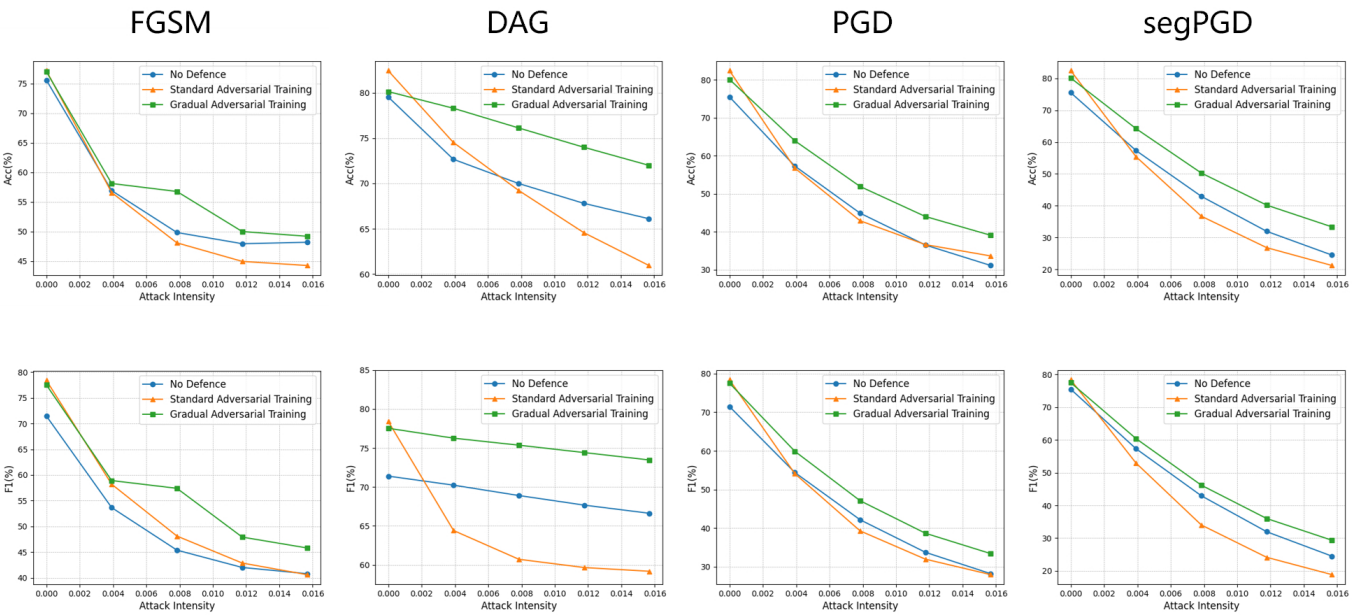


Figure 8. Metric curves of segmentation results on the ISPRS-Vaihingen dataset facing adversarial attacks with different attack intensities. The first row shows the Acc evaluation index curve, and the second row shows the F1 score evaluation index curve. From the first column to the fourth column, the attack algorithms using FGSM, DAG, PGD, and segPGD are shown. The horizontal axis of each graph is the attack intensity, which ranges from 0.00 to 0.0157, and the vertical axis is the evaluation index.

It can be seen from Figure 9 that the accuracy of the three methods decreased with the increase in attack intensity, while the method proposed in this paper was the highest in both accuracy and F1 score under different levels of attack intensity. Compared with the no defense method, when the attack intensity was 0.0196, the accuracy of GAT in the face of FGSM, DAG, PGD, and segPGD improved by 1.01%, 4.07%, 7.96%, and 8.89%, respectively. Compared with the SAT method, when the attack intensity was 0.01, the accuracy of GAT in the face of FGSM, DAG, PGD, and segPGD improved by 4.94%, 9.23%, 5.45%, and 12.13%, respectively. On the ISPRS-Vaihingen dataset, SAT performed better on unattacked examples, but, in the face of adversarial attacks, the accuracy of SAT was even worse than that of the undefended model. This may be due to the fact that SAT overfitted the dataset through two training sessions, basic training and adversarial training, resulting in special vulnerabilities.

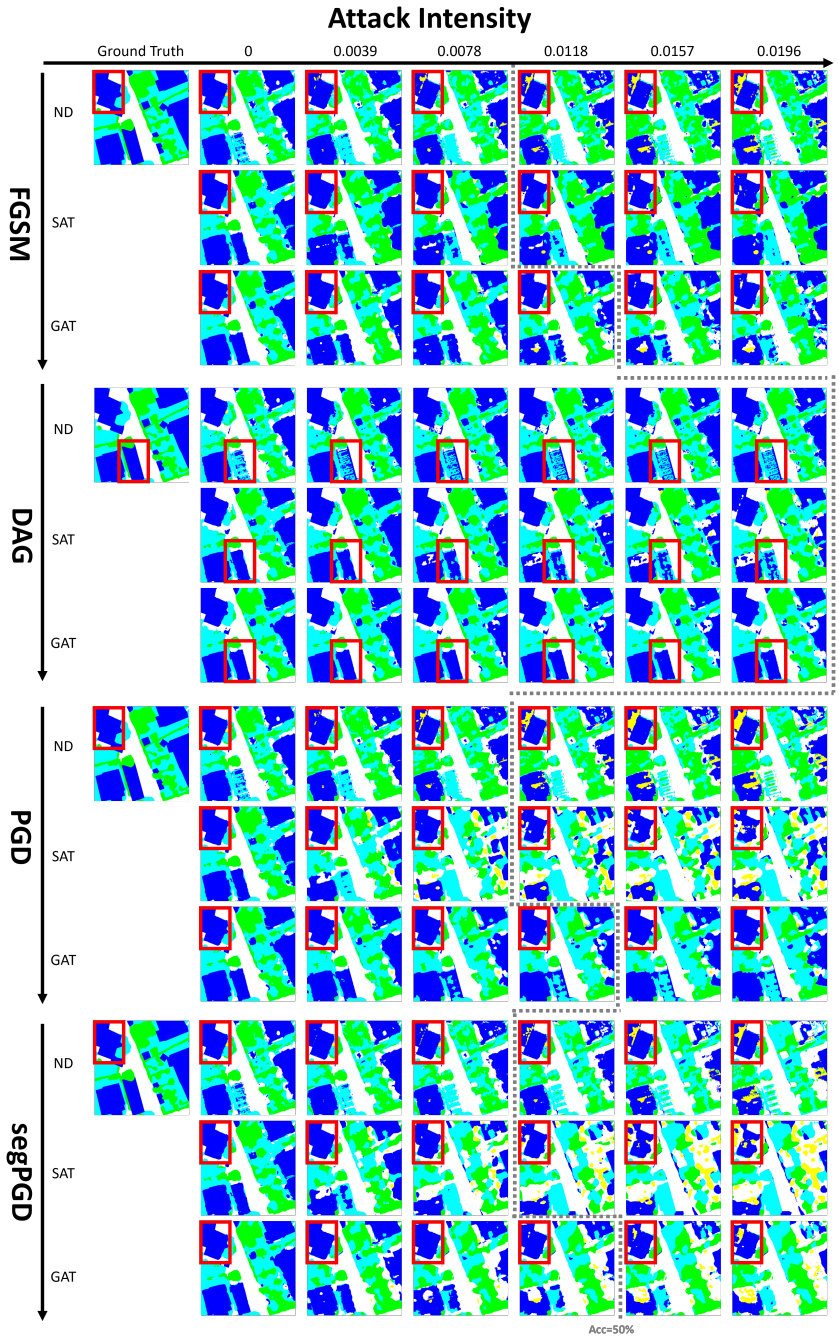


Figure 9. ISPRS-Vaihingen dataset segmentation results in the face of FGSM, DAG, PGD, and segPGD attacks. The attack intensity ranges from 0 to 0.0196. For each adversarial attack algorithm, the segmentation results of no defense, SAT, and GAT are compared in turn. Taking the red box area as an example, the feature type of this area is building, and it can clearly be seen that the segmentation accuracy of GAT is better than that of SAT and no defense. The dotted gray lines correspond to 50% accuracy. When accuracy is reduced to the same level, the GAT method can withstand a stronger attack intensity.

5. Discussion

The experimental results on the SF-RS2 and ISPRS-Vaihingen datasets in this paper comprehensively demonstrate the robustness of the defense method in the face of different adversarial attack algorithms. Because FGSM was used as the adversarial attack algorithm during training, the SAT-trained model had good robustness to gradient-based adversarial attacks such as FGSM and Basic Iterative Methods (BIMs). However, when

facing non-gradient-based adversarial attack algorithms such as DAG, the performance of the SAT-trained model was not as good as that of the defenseless model. In contrast, as shown in Tables 1 and 2, the GAT-trained model not only had better robustness than the SAT-trained model in the face of adversarial attack algorithms such as FGSM and BIM, but also maintained better adversarial robustness in the face of the DAG adversarial attack algorithm.

In order to evaluate the robustness of the defense methods under different attack intensities, experiments with multiple attack intensities were conducted on the two datasets. The models used in the defenseless, SAT, and GAT experiments did not change in experiments with different attack intensities. The experimental results showed that the accuracy of the SAT-trained model decreased faster when facing higher-intensity adversarial perturbations. On the contrary, as shown in Figures 6 and 8, the accuracy of the model trained by GAT decreased more slowly when facing high-intensity adversarial perturbations and had better adversarial robustness.

When conducting experiments on the ISPRS-Vaihingen optical dataset, without adding adversarial attacks, the accuracy of the model trained by the GAT method was lower than that of the model trained by the SAT method, but still higher than the undefended model. This may be because both SAT and GAT play a role in data enhancement to a certain extent. GAT focuses on allowing the model to learn the changing laws of adversarial perturbations, while SAT training can have more resources for the model to learn the laws of the data themselves in the dataset.

Overall, the proposed GAT method combines the theory of adversarial training and domain generalization to achieve a robust and accurate remote sensing image semantic segmentation model, which is an effective adversarial defense method.

6. Conclusions

In this study, a gradual adversarial training (GAT) method is proposed to enhance the robustness of DNN models in semantic segmentation by combining adversarial training with domain generalization theory. Experimental results for two datasets, SAR and optical, show that GAT is effective and can resist the adversarial attacks of different methods and attack intensities, which not only improves the robustness of the model against adversarial attacks, but also improves the accuracy of model detection. The proposed method is an adversarial defense method that is universal for SAR and optical images and is not specifically designed for the polarization characteristics of SAR images. Future research will focus on improving the adversarial defense method based on the physical characteristics of remote sensing images [58,59] to resist more realistic interference.

Author Contributions: Conceptualization, Y.Z. and P.L.; methodology, Y.Z.; validation, Y.Z. and P.L.; writing—original draft preparation, Y.Z. and T.M.; writing—review and editing, Y.Z., P.L. and T.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are openly available in github at https://github.com/ykliming/GAT, accessed on 13 November 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DNN	Deep Neural Network
SAR	Synthetic Aperture Radar
SAT	Standard Adversarial Training
GAT	Gradual Adversarial Training
FGSM	Fast Gradient Sign Method
PGD	Projected Gradient Descent

References

1. Rony, J.; Pesquet, J.C.; Ben Ayed, I. Proximal splitting adversarial attack for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 20524–20533.
2. Wang, Z.; Yang, H.; Feng, Y.; Sun, P.; Guo, H.; Zhang, Z.; Ren, K. Towards transferable targeted adversarial examples. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 20534–20543.
3. Williams, P.N.; Li, K. Black-box sparse adversarial attack via multi-objective optimisation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 12291–12301.
4. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv* **2013**, arXiv:1312.6199.
5. Maslovskiy, A.; Vasilets, V.; Nechitaylo, S.; Sukharevsky, O. The Antiradar Camouflage Method for Ground Military Objects. In Proceedings of the 2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Lviv, Ukraine, 2–6 July 2019; pp. 1–5.
6. He, W.; Weng, X.; Luo, W.; Chen, H.; Wu, X.; Li, K.; Huang, Y.; Liu, B.; Li, L. Investigation of radar cross-section reduction for dihedral corner reflectors based on camouflage grass. *IEEE Antennas Wirel. Propag. Lett.* **2021**, *20*, 2447–2451. [CrossRef]
7. Smy, T.J.; Gupta, S. Surface susceptibility synthesis of metasurface skins/holograms for electromagnetic camouflage/illusions. *IEEE Access* **2020**, *8*, 226866–226886. [CrossRef]
8. Han, S.; Lin, C.; Shen, C.; Wang, Q.; Guan, X. Interpreting adversarial examples in deep learning: A review. *ACM Comput. Surv.* **2023**, *55*, 1–38. [CrossRef]
9. Monroe, D. Deceiving ai. *Commun. ACM* **2021**, *64*, 15–16. [CrossRef]
10. Siegelmann, H. Defending Against Adversarial Artificial Intelligence. Technical Report. 2019. Available online: https://www.darpa.mil/news-events/2019-02-06 (accessed on 6 February 2019).
11. Nicolae, M.I.; Sinn, M.; Tran, M.N.; Buesser, B.; Rawat, A.; Wistuba, M.; Zantedeschi, V.; Baracaldo, N.; Chen, B.; Ludwig, H.; et al. Adversarial Robustness Toolbox v1. 0.0. *arXiv* **2018**, arXiv:1807.01069.
12. Sreeram, A.; Mehlman, N.; Peri, R.; Knox, D.; Narayanan, S. Perceptual-based deep-learning denoiser as a defense against adversarial attacks on ASR systems. *arXiv* **2021**, arXiv:2107.05222.
13. Joshi, S.; Villalba, J.; Żelasko, P.; Moro-Velázquez, L.; Dehak, N. Study of Pre-Processing Defenses Against Adversarial Attacks on State-of-the-Art Speaker Recognition Systems. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 4811–4826. [CrossRef]
14. Lo, S.Y. Robust Computer Vision Against Adversarial Examples and Domain Shifts. Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, USA, 2023.
15. Chen, J.; Wu, X.; Guo, Y.; Liang, Y.; Jha, S. Towards evaluating the robustness of neural networks learned by transduction. *arXiv* **2021**, arXiv:2110.14735.
16. Zhang, Y.; Jiang, Z.; Villalba, J.; Dehak, N. Black-Box Attacks on Spoofing Countermeasures Using Transferability of Adversarial Examples. In Proceedings of the Interspeech, Shanghai, China, 25–29 October 2020; pp. 4238–4242.
17. Cherepanova, V.; Goldblum, M.; Foley, H.; Duan, S.; Dickerson, J.; Taylor, G.; Goldstein, T. Lowkey: Leveraging adversarial attacks to protect social media users from facial recognition. *arXiv* **2021**, arXiv:2101.07922.
18. Luo, Y.; Ye, F.; Weng, B.; Du, S.; Huang, T. A novel defensive strategy for facial manipulation detection combining bilateral filtering and joint adversarial training. *Secur. Commun. Netw.* **2021**, *2021*, 4280328. [CrossRef]
19. Jiang, W.; He, Z.; Zhan, J.; Pan, W. Attack-aware detection and defense to resist adversarial examples. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2020**, *40*, 2194–2198. [CrossRef]
20. Xie, C.; Wang, J.; Zhang, Z.; Ren, Z.; Yuille, A. Mitigating adversarial effects through randomization. *arXiv* **2017**, arXiv:1711.01991.
21. Zhang, S.; Chen, S.; Liu, X.; Hua, C.; Wang, W.; Chen, K.; Zhang, J.; Wang, J. Detecting adversarial samples for deep learning models: A comparative study. *IEEE Trans. Netw. Sci. Eng.* **2021**, *9*, 231–244. [CrossRef]
22. Cao, X.; Gong, N.Z. Mitigating evasion attacks to deep neural networks via region-based classification. In Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, 4–8 December 2017; pp. 278–287.
23. Liu, N.; Du, M.; Guo, R.; Liu, H.; Hu, X. Adversarial attacks and defenses: An interpretation perspective. *ACM SIGKDD Explor. Newsl.* **2021**, *23*, 86–99. [CrossRef]
24. Wei, Z.; Wang, Y.; Guo, Y.; Wang, Y. Cfa: Class-wise calibrated fair adversarial training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 8193–8201.
25. Boenisch, F.; Sperl, P.; Böttinger, K. Gradient masking and the underestimated robustness threats of differential privacy in deep learning. *arXiv* **2021**, arXiv:2105.07985.
26. Tomar, D.; Vray, G.; Bozorgtabar, B.; Thiran, J.P. Tesla: Test-time self-learning with automatic adversarial augmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 20341–20350.
27. Qiu, S.; Liu, Q.; Zhou, S.; Wu, C. Review of artificial intelligence adversarial attack and defense technologies. *Appl. Sci.* **2019**, *9*, 909. [CrossRef]
28. Jia, X.; Zhang, Y.; Wei, X.; Wu, B.; Ma, K.; Wang, J.; Cao, X. Improving fast adversarial training with prior-guided knowledge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 6367–6383. [CrossRef]

29. Bae, H.; Jang, J.; Jung, D.; Jang, H.; Ha, H.; Lee, H.; Yoon, S. Security and privacy issues in deep learning. *arXiv* **2018**, arXiv:1807.11655.
30. Dhillon, G.S.; Azizzadenesheli, K.; Lipton, Z.C.; Bernstein, J.; Kossaifi, J.; Khanna, A.; Anandkumar, A. Stochastic activation pruning for robust adversarial defense. *arXiv* **2018**, arXiv:1803.01442.
31. Yanagita, Y.; Yamamura, M. Gradient masking is a type of overfitting. *Int. J. Mach. Learn. Comput.* **2018**, *8*, 203–207. [CrossRef]
32. Zhou, M.; Wang, L.; Niu, Z.; Zhang, Q.; Zheng, N.; Hua, G. Adversarial attack and defense in deep ranking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 5306–5324. [CrossRef] [PubMed]
33. Papernot, N.; McDaniel, P.; Wu, X.; Jha, S.; Swami, A. Distillation as a defense to adversarial perturbations against deep neural networks. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 23–25 May 2016; pp. 582–597.
34. Shi, C.; Liu, Y.; Zhao, M.; Pun, C.M.; Miao, Q. Attack-invariant attention feature for adversarial defense in hyperspectral image classification. *Pattern Recognit.* **2024**, *145*, 109955. [CrossRef]
35. Liu, D.; Wu, L.Y.; Li, B.; Boussaid, F.; Bennamoun, M.; Xie, X.; Liang, C. Jacobian norm with selective input gradient regularization for interpretable adversarial defense. *Pattern Recognit.* **2024**, *145*, 109902. [CrossRef]
36. Lo, S.Y.; Patel, V.M. Adaptive Batch Normalization Networks for Adversarial Robustness. *arXiv* **2024**, arXiv:2405.11708.
37. Hamidi, S.M.; Ye, L. Adversarial Training via Adaptive Knowledge Amalgamation of an Ensemble of Teachers. *arXiv* **2024**, arXiv:2405.13324.
38. Zhang, K.; Weng, J.; Luo, Z.; Li, S. Towards Adversarial Robustness via Debiased High-Confidence Logit Alignment. *arXiv* **2024**, arXiv:2408.06079.
39. Mustafa, A.; Khan, S.H.; Hayat, M.; Shen, J.; Shao, L. Image super-resolution as a defense against adversarial attacks. *IEEE Trans. Image Process.* **2019**, *29*, 1711–1724. [CrossRef]
40. Yu, S.; Wang, S. Multi-intermediate Feature with Multi-stage Fusion for Domain Adaptive Person Re-ID. In Proceedings of the 2023 6th International Conference on Image and Graphics Processing, Chongqing, China, 6–8 January 2023; pp. 36–43.
41. Wei, X.; Yuan, M. Adversarial pan-sharpening attacks for object detection in remote sensing. *Pattern Recognit.* **2023**, *139*, 109466. [CrossRef]
42. Huang, J.J.; Wang, Z.; Liu, T.; Luo, W.; Chen, Z.; Zhao, W.; Wang, M. DeMPAA: Deployable Multi-Mini-Patch Adversarial Attack for Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5623613. [CrossRef]
43. Liu, L.; Xu, Z.; He, D.; Yang, D.; Guo, H. Local pixel attack based on sensitive pixel location for remote sensing images. *Electronics* **2023**, *12*, 1987. [CrossRef]
44. Bai, T.; Cao, Y.; Xu, Y.; Wen, B. Stealthy Adversarial Examples for Semantic Segmentation in Remote Sensing. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5614817. [CrossRef]
45. Yu, Z.; Yang, W.; Xie, X.; Shi, Z. Attacks on Continual Semantic Segmentation by Perturbing Incremental Samples. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 6844–6852.
46. Agnihotri, S.; Jung, S.; Keuper, M. CosPGD: An efficient white-box adversarial attack for pixel-wise prediction tasks. In Proceedings of the Forty-First International Conference on Machine Learning, Vienna, Austria, 21–27 July 2024.
47. Cheng, B.; Schwing, A.; Kirillov, A. Per-pixel classification is not all you need for semantic segmentation. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 17864–17875.
48. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* **2014**, arXiv:1412.6572.
49. Xie, C.; Wang, J.; Zhang, Z.; Zhou, Y.; Xie, L.; Yuille, A. Adversarial examples for semantic segmentation and object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1369–1378.
50. Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv* **2017**, arXiv:1706.06083.
51. Gu, J.; Zhao, H.; Tresp, V.; Torr, P.H. Segpgd: An effective and efficient adversarial attack for evaluating and boosting segmentation robustness. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 308–325.
52. Kurakin, A.; Goodfellow, I.; Bengio, S. Adversarial machine learning at scale. *arXiv* **2016**, arXiv:1611.01236.
53. Malinin, A.; Gales, M. Prior networks for detection of adversarial attacks. *arXiv* **2018**, arXiv:1812.02575.
54. Gong, Z.; Wang, W. Adversarial and clean data are not twins. In Proceedings of the Sixth International Workshop on Exploiting Artificial Intelligence Techniques for Data Management, Seattle, WA, USA, 18 June 2023; pp. 1–5.
55. Liu, X.; Jiao, L.; Liu, F.; Zhang, D.; Tang, X. PolSF: PolSAR image datasets on san Francisco. In Proceedings of the International Conference on Intelligence Science, Xi'an, China, 28–31 October 2022; pp. 214–219.
56. Zhang, Z.; Guo, H.; Yang, J.; Wang, X.; Du, Y. Adversarial network with higher order potential conditional random field for PolSAR image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *17*, 1795–1812. [CrossRef]
57. Zhang, H.; Jiang, Z.; Zheng, G.; Yao, X. Semantic segmentation of high-resolution remote sensing images with improved U-Net based on transfer learning. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 181. [CrossRef]

58. Li, M.; Zou, H.; Dong, Z.; Qin, X.; Liu, S.; Zhang, Y. Unsupervised Semantic Segmentation of PolSAR Images Based on Multi-view Similarity. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 5317–5331. [CrossRef]
59. Zhang, S.; Cui, L.; Dong, Z.; An, W. A Deep Learning Classification Scheme for PolSAR Image Based on Polarimetric Features. *Remote Sens.* **2024**, *16*, 1676. [CrossRef]