# Adversarial Examples for CNN-Based SAR Image Classification: An Experience Study

Haifeng Li, *Member, IEEE*, Haikuo Huang, Li Chen, Jian Peng, Haozhe Huang, Zhenqi Cui, Xiaoming Mei, and Guohua Wu

***Abstract*—Synthetic aperture radar (SAR) has all-day and all-weather characteristics and plays an extremely important role in the military field. The breakthroughs in deep learning methods represented by convolutional neural network (CNN) models have greatly improved the SAR image recognition accuracy. Classification models based on CNNs can perform high-precision classification, but there are security problems against adversarial examples (AEs). However, the research on AEs is mostly limited to natural images, and remote sensing images (SAR, multispectral, etc.) have not been extensively studied. To explore the basic characteristics of AEs of SAR images (ASIs), we use two classic white-box attack methods to generate ASIs from two SAR image classification datasets and then evaluate the vulnerability of six commonly used CNNs. The results show that ASIs are quite effective in fooling CNNs trained on SAR images, as indicated by the obtained high attack success rate. Due to the structural differences among CNNs, different CNNs present different vulnerabilities in the face of ASIs. We found that ASIs generated by nontarget attack algorithms feature attack selectivity, which is related to the feature space distribution of the original SAR images and the decision boundary of the classification model. We propose the sample-boundary-based AE selectivity distance to successfully explain the attack selectivity of ASIs. We also analyze the effects of image parameters, such as image size and number of channels, on the attack success rate of ASIs through parameter sensitivity. The experimental results of this study provide data support and an effective reference for attacks on and the defense capabilities of various CNNs with regard to AEs in SAR image classification models.**

***Index Terms*—Adversarial example (AE), convolutional neural network (CNN), synthetic aperture radar (SAR).**

## I. Introduction

SYNTHETIC aperture radar (SAR) is a sensor that actively emits microwaves, which improves the azimuth resolution through the principle of a synthetic aperture to obtain large-area high-resolution radar images. The SAR image is the image data acquired by the microwave band, which contains only the echo information of one band. It is recorded in the form of binary complex numbers. The data of each pixel can extract the corresponding amplitude and phase information. The amplitude information is the backscattering intensity of the radar wave by the ground target, and it can be used to identify and classify the target. The radar target recognition technology is based on the radar echo signal to extract the target feature and realize the automatic judgment of the target attribute, class, or type [1]. Using image recognition technology for target recognition is the most intuitive method in the field of automatic target recognition, so our research work is mainly focused on the 2-D radar image formed by the amplitude information obtained by imaging the target by SAR.

Target recognition based on 2-D SAR images has three main steps: image preprocessing, feature extraction, and classification decision. The design and selection of features directly determine the accuracy of target recognition. Eryildirim and Cetin [2] extracted features based on 2-D cepstrum with the aim of discriminating between clutter and man-made objects in a SAR image. Gaglione *et al.* [3] proposed a recognition algorithm for full-polarimetric SAR images, which is based on the pseudo-Zernike moments (pZm) and the Krogager decomposition components and exploited the multisource data offered by different sensors. Clemente *et al.* [4] proposed a novel algorithm for automatic target recognition that is capable of exploiting single or multichannel SAR images to extract features based on pZm for target recognition. The method for SAR target recognition based on dictionary learning and joint dynamic sparse representation was proposed by Sun *et al.* [5] and combined amplitude features and scale-invariant feature transform features in the recognition process. An algorithm for automatic target recognition based on Krawtchouk moments was proposed by Clemente *et al.* [6] and had high reliability in the presence of noise and reduced sensitivity to discretization errors.

Because the SAR imaging mechanism is different from the optical imaging system, different terrains in the two-dimensional images generated by the SAR imaging system exhibit several special phenomena, such as shadows, overlap, and perspective shrinkage. In addition, SAR images have coherent speckle noise, and the visual readability is poor. It is difficult to manually design effective features for SAR image target recognition. Different from the traditional automatic target recognition technology based on artificial design features [7], [8], deep neural networks, especially convolutional neural networks (CNNs), can automatically learn target features for automatic target recognition

Manuscript received July 25, 2020; revised October 18, 2020 and November 5, 2020; accepted November 13, 2020. Date of publication November 17, 2020; date of current version January 6, 2021. This work was supported by the National Natural Science Foundation of China under Grant 41871364, Grant 41871276, Grant 41871302, and Grant 41861048. *(Corresponding author: Guohua Wu.)*

Haifeng Li, Haikuo Huang, Li Chen, Jian Peng, Haozhe Huang, Zhenqi Cui, and Xiaoming Mei are with the School of Geosciences and Info-Physics, Central South University, Changsha 410083, China (e-mail: lihaifeng@csu.edu.cn; 969338839@qq.com; vchenlil@csu.edu.cn; PengJ2017@csu.edu.cn; hz_huang@csu.edu.cn; cuizhenqi@csu.edu.cn; meixiaoming17@163.com).

Guohua Wu is with the School of Traffic and Transportation Engineering, Central South University, Changsha 410083, China (e-mail: guohuawu@csu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2020.3038683

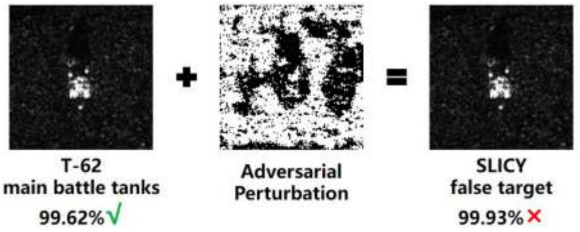


Fig. 1. On the left is the original SAR image. A CNN correctly identified it as a T-62 main battle tank with a confidence of 99.62%. On the right is the ASI. After adding an AP to the original SAR image, the CNN identified it as a Sandia Laboratory Implementation of Cylinders false target with a confidence of 99.93%.

[9], which reduces the computational cost and improves the recognition accuracy.

With the rapid development of deep learning in the field of computer vision, image classification and recognition methods based on CNN models have been widely used for target recognition in SAR images. For example, Shao *et al.* [10] analyzed and compared the performance of different CNNs on the MSTAR [11] dataset based on accuracy, number of parameters, training time, and other metrics to verify the superiority of CNNs for SAR image target recognition. Wang *et al.* [9] proposed despeckling and classification coupled CNNs to distinguish multiple categories of ground targets in SAR images with strong and varying speckle. Shang *et al.* [12] proposed deep memory convolution neural networks to alleviate the problem of overfitting caused by insufficient SAR image samples, and their method achieved higher accuracy than several other well-known SAR image classification algorithms. Huang *et al.* [13] proposed an improved deep Q-network method for polarimetric SAR image classification, and the experimental results demonstrated that the proposed method has a better classification performance than traditional supervised classification methods.

Although CNNs perform very well in the field of image classification and recognition, they are highly sensitive to adversarial perturbations (APs). APs can easily fool classifiers based on CNNs [14]–[18]. The purpose of adding APs to the original images is to fool a classifier into misclassifying them. We refer to such an image with an AP as an adversarial example (AE). In many cases, APs are difficult for humans to perceive, but they still cause CNNs to incorrectly predict the labels of AEs. The existence of AEs carries hidden dangers in the military and other fields with high security requirements. Fig. 1 shows that the adversarial SAR image (ASI) is misclassified with high confidence by the CNN, and humans can barely perceive any differences between the original SAR image and the ASI.

Previous research on AEs has shown that they can easily fool natural image classifiers [19]–[22]. However, there are few studies on AEs involving remote sensing fields, such as SAR, which limits our understanding of the security of remote sensing image classification models. Czaja *et al.* [23] were the first to analyze the problem of AEs in remote sensing images. The results showed that AEs can cause errors in even the most advanced remote sensing image classifiers. Li *et al.* [24] proposed a robust structure that can detect AEs in high-resolution remote sensing images. Nevertheless, we are unclear regarding the characteristics of ASIs at present. In this article, we mainly discuss the characteristics of AEs for SAR image classification models based on CNNs.

CNN classification models trained on SAR images are easily fooled by ASIs with special added noise. We can use attack algorithms to generate this special noise. It is special as follows: First, this noise is so small that it is almost imperceptible when added to the original image. Second, the pixels of the original image are modified to generate an adversarial image using gradient descent along the direction in which the CNN incorrectly predicts, thus easily fooling the CNN. Speckle noise is caused by the coherence principle on which SAR imaging is based. Speckle noise is very obvious in SAR images, and its randomness means it is not necessarily easy to fool the CNN into making the wrong classification.

We describe a possible problem in SAR image recognition, i.e., AEs lead to misclassification of CNNs. We discovered an interesting phenomenon of the AE by experiment. This phenomenon has not yet been studied, so we proposed a hypothesis for the occurrence of this phenomenon, and verified our conjecture through experiments. For the CNN trained on SAR images, we observed the common feature of the ASIs generated by different attack algorithms, i.e., ASIs generated by the nontarget attack algorithm have a preference for selecting attack classes. The research on AE in the field of remote sensing did not explain this fact, but we proposed a new metric, AE selective distance (AESD), to successfully explain the phenomenon of the attack selectivity of ASIs. Our research on this phenomenon will provide a theoretical basis for designing new attack methods.

The main contributions of our work are listed as follows.

1) We propose the AESD to analyze the attack selectivity of ASIs generated by different nontarget attack algorithms, which provided us with the opportunity to quickly achieve targeted attacks and understand the underlying geometry of ASIs.
2) Massive analyzing the AEs on MSTAR and SENSAR datasets with two attack model on six popular deep CNN models, we obtain several interesting conclusions.
3) The network structure of the CNN has a great influence on robustness. The simpler the structure is, the higher the robustness of the CNN.
4) For the training data, adding auxiliary information in the form of channels can improve the accuracy of the CNN, but it will reduce its robustness.
5) In addition, the use of cropping to remove unnecessary information for classification and recognition in the image can improve the robustness of CNNs against adversarial attacks.
6) Simply pursuing higher accuracy will lose robustness. We suggest that the designers should weigh the balance between accuracy and robustness when designing new CNNs.

The rest of this article is organized as follows. In Section II, we briefly review the related works on AEs. Section III introduces the principle of the attack algorithms and defines the

AESD. Section IV reports on the experimental results and provides an analysis. Finally, Section V provides a summary and conclusion.

## II. Related Works

In recent years, breakthroughs have been achieved in the application of CNNs to the field of remote sensing. Remote sensing automatic discrimination technology based on CNNs has been widely used in social life and military fields [25]–[29]. In the past, researchers have expended considerable effort in studying how to improve the accuracy of models by designing better model architectures, proposing more effective loss functions, and expanding datasets to improve data diversity. Before the discovery of AEs, the accuracy of CNNs was a consistent research concern. However, few people have questioned the safety and reliability of CNNs. Szegedy *et al.* [30] were the first to discover that CNNs are easily fooled by tiny perturbations. These perturbations cause CNNs to misclassify AEs with high confidence. Moreover, AEs generated by one neural network can fool other neural networks. These findings have caused widespread concern among researchers about the safety of deep learning. Many other later works also studied these interesting properties, but no complete theory yet exists to explain this phenomenon [31]–[34]. Goodfellow *et al.* [19] believe that the linear nature of neural networks in high-dimensional spaces leads to the generation of AEs. In a high-dimensional space, the infinitely small perturbations in AEs accumulate during the forward propagation of the network, which can cause large changes in the output and result in errors.

As CNNs are increasingly applied in practice, many researchers have begun to focus on the security of CNNs and have studied AEs in areas, such as autonomous driving [35] and face recognition [36], [37], and found that AEs reduce the robustness of CNNs. AEs exist not only in computer vision fields, such as image classification [20]–[22], [38]–[40], object detection [41], [42], and semantic segmentation [43], [44], but also in the natural language processing [45], [46] and speech recognition [47], [48].

Various attack algorithms exist for generating AEs. Taking image classification as an example, the attack algorithms can be divided into different types. Based on information that attackers can obtain, the attack algorithms can be divided into white-box attacks [19]–[22], [30], [38], [39] and black-box attacks [14], [40]. In a white-box attack, an attacker can obtain complete information about the target model, including its parameters, structure, training method, and even training data. In contrast, in a black-box attack, the attacker does not know specific information about the model but can observe the output by submitting various inputs. The attacker then uses the correspondence relationships between the input and output to find suitable AEs with which to attack the model. According to whether the attack class is directional, the attack algorithm can be divided into target attacks [19], [21], [30], [38], [40] and nontarget attacks [14], [20], [22], [39]. A target attack deception model incorrectly predicts the AEs as the specified labels while a nontarget attack needs only to cause the model to predict the labels of the AEs incorrectly.

With the rapid development of remote sensing technology, the amount of data we can obtain is constantly increasing, and it has become increasingly difficult to process massive amounts of data manually. Remote sensing image processing technology has gradually transformed from using traditional visual interpretation to relying on automated methods. Data-driven CNNs have made good progress in automatic remote sensing image processing. Currently, many CNN methods have been applied to automatic remote sensing image processing. The existence of AEs may have a serious impact on the practical application of these methods. SAR is widely used in the military field, and military applications have extremely high requirements for security. Nevertheless, we know nothing about ASIs at the moment. Therefore, we start with the classic attack algorithm to study the characteristics of ASIs and hope that our work can be helpful for future research.

## III. Methodology

In this section, we will introduce the methods used in the experiment in detail. First, the definition of ASIs is introduced. Next, the methods of generating ASIs will be introduced subsequently. Finally, the AESD we proposed will be introduced.

### A. Problem Description of ASIs

First, we formalize ASIs. Let $\boldsymbol{X}$ denote a SAR image dataset in which each SAR image is denoted as $\boldsymbol{x} \in \mathbb{R}^{C\times H\times W}$ and $\boldsymbol{k}$ represents a classification model that outputs a predicted label $\boldsymbol{k}(\boldsymbol{x})$ for each SAR image $\boldsymbol{x} \in \boldsymbol{X}$

$$\begin{aligned} k(\widetilde{x}) &\neq k(x) \quad \text{for most } x \in X \\ \widetilde{x} &= x + \eta \end{aligned} \tag{1}$$

where $\boldsymbol{x}$ is the original SAR image, $\widetilde{\boldsymbol{x}}$ is the ASI, and $\boldsymbol{\eta} \in \mathbb{R}^{C\times H\times W}$ is the AP, and $C$, $H$, and $W$ are the number of channels, height, and width of a SAR image, respectively. The classification model $\boldsymbol{k}$ misclassifies the ASI $\widetilde{\boldsymbol{x}}$ generated by the vast majority of original SAR images $\boldsymbol{x}$ from the SAR image dataset $\boldsymbol{X}$ after the perturbation $\boldsymbol{\eta}$ is added. We use the $\infty$ norm to constrain the perturbation. When the perturbation $\boldsymbol{\eta}$ is sufficiently small, humans cannot visually distinguish an ASI from an original SAR image

$$\|\eta\|_p = \left( \sum_{\substack{0\le i<H \\ 0\le j<W}} |\eta_{ij}|^p \right)^{1/p} < \delta \tag{2}$$

$$\xRightarrow{p=\infty}$$

$$\|\eta\|_\infty = \max_{\substack{0\le i<H \\ 0\le j<W}} |\eta_{ij}| \tag{3}$$

where $p$ is the norm. Here, $p = \infty$. $\delta$ controls the value of perturbation.

### B. Method for Generating ASIs

When training the CNN, the parameters in the CNN are updated by subtracting the gradient obtained by backpropagation so that the loss value becomes increasingly smaller, and the probability of the model prediction becomes increasingly higher. In a nontarget attack, the goal is that the model misclassifies the input image into any class other than the correct class. We need only to increase the loss value to achieve this goal. This is the opposite of the purpose of updating parameters when training the CNN.

Considering the speed and attack success rate of ASIs generated by the attack algorithm, we use two classic white-box attack algorithms, fast gradient sign method (FGSM) [19] and basic iteration method (BIM) [20]. FGSM is currently the attack algorithm most widely used to attack image classifiers based on CNNs because it needs only a one-step gradient-increasing operation to generate ASIs. Using FGSM to generate ASIs is very fast, but it also limits the attack success rate. We also use BIM, which has a higher attack success rate. BIM is an iterative algorithm that makes it easier to fool CNNs by generating ASIs through its multiple iterations, but it is slower than FGSM.

*1) Fast Gradient Sign Method:* Goodfellow *et al.* [19] proposed the FGSM to generate AEs. The FGSM attack algorithm is as follows:

$$\widetilde{x} = x + \epsilon\text{sign}(\nabla_x J(\theta, x, y)) \tag{4}$$

$$\nabla_x J(\theta, x, y) = -\sum_{k=1}^{c} y_k \log p_k \tag{5}$$

$$\text{sign}(x) = \begin{cases} -1, & x < 0 \\ 0, & x = 0 \\ 1, & x > 0 \end{cases} \tag{6}$$

where $\boldsymbol{x}$ is the original SAR image, $\widetilde{\boldsymbol{x}}$ is the ASI, and $J(\theta, x, y)$ is the loss function. Here, $J(\theta, x, y)$ is the multiclass cross-entropy, $\boldsymbol{\theta}$ is the model parameter, $c$ is the number of classes, $\boldsymbol{y}$ is the label of $\boldsymbol{x}$, and $y_k$ is the indicator variable. If the ground truth label is the $k$th class, $y_k$ is 1. Otherwise, $y_k$ is 0. $p_k$ is the predicted probability of the $k$th class obtained from the $\boldsymbol{x}$ input model, and $\boldsymbol{\nabla_x}$ is the partial derivative of $J(\theta, x, y)$ of $\boldsymbol{x}$. The sign function $\text{sign}(\cdot)$ retains only the gradient direction. It does not consider the specific gradient value. $\epsilon$ is a scalar value used to limit the value of perturbation.

The specific process of the FGSM algorithm is as follows: We input an original image $\boldsymbol{x}$ into the CNN to output a predicted probability through forward propagation, then calculate the loss value $J(\theta, x, y)$ between the predicted probability and the label $\boldsymbol{y}$ of the image, and finally use the lost value for backpropagation to obtain the gradient $\boldsymbol{\nabla_x} J(\theta, x, y)$ of the input image. To control the perturbation $\boldsymbol{\eta}$ without causing great damage to the original image, the norm limit is imposed on the perturbation. Therefore, instead of directly using the gradient value, only the gradient direction $\text{sign}(\cdot)$ is used, and a step-length is added to the gradient direction to obtain the perturbation. This step-length parameter $\epsilon$ can be used to control the amplitude of the attack noise. The larger the parameter value is, the greater the attack intensity and the easier it is to observe the noise. We add the perturbation $\boldsymbol{\eta}$ to the original image $\boldsymbol{x}$ to obtain the adversarial image $\widetilde{\boldsymbol{x}}$.

*2) Basic Iterative Method:* The FGSM is fast, but it uses only one gradient update, and sometimes one update is not enough to attack successfully. Thus, Kurakin *et al.* [20] proposed the BIM to generate AEs. The BIM attack algorithm is as follows:

$$x_0 = x, \quad x_{i+1} = \text{clip}(x_i + \alpha\,\text{sign}(\nabla_x J(\theta, x, y))) \tag{7}$$

where the initial ASI $\boldsymbol{x_0}$ is the original SAR image $\boldsymbol{x}$, $\boldsymbol{x_i}$ is the ASI at the $i$th iteration, $\text{clip}(\cdot)$ denotes that the value is limited to [0, 1], $\alpha$ is the attack step-length of each iteration, and the remaining symbols have the same meanings as in (4).

The specific process for the BIM to generate AEs is similar to FGSM. The difference is that BIM allocates the total noise amplitude in FGSM to each iteration. Given the total noise amplitude $\epsilon$, we use $\alpha = \epsilon/N$ to set $\alpha$ and $N$. In addition, $\text{clip}(\cdot)$ means that the overflowed value is replaced by the boundary value. This is because in the iterative update, as the number of iterations increases, some pixel values may overflow (outside the range of 0 to 1). Replacing these values with 0 or 1 will eventually generate an effective AE. BIM iteratively uses multiple small steps to create an attack, adjusts the direction after each small step, and adds a recalculated perturbation to achieve a better attack effect.

### C. AE Selectivity Distance

Perturbation generated by a nontarget attack algorithm causes an original sample to easily cross the nearest decision boundary to generate an AE. We call the distance between the sample and the nearest decision boundary the AESD. AESD provides a reasonable explanation for the attack selectivity of AEs generated by the nontarget attack algorithm.

*1) Geometric Interpretation:* Fig. 2 shows the AESD. The shapes represent the class that humans consider correct, whereas the colors represent the labels predicted by the classifier. For example, the original sample represented by a red sphere is perturbed to generate an AE represented by a blue sphere. From the human perspective, the AE is almost identical to the original sample. Thus, humans can still correctly classify the AE, i.e., humans still see the shape as the sphere, but the classifier will misclassify the AE, i.e., the color changes from red to blue.

Although the samples of class $A$ and class $B$ have the closest distance in the feature space, most of the AEs generated by the original samples belonging to class $A$ are misclassified as class $C$ instead of class $B$. We hold that the nontarget attack algorithm does not use the similarity of the features of the samples but generates the perturbation selectively according to the difficulty encountered for the sample to cross the decision boundary. Therefore, we propose AESD to verify the cause of the attack selectivity of AEs generated by nontarget attack algorithms. The choice of attack class is determined by the distance between the original sample and the decision boundary. The closer the distance is, the easier it is for the sample to cross the decision boundary and be misclassified as the class on the other side of the decision boundary.

*2) Definition of AESD:* In a multiclassification task, the distance from a sample to the decision boundary formed between different classes is different. These distance differences will

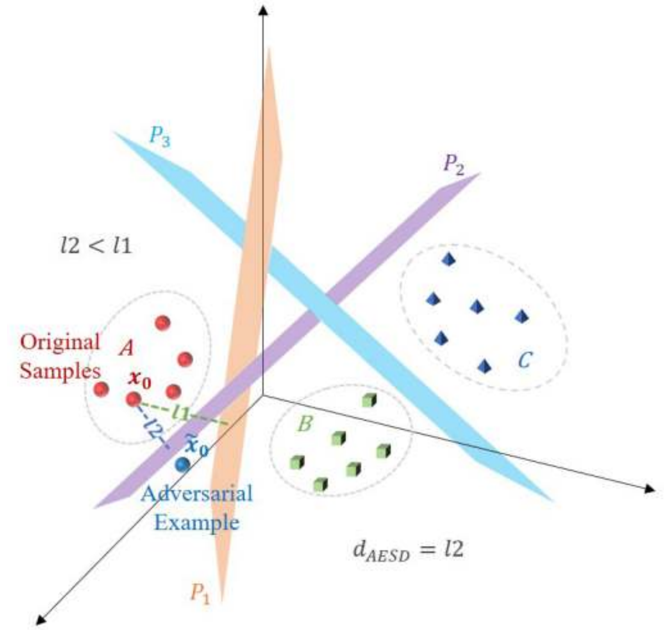


Fig. 2. Classes $A$, $B$, and $C$ are represented by a red sphere, a green cube, and a blue triangular pyramid, respectively, where the decision boundary between $A$ and $B$ is $P_1$, the decision boundary between $A$ and $C$ is $P_2$, and the decision boundary between $B$ and $C$ is $P_3$. The samples inside the ellipses are the original samples, and the samples outside the ellipses are the AEs. $\widetilde{\boldsymbol{x}_\mathbf{0}}$ is the AE produced by the original sample $\boldsymbol{x}_\mathbf{0}$. The distance from $\boldsymbol{x}_\mathbf{0}$ to the decision boundary $P_1$ is $l_1$, and the distance to the decision boundary $P_2$ is $l_2$. Because $l_2 < l_1$, $l_2$ is the AESD of $\boldsymbol{x}_\mathbf{0}$.

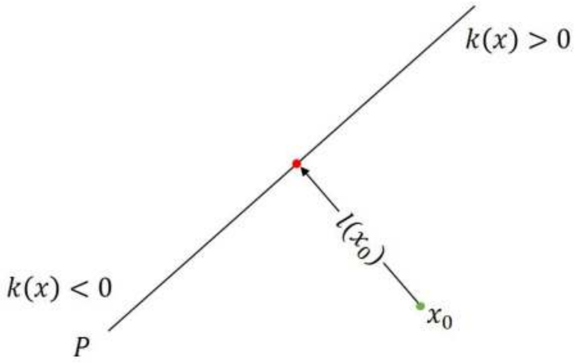


Fig. 3. Sample-boundary distance for a linear binary classifier.

inevitably lead to the attack selectivity of AEs. Therefore, we define AESD as follows:

$$d_{\text{AESD}} = \min_{1\leq i\leq n} l_i \tag{8}$$

where $l_i$ is the sample-boundary distance from the original sample to the $i$th decision boundary and $n$ is the number of decision boundaries in the model.

*3) Sample-Boundary Distance:*

*a) Binary classification:* The decision boundary of a classification model based on a CNN is extremely complicated. We cannot use mathematical formulas to accurately describe the decision boundary of a CNN. To simplify the problem, we first study linear binary classification. As shown in Fig. 3, we assume that the classification function is $k(x) = \text{sign}(f(x))$, where $f(x) = w^T x + b$, and $\boldsymbol{w}$ and $\boldsymbol{b}$ are parameters of $f(x)$. See (6), sign$(\cdot)$ is the sign function. $P : \{x : f(x) = 0\}$ represents the decision boundary. According to the distance formula from a point to a straight line, we can obtain the distance $l(x_0)$ formula from the sample $\boldsymbol{x}_\mathbf{0}$ to the decision boundary $P$ as follows:

$$l(x_0) = \frac{|f(x_0)|}{\|w\|_2}. \tag{9}$$

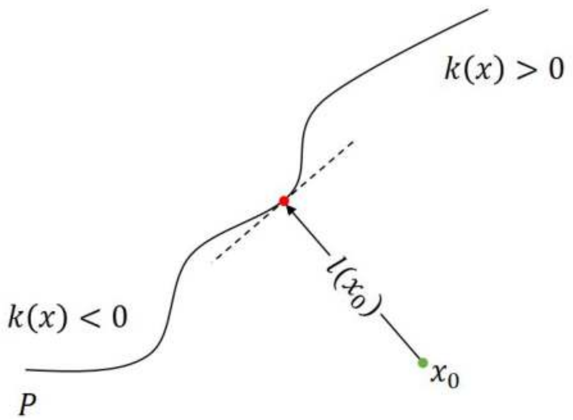


Fig. 4. Sample-boundary distance for a nonlinear binary classifier.

However, the CNN is by nature a nonlinear approximator, and its decision boundary is not a plane. For a nonlinear binary classification task, our assumption is that the decision functions of CNNs are all locally linearly approachable or that the classification boundaries are all smooth. If a Taylor expansion is performed at a certain point of the decision function, the tangent plane of this point is roughly similar to the decision boundary in a neighborhood of that point, as shown in Fig. 4. We linearize $f$ at $\boldsymbol{x}_\mathbf{0}$ and use $f(x_0)$ to derive the derivative of $\boldsymbol{x}_\mathbf{0}$ instead of the parameter $\boldsymbol{w}$, i.e., $\boldsymbol{w} = \boldsymbol{\nabla}_{\boldsymbol{x}} f(x_0)$ approximates the distance between the sample $\boldsymbol{x}_\mathbf{0}$ and the decision boundary $P$ as follows:

$$l(x_0) = \frac{|f(x_0)|}{\|\nabla_x f(x_0)\|_2}. \tag{10}$$

*b) Multiclassification:* Next, we extend the discussion from binary classification to multiclassification. We assume that a classifier has $n$ outputs, where $n > 2$ is the number of classes. In Fig. 5, $n = 3$, and $c1$, $c2$, and $c3$ are indexes of three different classes, where the class of sample $\boldsymbol{x}_\mathbf{0}$ is the $c2$th class. The decision boundary between the $c1$th class and the $c2$th class is $P : \{x : f_{c1}(x) - f_{c2}(x) = 0\}$. The distance from $\boldsymbol{x}_\mathbf{0}$ to the boundary $P$ formed by these two classes is $l(x_0)$ as follows:

$$l(x_0) = \frac{|f_{c2}(x_0) - f_{c1}(x_0)|}{\|\nabla_x f_{c2}(x_0) - \nabla_x f_{c1}(x_0)\|_2} \tag{11}$$

where $\boldsymbol{f(x)}$ is the predicted vector output by the model. $f_c(x)$ is the value of the $c$th dimension of the vector $\boldsymbol{f(x)}$, i.e., the model predicts the sample as the output value of the $c$th class. $\boldsymbol{\nabla_x}$ is the partial derivative of the predicted vector $\boldsymbol{f(x)}$ of sample $\boldsymbol{x}$.

Fig. 6 shows the main process of analyzing ASIs. We divided the SAR dataset into a training dataset and a test dataset. The CNN is trained on the training dataset, and the accuracy of the model is calculated using the test dataset. The misclassification

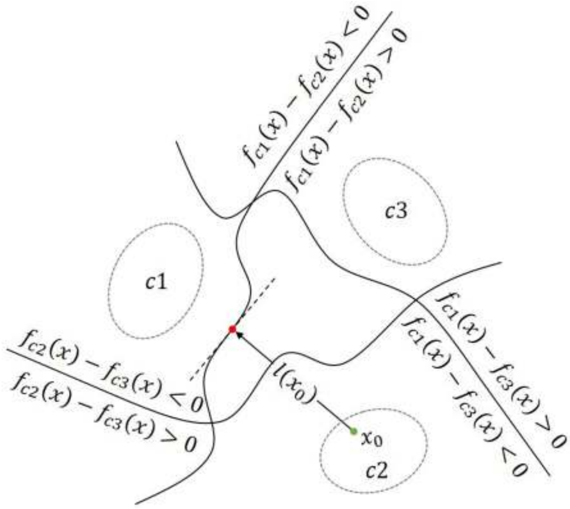

$f_{c1}(x) - f_{c2}(x) < 0$
$f_{c1}(x) - f_{c2}(x) > 0$
$c3$
$c1$
$f_{c1}(x) - f_{c3}(x) > 0$
$f_{c1}(x) - f_{c3}(x) < 0$
$f_{c2}(x) - f_{c3}(x) < 0$
$f_{c2}(x) - f_{c3}(x) > 0$
$l(x_0)$
$x_0$
$c2$

Fig. 5. Sample-boundary distance for a nonlinear multiclassifier.

of the model itself affects our calculation of the attack success rate of ASIs. Therefore, for the test dataset, we set a confidence threshold to filter out the test data that the model can correctly classify. In our experiments, the threshold is 0.7. The confidence of the test image is less than the threshold, which is considered to be an unreliable false prediction. We use the filtered test images as the original images to generate the adversarial images by two attack algorithms, FGSM and BIM. The correctness of AESD is verified by calculating the AESD of original images and comparing whether the class of AESD is consistent with the predicted label of the adversarial images. The attack success rate is used to evaluate the vulnerability of CNNs and show the impact of the adversarial image parameters, such as image size and number of channels.

## IV. Experiments and Analysis

To analyze the characteristics of ASIs, we used two attack algorithms, FGSM and BIM, to attack six CNNs trained on MSTAR and SENSAR datasets. The two datasets used in our experiment are the amplitude information retained by the SAR data after processing, i.e., the single-band grayscale image. The CNN models were VGG16 [49], GoogLeNet [50], InceptionV3 [51], ResNet50 [52], ResNeXt50 [53], and DenseNet121 [54]. At the end, the experimental results are analyzed comprehensively.

### A. Databases

*1) MSTAR:* MSTAR [11] was produced by the US Defense Advanced Research Projects Agency in the mid-1990s using high-resolution spotlight SAR to collect SAR images of various Soviet military vehicles. The collection conditions for the MSTAR images are divided into two types: standard operating condition (SOC) and extended operating condition (EOC). In this study, we use SAR images collected by SOC. The dataset includes ten ground target classes, and the classes have different sizes in different pictures. The targets are located in the center of the images and occupy only a small area. To simplify identification, we center-crop the image to $128 \times 128$. The training dataset was collected at a $17^\circ$ imaging side view, and the test dataset was collected at a $15^\circ$ imaging side view. Detailed information regarding the dataset is shown in Table I. Fig. 7 shows examples of SAR images for each of the classes in MSTAR.

TABLE I
Details of MSTAR, Including Target Class, Data Number, and Class Name

| Target Class | Train Number ($17^\circ$) | Test Number ($15^\circ$) | Class Name |
|---|---|---|---|
| 2S1 | 299 | 274 | Self-propelled howitzer |
| BMP2 | 233 | 196 | Infantry fighting vehicle |
| BRDM2 | 298 | 274 | Armored reconnaissance vehicle |
| BTR60 | 256 | 195 | Wheeled armored transport vehicle |
| BTR70 | 233 | 196 | Wheeled armored transport vehicle |
| D7 | 299 | 274 | Bulldozer |
| T62 | 299 | 273 | Main battle tanks |
| T72 | 232 | 196 | Main battle tanks |
| ZIL131 | 299 | 274 | Cargo truck |
| ZSU234 | 299 | 274 | Self-propelled artillery |

*2) SENSAR:* SEN1-2 [55] is a SAR-optical image-pair dataset divided into four different subsets according to the four seasons: spring, summer, autumn, and winter (ROIs1158 spring, ROIs1868 summer, ROIs1970 fall, and ROIs2017 winter). Each subset contains pairs of SAR-optical images taken at different locations in the Northern Hemisphere. In this study, we use only the ROIs1868 summer subset, which is divided into two folders, s1 and s2, where s1 holds the SAR images and s2 holds the optical images. In our experiments, we use only the SAR images. Folder s1 contains 49 subfolders, and each subfolder corresponds to the SAR images taken of the same area, i.e., the SAR images in a given folder belong to the same area and class, and each area's class is represented by a serial number (e.g., 0, 1, 2, ...). We selected 20 of the 49 folders and randomly selected SAR images from each folder according to a training-to-testing ratio of 1: 1 to construct a new dataset called SENSAR, which contains 10 581 training samples and 10 575 test samples. The SAR images in SENSAR have a size of $256 \times 256$. Fig. 8 shows examples of SAR images for each class in SENSAR.

### B. Metrics and Implementation Details

To evaluate the vulnerability of CNNs, we used the evaluation indicator attack success rate. In the attack task, the higher the attack success rate value is, the more fragile the CNN. It is expressed as follows:

$$\text{attack success rate} = \frac{N_{\text{diff}}}{N_{\text{all}}} \tag{12}$$

where $N_{\text{diff}}$ is the number of ASIs whose predicted labels differ from the true classes and $N_{\text{all}}$ is the total number of ASIs generated by the attack algorithm against the classification model.

In general, the pixel value range of an 8-b image is between 0 and 255, and the pixel value range of a 16-b image is between 0 and 65535. To make the loss function converge as quickly as possible in the process of model training, the image is usually

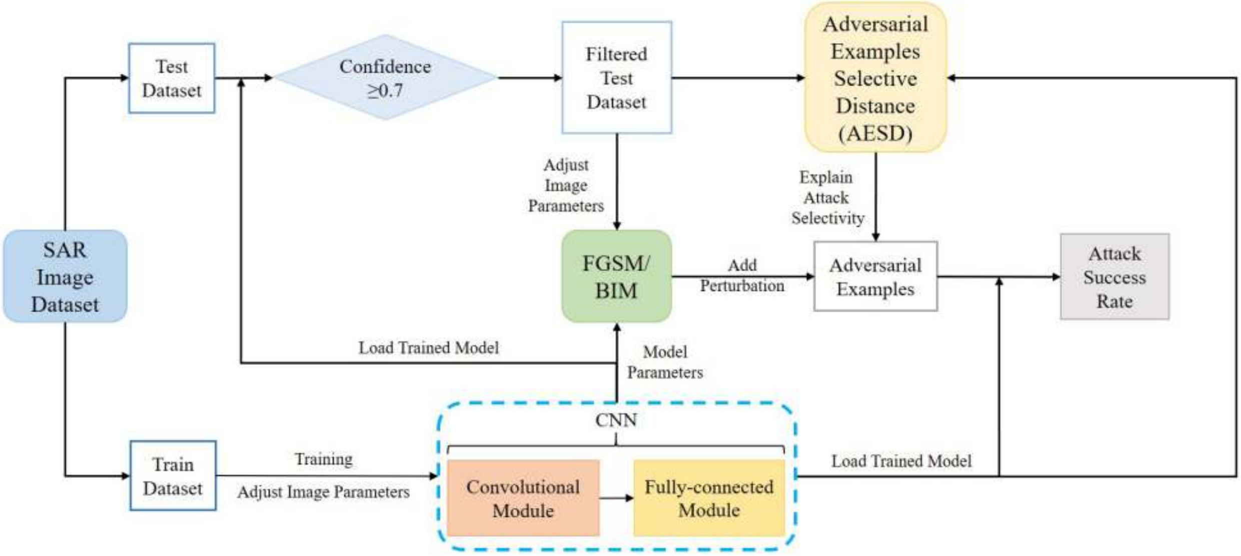


Fig. 6. Process for analyzing the characteristics of ASIs.

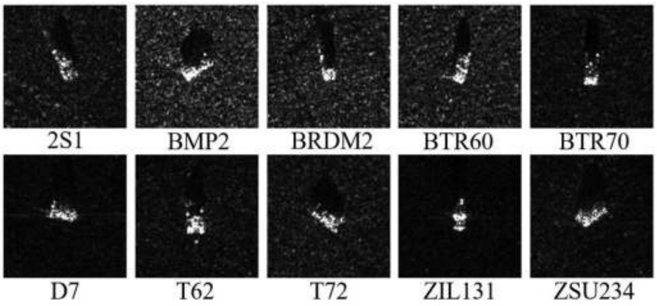


Fig. 7. Examples of SAR images in MSTAR.

normalized, i.e., the pixel value of the image is input to the model as a 0–1 floating point number. In the case of 8-b images, to save an AE (0–1 floating point number) as an image (0–255 unsigned integer), the decimal places are rounded off. If $\epsilon = 0.01$, $2 < 255 \times 0.01 = 2.55 < 3$, the upper limit of the pixel value that each pixel in the image can modify is 3. The greater the value of $\epsilon$, the more obvious the noise in the adversarial image is [20]. To make the original image and the adversarial image have no obvious difference, the value of $\epsilon$ should be set as small as possible. However, if $\epsilon$ is overly small, such as a pixel value that can be modified by most $\pm 1$ for each pixel, it is difficult to damage the classification of the CNN with such an adversarial image.

Therefore, for FGSM, we set $\epsilon$ to 0.01, which not only ensures that we cannot perceive the difference between the adversarial image and the original image but also enables the confrontation image to successfully attack the CNN. BIM is an iterative attack algorithm, and FGSM is a single-step attack algorithm. To compare BIM and FGSM, variable control should be performed. For BIM, $\alpha$ was set to 0.002, and the number of iterations $N$ was set to 5. Let $\epsilon = \alpha \times N$, to ensure that the upper limit of the pixel value of each pixel modified after the iterative attack of BIM is the same as the single-step attack of FGSM.

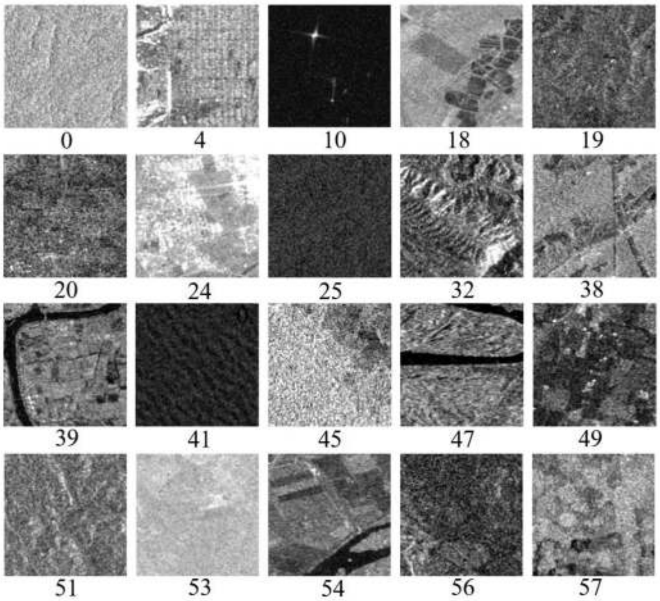


Fig. 8. Examples of SAR images in SENSAR.

### C. Vulnerability of CNNs

Some examples of the attack results are shown in Fig. 9. In Fig. 9(a), the class of the original SAR image is BTR70, and VGG16 correctly classified it with a confidence above 99%. The first column shows the original image and its corresponding predicted label and confidence. The third column shows the adversarial image and its corresponding predicted label and confidence. The second column shows the APs, and the perturbation image is obtained by subtracting the original image from the adversarial image. We set $\epsilon$ to 0.01, so the maximum value of pixels in the perturbation image is 3. We cannot observe such small pixel changes, so the second column shows the image after the pixel value of perturbation is enlarged.

TABLE II
Attack Success Rates of ASIs

| SAR Dataset | Attack Algorithm | VGG16 | GoogLeNet | InceptionV3 | ResNet50 | ResNeXt50 | DenseNet121 |
|---|---|---|---|---|---|---|---|
| MSTAR | FGSM | 68.25% | 70.03% | 72.81% | 71.63% | 76.14% | 91.36% |
| | BIM | 78.41% | 84.97% | 99.28% | 90.96% | 96.62% | 99.20% |
| SENSAR | FGSM | 72.17% | 87.92% | 72.65% | 90.35% | 85.98% | 91.01% |
| | BIM | 90.83% | 99.52% | 97.94% | 96.94% | 99.96% | 99.36% |

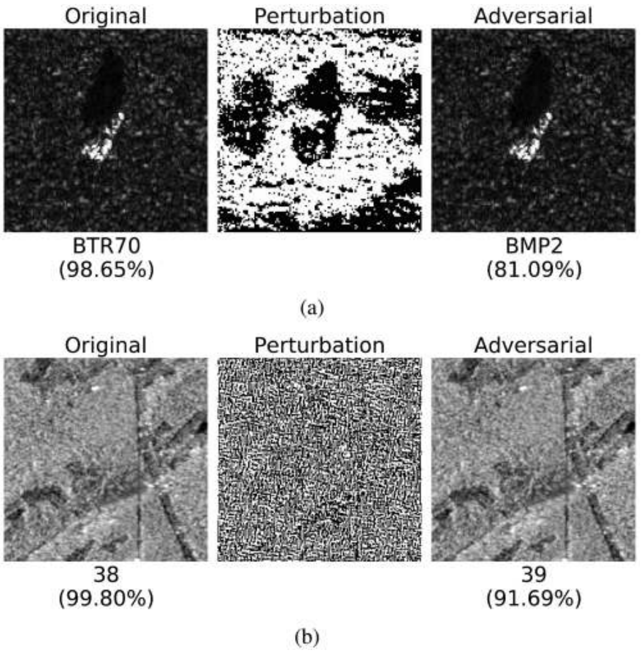


Fig. 9. Examples of adversarial images generated by the FGSM attacks using VGG16. (a) MSTAR. (b) SENSAR.

However, VGG16 not only incorrectly classified the adversarial image as BMP2 but also provided a confidence of over 81%. “Ori.” and “Adv.” in the chart represent the original image and the adversarial image, respectively. We cannot perceive the difference between the adversarial image and the original image, but we can already clearly see the difference in the feature map extracted by the CNN. As shown in Fig. 10, we use a heat map to visualize these features. The difference between the feature maps at the corresponding positions framed by the red frame is very obvious, which shows that the disturbances that human eyes cannot perceive affect the features extracted by the CNN. Fig. 11 shows the feature vectors of the original image and the adversarial image for classification, which ultimately leads to the result of Fig. 12. The class BTR70 has the largest predicted value in the original image. After adding the perturbation, the predicted value of class BTR70 drops significantly, and the value corresponding to class BMP2 becomes the largest predicted value.

Table II shows the attack success rate of ASIs generated by FGSM and BIM against six CNNs trained on MSTAR and SENSAR. The results show that the average attack success rate of ASIs is 86.85%, the highest is 99.96%, and the lowest is 68.25%. Most original SAR images generate ASIs after specific perturbations are added, proving that ASIs are quite effective in fooling CNNs trained on SAR images, as indicated by the obtained high attack success rate, which means that there are many ASIs in high-dimensional space. Table III shows the average confidence of ASIs. In Table III, the maximum value of the average confidence on ASIs is 0.9997, and the minimum value is 0.8060. After being attacked, the CNNs revealed their vulnerabilities by misclassifying the ASIs with high confidence, which means that CNNs trained on SAR images are highly vulnerable to ASIs.

In addition, different CNNs that were attacked by ASIs showed different vulnerabilities. We hold that model structure differences are the main reason for the differences in the vulnerability of CNNs. The more modules are stacked or aggregated in the model structure, the more vulnerable the CNNs.

Both GoogLeNet and InceptionV3 include the inception structure, which stacks the feature maps using multiple small convolution kernels and increases the width of the network by increasing the number of channels in the feature map. Both ResNet and DenseNet contain the connection structure that propagates low-layer information through identity connections and residual connections to increase the network depth. ResNeXt uses a multibranched isomorphic structure in which the number of branches is called the “cardinality” and forms another key factor for measuring neural networks in addition to depth and width. Although these structures improve model classification accuracy, they are also more likely to accumulate small perturbations in the lower layers that are amplified in the higher layers and, thus, have a substantial impact on the final output.

From Table II, the other five CNNs have significantly higher attack success rates than does VGG16, especially DenseNet121. The attack success rate of DenseNet121 is also higher than that of other CNNs. VGG16 has the simplest structure and includes no complex modules. Consequently, VGG16 has the lowest attack success rate by ASIs. DenseNet121 has a very dense connection structure that directly connects all the layers. The input to each layer consists of the feature maps of all the preceding layers. Therefore, the small perturbations in ASIs are accumulated and enlarged through the feature map aggregation, causing DenseNet121 to be more vulnerable to attacks.

### *D. Attack Selectivity of ASIs*

By collecting statistics on the predicted labels of ASIs, we found that the predicted label distribution of ASIs is highly concentrated. Fig. 13 shows the distribution and the proportion of the predicted labels of ASIs generated by the FGSM attack on GoogLeNet on MSTAR. The predicted labels of the ASIs concentrate primarily on a few classes and most of the classes only distributed a small number of ASIs. Fig. 13(a) shows the nature of the fourth- and fifth-order truncation. We will call it a

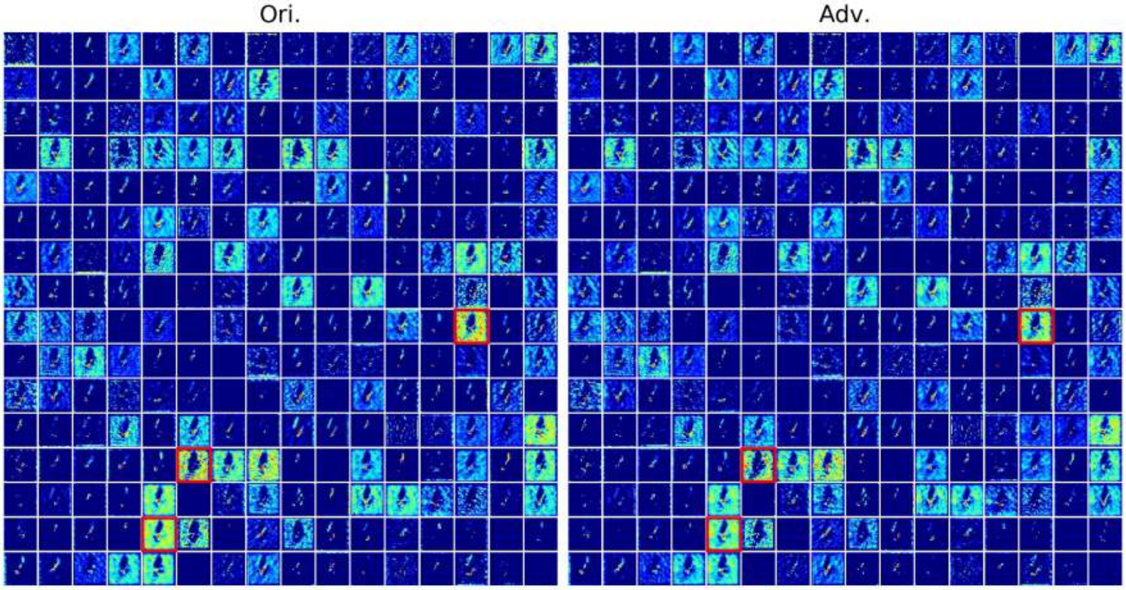


Fig. 10. The middle-layer features of the SAR image in Fig. 9(a) extracted by VGG16.

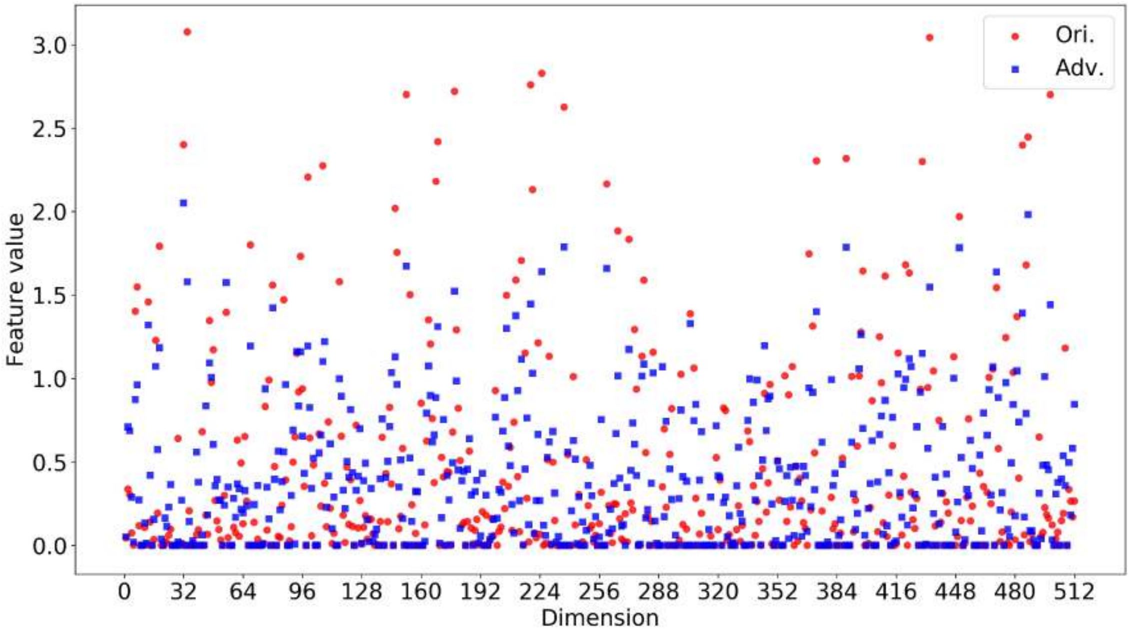


Fig. 11. The SAR image in Fig. 9(a) outputs a 512-D feature vector through the convolutional layer of VGG16.

$k$th-order truncation when it is larger than a certain constant $k$. Fig. 13(b) shows that these 4–5 classes account for more than 90% of all the predicted classes, indicating that ASIs are more likely to be misidentified as these 4–5 classes. For example, the ASIs generated by the original SAR images of class T72 were classified as class T62, class ZIL131, and class BMP2, comprising 38.13%, 34.53%, and 22.30%, respectively, whereas the remaining classes accounted for only 5.04%.

The aggregation of the predicted labels of ASIs is related to the distribution of the original SAR images in the feature space, and it is also affected by the attack selectivity of the ASIs. In general, the distribution of the original SAR images belonging to the same class is concentrated in the feature space, and there is always a decision boundary closest to these images. The attack selectivity makes an original image tend to choose to cross the closest decision boundary to generate an AE. Consequently, the predicted labels of ASIs generated by the original SAR images of the same class are highly concentrated. In Fig. 2, the distribution of the three classes is relatively concentrated. Most samples of class $A$ are closer to the classification boundary $P_2$. Since the AEs generated by the nontarget attack algorithm have attack selectivity, the predicted labels of AEs generated by the original samples of class $A$ are highly concentrated, and most of the predicted labels are class $C$.

TABLE III
AVERAGE CONFIDENCE OF ASIs

| SAR Dataset | Attack Algorithm | VGG16 | GoogLeNet | InceptionV3 | ResNet50 | ResNeXt50 | DenseNet121 |
|---|---|---|---|---|---|---|---|
| MSTAR | FGSM | 0.9377 | 0.8936 | 0.9694 | 0.9068 | 0.8980 | 0.8803 |
| | BIM | 0.9647 | 0.9746 | 0.9997 | 0.9847 | 0.9919 | 0.9950 |
| SENSAR | FGSM | 0.8060 | 0.9023 | 0.9084 | 0.9547 | 0.9506 | 0.9122 |
| | BIM | 0.9577 | 0.9877 | 0.9963 | 0.9907 | 0.9947 | 0.9887 |

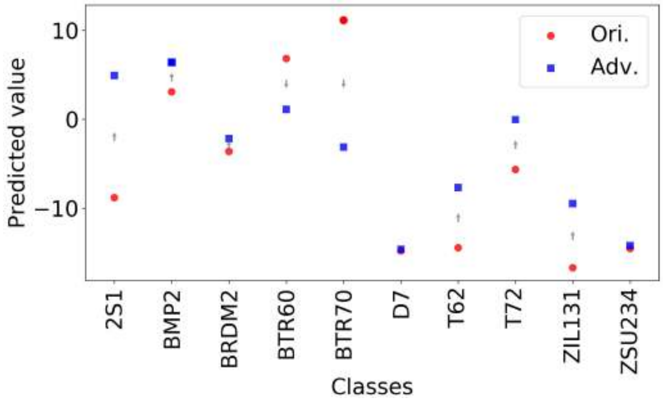


Fig. 12. The feature vector in Fig. 11 outputs a 10-D prediction vector through the fully connected layer of VGG16, and each dimension represents a class.

Fig. 14 shows the sample-boundary distance of the SAR images. In Fig. 13, GoogLeNet classified the ASI generated by the SAR original image of class BTR70 as class 2S1. Class 2S1 corresponds to the class with the maximum proportion in class BTR70 of Fig. 13(b), so we randomly select an image from the original images of class RRT70 to generate an AE, and this AE is most likely to be misclassified as class 2S1. The class calculated from the AESD of this original image is also class 2S1, which corresponds to the class of the shortest bar in Fig. 14 and is the same as the class most likely to be attacked. The result illustrates that AESD can explain the attack selectivity of ASIs.

To verify the reliability of AESD, we counted the proportion of the predicted labels of ASIs that matched the classes indicated by AESD, i.e., the accuracy of AESD, as shown in Table IV. Min-Dist-1 is the proportion of the predicted labels of ASIs that are the same as the classes pointed by AESD. Because the AESD is obtained based on the sample-boundary distance and the sample-boundary distance is calculated approximately, we also count Min-Dist-3, which is the proportion of the predicted labels of ASIs that are the same as one of the classes represented by the smallest three sample-boundary distances. From Table IV, on MSTAR, the average accuracy of Min-Dist-1 for the six CNNs is 81.84%, and the average accuracy of Min-Dist-3 is 97.35%. On SENSAR, the average accuracy of Min-Dist-1 is 68.60%, and the average accuracy of Min-Dist-3 is 89.88%. These results show that AESD is reliable in explaining the attack selectivity of ASIs.

TABLE IV
ACCURACY OF AESD

| SAR Dataset | Model Name | Attack Algorithm | Min-Dist-1 | Min-Dist-3 |
|---|---|---|---|---|
| MSTAR | VGG16 | FGSM | 80.25% | 96.91% |
| | | BIM | 80.11% | 96.71% |
| | GoogLeNet | FGSM | 89.75% | 98.95% |
| | | BIM | 84.44% | 98.43% |
| | InceptionV3 | FGSM | 87.58% | 98.86% |
| | | BIM | 78.37% | 96.86% |
| | ResNet50 | FGSM | 84.54% | 98.51% |
| | | BIM | 77.91% | 96.94% |
| | ResNeXt50 | FMSM | 86.27% | 98.69% |
| | | BIM | 76.45% | 95.05% |
| | DenseNet121 | FGSM | 78.25% | 96.27% |
| | | BIM | 78.17% | 96.00% |
| SENSAR | VGG16 | FGSM | 71.81% | 93.24% |
| | | BIM | 68.06% | 88.74% |
| | GoogLeNet | FGSM | 67.08% | 89.61% |
| | | BIM | 63.03% | 86.97% |
| | InceptionV3 | FGSM | 76.67% | 95.87% |
| | | BIM | 66.31% | 89.95% |
| | ResNet50 | FGSM | 71.90% | 91.32% |
| | | BIM | 64.86% | 86.80% |
| | ResNeXt50 | FMSM | 70.50% | 91.13% |
| | | BIM | 59.44% | 79.95% |
| | DenseNet121 | FGSM | 74.40% | 93.94% |
| | | BIM | 69.15% | 91.05% |

In Table IV, the accuracy of AESD when using BIM is generally lower than the accuracy of AESD when using FGSM. We consider that a reasonable and possible reason is that BIM is an iterative algorithm and it recalculates the perturbation in each iteration. The ASIs may continuously cross different decision boundaries during the iterative process. However, the AESD we proposed is calculated by the finally generated ASIs. Thus, our method ignores the iterative process, which results in the predicted labels of ASIs generated by the iterative attack algorithm matching the AESD with a lower accuracy rate than those of the single-step attack algorithm.

## E. Parameter Sensitivity Analysis

By adjusting the hyperparameters of CNNs, the model accuracies can be improved to a certain extent. From the perspective of hyperparameters, we maintain that changing the image-related hyperparameters should also have an impact on the attack success rate of ASIs. Therefore, we used MSTAR to analyze the sensitivity of the attack success rate of ASIs to changes in the image parameters (size and number of channels).

*1) Sensitivity Analysis of the Number of Channels:* To analyze the effect of the number of channels on the attack success rate of ASIs, we trained CNNs using SAR images read in both RGB and GRAY modes. The RGB mode makes two copies of the single-channel image in the channel direction to form a three-channel image.

A CNN is a black-box model, which is different from traditional machine learning models that require artificial design features, such as color and texture. A CNN can learn a pattern from data to extract the features it can distinguish, and these

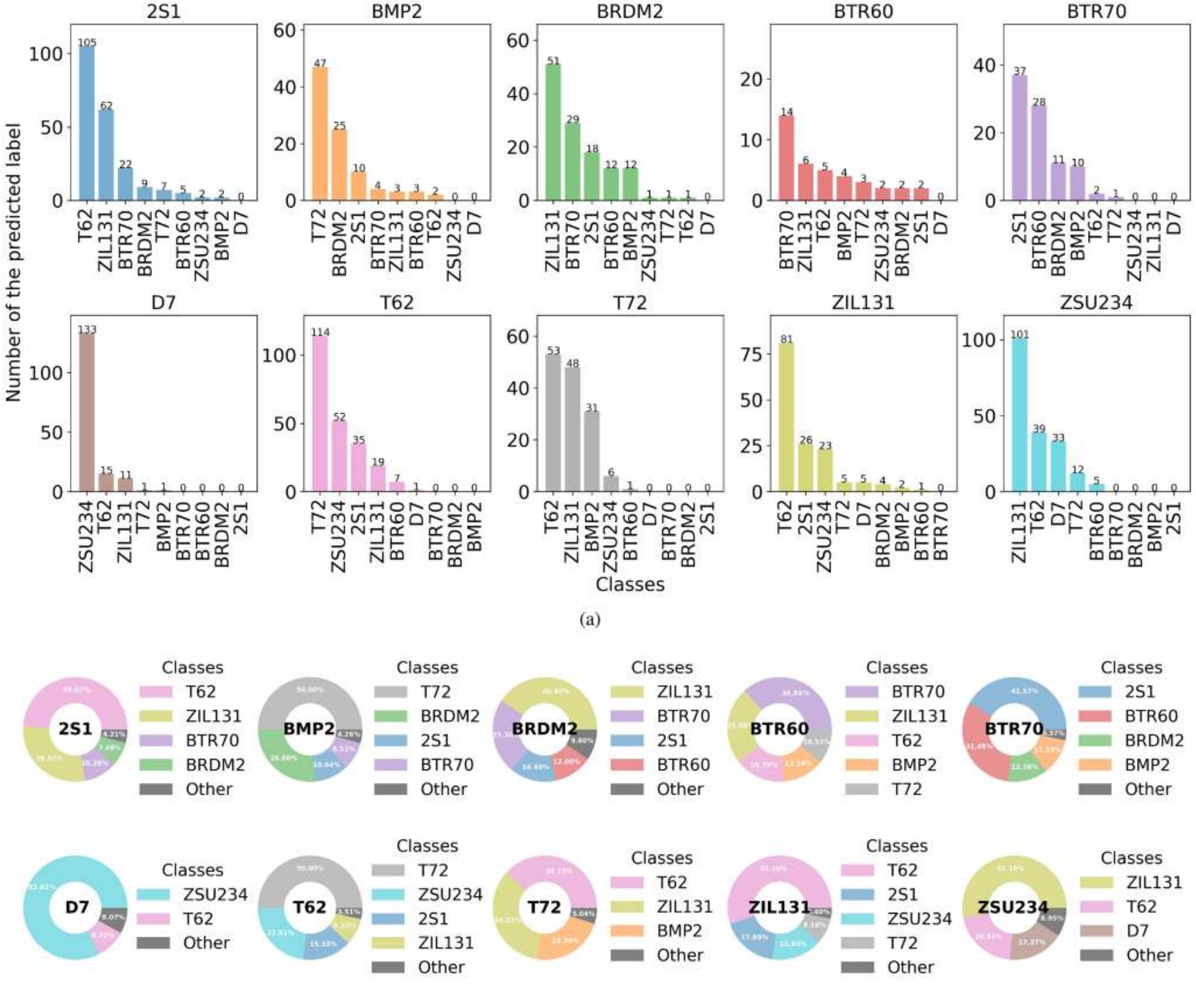


Fig. 13. Distribution and the proportion of the predicted labels of ASIs generated by FGSM attack GoogLeNet on MSTAR. (a) Distribution of the predicted labels of ASIs. The titles over the bar graphs indicate the classes to which the original SAR images belong. (b) Proportion of the predicted labels of ASIs. The tags in the centers of the ring charts indicate the class to which the original SAR images belong.

features are incomprehensible to humans. Regardless of whether it is applied to an optical image or a SAR image, a CNN is based on the pattern learned in the data and, thus, has the ability to correctly classify images. Because the principle of SAR imaging is different from that of optical sensors, there is a considerable amount of noise and artifacts in SAR images. These are the characteristics that optical images do not have, and a CNN designed in combination with these characteristics has higher accuracy in recognizing SAR images. In our experiment, we did not use CNNs specially designed for SAR images, and only common CNNs were used. For the recognition of a common CNN, there is no obvious difference between optical images and SAR images. Moreover, the grayscale information of the optical images has a great correlation with the amplitude information of the SAR images, so the adversarial properties of the SAR images and the optical images are similar on the digital level.

TABLE V
CLASSIFICATION ACCURACY OF CNNS TRAINED WITH RGB AND GRAY MODES AND THE ATTACK SUCCESS RATES OF ASIS GENERATED BY THE RGB MODE AND GRAY MODE

| Model Name | Ori. | | Adv. | | | |
|---|---|---|---|---|---|---|
| | | | FGSM | | BIM | |
| | RGB | GRAY | RGB | GRAY | RGB | GRAY |
| VGG16 | 96.46% | 96.83% | 69.69% | 68.25% | 84.63% | 78.41% |
| GoogLeNet | 99.55% | 99.51% | 82.51% | 70.03% | 91.88% | 84.97% |
| InceptionV3 | 98.15% | 98.27% | 89.90% | 72.81% | 100.00% | 99.28% |
| ResNet50 | 96.41% | 97.49% | 85.32% | 71.63% | 97.53% | 90.96% |
| ResNeXt50 | 96.91% | 97.57% | 78.51% | 76.14% | 98.62% | 96.62% |
| DenseNet121 | 98.60% | 98.89% | 99.41% | 91.36% | 100.00% | 99.20% |

However, the CNN classification model trained on optical images is more likely to be fooled against the adversarial images than the CNN classification model trained on SAR images. Increasing the number of channels of SAR images improves the attack success rate of ASIs. From Table V, using FGSM and

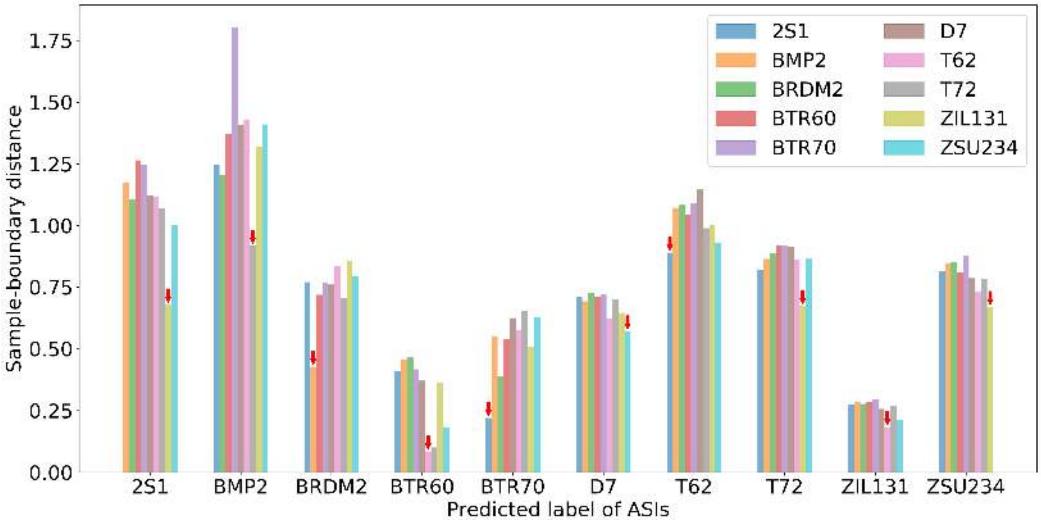


Fig. 14. Sample-boundary distance of the SAR images. We selected one SAR image randomly from each class of the test dataset and calculated the sample-boundary distances of these images. The class of the color of the bar indicated by the red arrow in the figure indicates the predicted label of ASI, and the class of the color of the shortest bar is the class indicated by AESD.

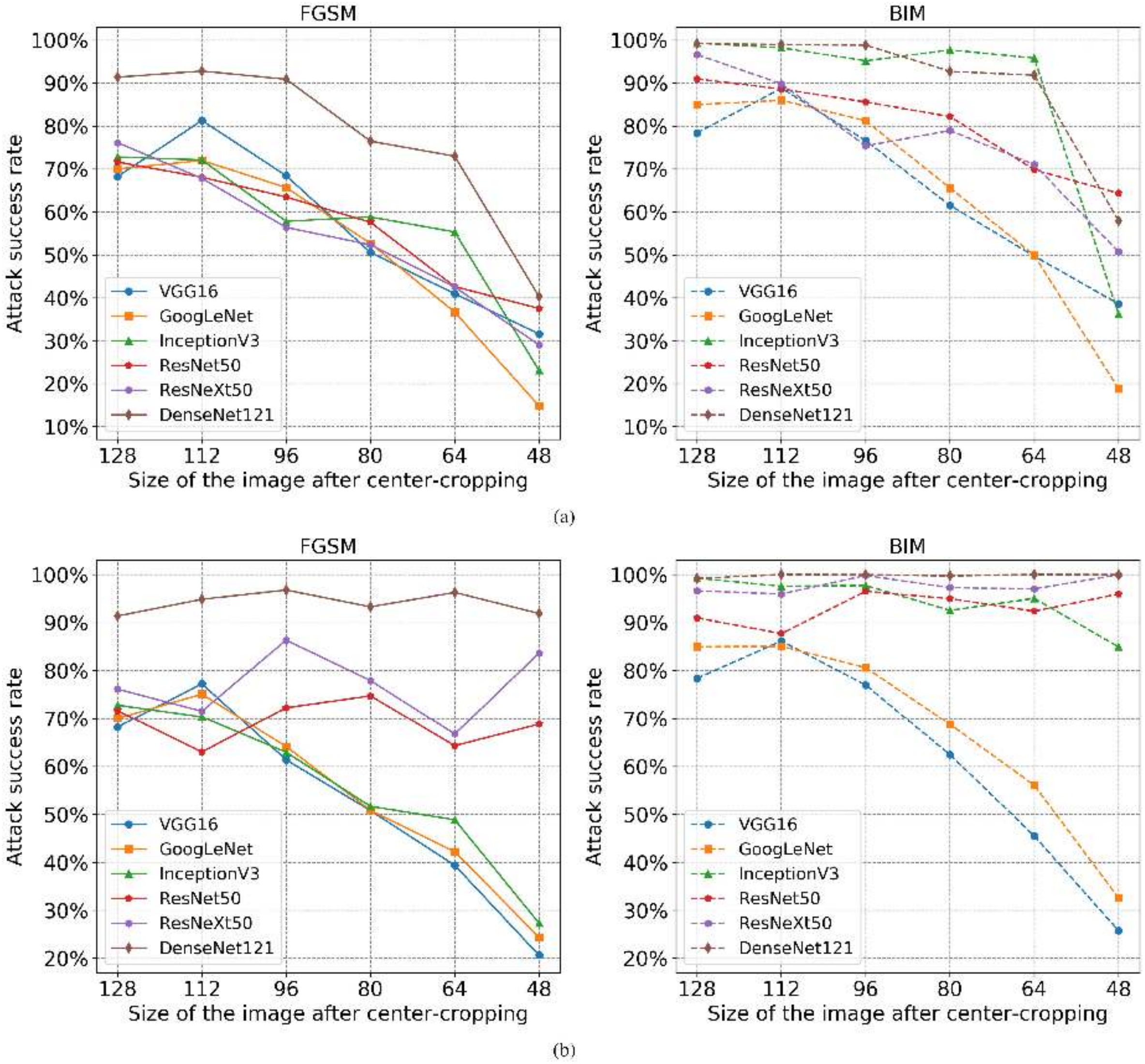


Fig. 15. Attack success rate trends of the ASIs as the input image size changes. (a) Image was not processed after center-cropping the image. (b) Image was resized to 128 × 128 after center-cropping the image.

BIM, the attack success rates of the three-channel ASIs of all CNNs are higher than those of the single-channel ASIs. Compared with SAR images, optical images contain more bands, and the AEs of optical images can carry more adversarial information and are more likely to fool a CNN from misclassification.

*2) Sensitivity Analysis of the Size:* To analyze the impact of image size on the attack success rate, we cropped the MSTAR images to six different sizes: $128 \times 128$, $112 \times 112$, $96 \times 96$, $80 \times 80$, $64 \times 64$, and $48 \times 48$. Then, we used two methods in the experimental analysis: one performed no processing on the center-cropped image, and the other resized the center-cropped image to $128 \times 128$. Fig. 15 shows that decreasing the image size can reduce the attack success rate of ASIs, and different attack algorithms and model structures have different sensitivities to the sizes of ASIs.

For VGG16 and GoogLeNet, ASIs generated by FGSM and BIM reduce the attack success rate when the image size is reduced, and the attack success rate still maintains a significant downward trend even after being zoomed to the same size. These results show that VGG16 and GoogLeNet are sensitive to changes in the size of ASIs and that reducing the sizes of ASIs is not the direct cause of the decline in the attack success rate. Instead, other factors regarding ASIs lead to the decrease.

For InceptionV3, the attack success rate of ASIs generated by FGSM decreases as the image size decreases, and they still maintain a significant downward trend even after being zoomed to the same size. The attack success rate of ASIs generated by BIM remains basically flat. These results show that InceptionV3 is sensitive to the size of ASIs generated by FGSM. The reduction in image size also affects other factors that indirectly reduce the attack success rate. The attack success is not sensitive to the size of ASIs generated by BIM, and reducing the image size does not cause a significant decrease in the attack success rate.

For ResNet50, ResNeXt50, and DenseNet121, the attack success rates of ASIs generated by FGSM also decrease as the image size decreases. After the images are zoomed to the same size, the attack success rate does not decrease; instead, it fluctuates, whereas the attack success rate of ASIs generated by BIM remains basically unchanged. The results show that ResNet50, ResNeXt50, and DenseNet121 are sensitive to the sizes of ASIs generated by FGSM, and the reduction in image size leads directly to a decrease in the attack success rate. They are not sensitive to variations in the sizes of ASIs generated by BIM.

In our experiments, SAR images read in RGB mode are three copies of single-band images in the channel direction, which does not increase the effective information in the image. The classification accuracy is not improved, but the robustness of the CNN is reduced. The center-cropped SAR image removes part of the background information in the image but does not reduce the effective information in the image, so the accuracy of the CNN has not decreased, but the robustness has been improved. The more effective the information in the image extracted by the CNN is, the higher the classification accuracy of the CNN. The less redundant the information of the image used to train the CNN is, the more robust the model. The balance between accuracy and robustness should be considered when using CNN.

## V. Conclusion and Discussion

In this article, we used FGSM and BIM to generate AEs of SAR image classification models based on CNNs. The results show that CNNs trained on SAR images are very susceptible to ASIs, and the more complex the structure of the CNN is, the easier it is for the CNN to be successfully attacked by ASIs. We then proposed AESD to expound on the attack selectivity of ASIs, which provided the theoretical basis for targeted attacks. Finally, we analyzed the effects of parameters, such as image size and number of channels, on the attack success rates of ASIs. The results show that reducing the image size and number of channels can reduce the attack success rate of ASIs. Our work provides an experimental reference for the attack and defense capabilities of various CNNs against AEs in SAR image classification models.

The proposal of AESD provides theoretical guidance for our next work. We will design new attack and defense methods based on the aforementioned research in the future. On the one hand, specifying the attack direction on the basis of AESD can make the original sample cross the decision boundary with almost the shortest distance and reach the space of the specified class to quickly achieve the target attack. On the other hand, AESD, which is used to quantify the attack selectivity of AEs as a feature of the sample, can be used to detect whether the input sample is an AE.

However, many problems still require further research. Because there are no pixel-level labels of the SAR target, we are forced to study the AEs of SAR images by perturbing the whole image. If only perturbations are generated for the target, the pixel-level labeling of the target is required. We can discard the AP in the background and only add the AP to the target to enable the target to be incorrectly recognized by the CNN. The rationale could be adopted as follows: We can create a mask based on the pixel-level label (the value of the target area is 1, and the value of the background area is 0). Then, the gradient of the loss function to the input image can be calculated and multiplied by the mask to change only the pixel occupied by the target, which can achieve the purpose of only adding the AP to the target.

Our current experiment is purely digital, i.e., we have not implemented AE physically. Currently, it is extremely difficult and expensive for us to obtain SAR images at any time to test the attack effect of AEs in the physical world. The task of generating AEs in the remote sensing field under realistic physical constraints is still an unresolved problem. Physical attacks can be performed by changing properties, such as reflectivity, for example by adding materials or changing surface textures. Using the material reflection feature database, we can determine the set of achievable reflection perturbations. We can limit the AP that we generate through the perturbation set to meet the reflection characteristics of these materials and ensure that the AP can be restored in the real world. AEs are very susceptible to environmental changes (such as light, clouds, etc.), and physical AEs should be robust to such changes. While the exact mechanism for enhancing the robustness of AEs is beyond the scope of our manuscript, we hope that interested researchers can start related research to jointly solve the current problems.

## Acknowledgment

This work was carried out in part using computing resources at the High Performance Computing Platform of Central South University.

## References

[1] P. Tait, *Introduction to Radar Target Recognit.*, vol. 18. London, U.K.: Inst. Eng. Technol., 2005.

[2] A. Eryildirim and A. E. Cetin, "Man-made object classification in SAR images using 2-D cepstrum," in *Proc. IEEE Radar Conf.*, 2009, pp. 1–4.

[3] D. Gaglione, C. Clemente, L. Pallotta, I. Proudler, A. De Maio, and J. J. Soraghan, "Krogager decomposition and pseudo-Zernike moments for polarimetric distributed ATR," in *Proc. Sensor Signal Proc. Defence*, 2014, pp. 1–5.

[4] C. Clemente, L. Pallotta, I. Proudler, A. De Maio, J. J. Soraghan, and A. Farina, "Pseudo-Zernike-based multi-pass automatic target recognition from multi-channel synthetic aperture radar," *IET Radar, Sonar, Navigat.*, vol. 9, no. 4, pp. 457–466, 2015.

[5] Y. Sun, L. Du, Y. Wang, Y. Wang, and J. Hu, "SAR automatic target recognition based on dictionary learning and joint dynamic sparse representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1777–1781, Dec. 2016.

[6] C. Clemente, L. Pallotta, D. Gaglione, A. De Maio, and J. J. Soraghan, "Automatic target recognition of military vehicles with Krawtchouk moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 1, pp. 493–500, Feb. 2017.

[7] X. Yuan, T. Tang, D. Xiang, Y. Li, and Y. Su, "Target recognition in SAR imagery based on local gradient ratio pattern," *Int. J. Remote Sens.*, vol. 35, no. 3, pp. 857–870, 2014.

[8] D. Xiang, T. Tang, Y. Ban, and Y. Su, "Man-made target detection from polarimetric SAR data via nonstationarity and asymmetry," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 9, no. 4, pp. 1459–1469, Apr. 2016.

[9] J. Wang, T. Zheng, P. Lei, and X. Bai, "Ground target classification in noisy SAR images using convolutional neural networks," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 4180–4192, Nov. 2018.

[10] J. Shao, C. Qu, and J. Li, "A performance analysis of convolutional neural network models in SAR target recognition," in *Proc. SAR Big Data Era: Models, Methods, Appl.*, 2017, pp. 1–6.

[11] O. Lay *et al.*, "MSTAR: A submicrometer absolute metrology system," *Opt. Lett.*, vol. 28, no. 11, pp. 890–892, 2003.

[12] R. Shang, J. Wang, L. Jiao, R. Stolkin, B. Hou, and Y. Li, "SAR targets classification based on deep memory convolution neural networks and transfer parameters," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2834–2846, Aug. 2018.

[13] K. Huang, W. Nie, and N. Luo, "Fully polarized SAR imagery classification based on deep reinforcement learning method using multiple polarimetric features," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 12, no. 10, pp. 3719–3730, Oct. 2019.

[14] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 828–841, Oct. 2019.

[15] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 427–436.

[16] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2805–2824, Sep. 2019.

[17] J. Gilmer, N. Ford, N. Carlini, and E. Cubuk, "Adversarial examples are a natural consequence of test error in noise," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2280–2289.

[18] G. Elsayed *et al.*, "Adversarial examples that fool both computer vision and time-limited humans," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 3910–3920.

[19] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Statistics*, vol. 1050, p. 20, 2015.

[20] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," 2016, *arXiv:1607.02533*.

[21] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy*, 2017, pp. 39–57.

[22] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1765–1773.

[23] W. Czaja, N. Fendley, M. Pekala, C. Ratto, and I.-J. Wang, "Adversarial examples in remote sensing," in *Proc. 26th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2018, pp. 408–411.

[24] W. Li *et al.*, "Spear and shield: Attack and detection for CNN-based high spatial resolution remote sensing images identification," *IEEE Access*, vol. 7, pp. 94583–94592, 2019.

[25] Z. Deng, H. Sun, S. Zhou, J. Zhao, and H. Zou, "Toward fast and accurate vehicle detection in aerial images using coupled region-based convolutional neural networks," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3652–3664, Aug. 2017.

[26] F. Wu, Z. Zhou, B. Wang, and J. Ma, "Inshore ship detection based on convolutional neural network in optical satellite images," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 4005–4015, Nov. 2018.

[27] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, "Urban land cover classification with missing data modalities using deep convolutional neural networks," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 11, no. 6, pp. 1758–1768, Jun. 2018.

[28] H. Luo, C. Chen, L. Fang, X. Zhu, and L. Lu, "High-resolution aerial images semantic segmentation using deep fully convolutional network with channel attention mechanism," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3492–3507, Sep. 2019.

[29] H. Li *et al.*, "RS-MetaNet: Deep meta metric learning for few-shot remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–12, 2020.

[30] C. Szegedy *et al.*, "Intriguing properties of neural networks," in *Proc. 2nd Int. Conf. Learn. Representations*, 2014. pp. 1–10.

[31] S. Bubeck, E. Price, and I. Razenshteyn, "Adversarial examples from computational constraints," in *Proc. Int. Conf. Mach. Learn. PMLR*, 2019, pp. 831–840.

[32] X. Ma *et al.*, "Characterizing adversarial subspaces using local intrinsic dimensionality," in *Proc. 6th Int. Conf. Learn. Representations*, 2018. pp. 1–15.

[33] S. Gulshad, J. H. Metzen, A. Smeulders, and Z. Akata, "Interpreting adversarial examples with attributes," 2019, *arXiv:1904.08279*.

[34] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," in *Proc. Adv. Neural Inf. Proc. Syst.*, 2019, pp. 125–136.

[35] A. Boloor, X. He, C. Gill, Y. Vorobeychik, and X. Zhang, "Simple physical adversarial examples against end-to-end autonomous driving models," in *Proc. IEEE Int. Conf. Embedded Softw. Syst.*, 2019, pp. 1–7.

[36] A. J. Bose and P. Aarabi, "Adversarial attacks on face detectors using neural net based constrained optimization," in *Proc. IEEE 20th Int. Workshop Multimedia Signal Proc.*, 2018, pp. 1–6.

[37] Z.-A. Zhu, Y.-Z. Lu, and C.-K. Chiang, "Generating adversarial examples by makeup attacks on face recognition," in *Proc. IEEE Int. Conf. Image Proc.*, 2019, pp. 2516–2520.

[38] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *Proc. IEEE Eur. Symp. Secur. Privacy*, 2016, pp. 372–387.

[39] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2574–2582.

[40] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 3905–3911.

[41] K. Eykholt *et al.*, "Physical adversarial examples for object detectors," in *Proc. 12th USENIX Conf. Offensive Technol.*, 2018, p. 1.

[42] S.-T. Chen, C. Cornelius, J. Martin, and D. H. P. Chau, "ShapeShifter: Robust physical adversarial attack on faster R-CNN object detector," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2018, pp. 52–68.

[43] V. Fischer, M. C. Kumar, J. H. Metzen, and T. Brox, "Adversarial examples for semantic image segmentation," *Statistics*, vol. 1050, p. 3, 2017.

[44] A. Arnab, O. Miksik, and P. H. Torr, "On the robustness of semantic segmentation models to adversarial attacks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 888–897.

[45] M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang, "Generating natural language adversarial examples," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2890–2896.

[46] S. Ren, Y. Deng, K. He, and W. Che, "Generating natural language adversarial examples through probability weighted word saliency," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1085–1097.

[47] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *Proc. IEEE Secur. Privacy Workshops*, 2018, pp. 1–7.

[48] Y. Qin, N. Carlini, I. Goodfellow, G. Cottrell, and C. Raffel, "Imperceptible, robust, and targeted adversarial examples for automatic speech recognition," 2019, *arXiv:1903.10346*.

[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[50] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

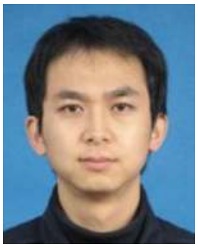
[51] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[53] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1492–1500.

[54] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[55] M. Schmitt, L. H. Hughes, and X. X. Zhu, "The SEN1-2 dataset for deep learning in SAR-optical data fusion," *ISPRS Annals Photogrammetry, Remote Sensing Spatial Inf. Sci.*, vol. 4, no. 1, pp. 141–146, 2018.



**Haifeng Li** (Member, IEEE) received the master's degree in transportation engineering from the South China University of Technology, Guangzhou, China, in 2005, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2009.

He is currently a Professor with the School of Geosciences and Info-Physics, Central South University, Changsha, China. He was a Research Associate with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong, in 2011, and a Visiting Scholar with the University of Illinois at Urbana–Champaign, Urbana, IL, USA, from 2013 to 2014. He is a Reviewer for many journals. He has authored more than 30 journal papers. His current research interests include geo/remote sensing big data, machine/deep learning, and artificial/brain-inspired intelligence.

**Haikuo Huang** is currently working toward the master's degree in surveying and mapping science and technology from Central South University, Changsha, China.

His research interests include transfer learning, deep learning and visualization.

**Li Chen** received the B.S. and M.S. degrees in software engineering from Central South University, Changsha, China, in 2015 and 2018, respectively. He is currently working toward the Ph.D. degree in surveying and mapping science and technology from Central South University.

His research interests include remote sensing images understanding and robust machine learning.

**Jian Peng** received the B.S. degree in geographic information science from Yunnan University, Kunming, China, in 2015. He is currently working toward the Ph.D. degree in surveying and mapping science and technology from Central South University, Changsha, China.

His research interests include memory model, deep learning, and remote sensing images understanding.

**Haozhe Huang** received the B.S. degree in remote sensing science and technology from Wuhan University, Wuhan, China. He is currently working toward the Ph.D. degree in surveying and mapping science and technology from Central South University, Changsha, China.

His research interests include remote sensing images understanding and auto machine learning.

**Zhenqi Cui** received the B.S. degree in the geographic information science from Central South University, Changsha, China. He is currently working toward the Ph.D. degree in surveying and mapping science and technology from Central South University.

His research interests include meta learning and remote sensing images understanding.

**Xiaoming Mei** received the B.S. degree in applied geophysics and the M.S. degree in geodetection and information technology from the Central South University, Changsha, China, in 2000 and 2003, respectively, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2007.

His research interests include multi- and hyperspectral remote sensing data processing, high-resolution image processing and scene analysis, LiDAR data processing, and computational intelligence.

**Guohua Wu** received the B.S. degree in information systems and the Ph.D. degree in operations research from the National University of Defense Technology, Changsha, China, in 2008 and 2014, respectively.

From 2012 to 2014, he was a Visiting Researcher with the University of Alberta, Edmonton, AB, Canada. He is currently a Professor with the School of Traffic and Transportation Engineering, Central South University, Changsha, China. He has authored more than 50 referred papers, including those published in the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, *Information Sciences*, and *Computers and Operations Research*. His research interests include intelligent planning and scheduling, evolutionary computation, and data mining.

Dr. Wu serves as an Editorial Board Member of the *International Journal of Bio-Inspired Computation*, as a Guest Editor for *Information Sciences* and *Memetic Computing*, and as an Associate Editor for the *Swarm and Evolutionary Computation*.