Black-Box Universal Adversarial Attack for DNN-Based Models of SAR Automatic Target Recognition

Xuanshen Wan, Wei Liu, Chaoyang Niu, Wanjie Lu, Meng Du, and Yuanli Li

***Abstract*—Synthetic aperture radar automatic target recognition (SAR-ATR) models based on deep neural networks (DNNs) are vulnerable to attacks of adversarial examples. Universal adversarial attack algorithms can help evaluate and improve the robustness of the SAR-ATR models and have become a research hotspot. However, current universal adversarial attack algorithms have limitations. First, considering the difficulty in obtaining information on the attacking SAR-ATR models, there is an urgent need to design a universal adversarial attack algorithm under a black-box scenario. Second, given the difficulty of acquiring synthetic aperture radar images, the effectiveness of attacks under small-sample conditions requires improvement. To address these limitations, this study proposed a black-box universal adversarial attack algorithm: transferable universal adversarial network (TUAN). Based on the idea of the generative adversarial network, we implemented the game of generator and attenuator to improve the transferability of universal adversarial perturbation (UAP). We designed loss functions for the generator and the attenuator, respectively, which can effectively improve the success rate of black-box attacks and the stealthiness of attacks. In addition, U-Net was used as a network structure of the generator and the attenuator to fully learn the distribution of examples, thereby enhancing the attack success rate under small-sample conditions. The TUAN attained a higher black-box attack success rate and superior stealthiness than up-to-date UAP algorithms in non-targeted and targeted attacks.**

***Index Terms*—Adversarial example, automatic target recognition, deep neural network (DNN), synthetic aperture radar (SAR), transferability, universal adversarial perturbation (UAP).**

I. Introduction

In RECENT years, deep neural networks (DNNs) have been widely used in the field of synthetic aperture radar automatic target recognition (SAR-ATR) [1], [2], [3]. However, existing research proves that SAR-ATR models based on DNNs [4], [5], [6], [7], [8] are vulnerable to adversarial examples. Therefore, studying universal adversarial attack algorithms is conducive to evaluating and improving the recognition performance of SAR-ATR models under small disturbance conditions.

Manuscript received 6 December 2023; revised 31 January 2024, 29 February 2024, and 23 March 2024; accepted 23 March 2024. Date of publication 2 April 2024; date of current version 29 April 2024. This work was supported by the National Natural Science Foundation of China under Grant 42201472. *(Corresponding author: Wei Liu.)*

The authors are with the Information Engineering University, Zhengzhou 450001, China (e-mail: allwan0010@163.com; greatliuliu@163.com; niucy2017@outlook.com; lwj285149763@163.com; dumeng_nudt@163.com; likaojin@163.com).

Digital Object Identifier 10.1109/JSTARS.2024.3384188

At present, the adversarial attack algorithm for the SAR-ATR model is in its infancy, and scholars mainly migrate the adversarial attack algorithm in optical images to synthetic aperture radar (SAR) images. In the field of optical imaging, many adversarial attack algorithms have been proposed. Szegedy et al. [9] first proposed the concept of adversarial examples, which injects small perturbations that are imperceptible to the human eye into the image, thereby causing model recognition errors. This method is called an adversarial attack. On this basis, researchers have proposed a series of universal adversarial attack algorithms. Moosavi-Dezfooli et al. [10] first proposed a universal adversarial perturbation (UAP) generation algorithm. This type of attack algorithm can generate adversarial examples that are independent of the input image. It does not need to generate a specific perturbation image for each input image and can attack most examples. Subsequently, Hayes and Danezis [11] designed a trainable DNN to learn the mapping of noise to UAP and demonstrated that this method can quickly and effectively deceive the victim model. Mopuri et al. [12], [13] considered that it is difficult for attackers to obtain the training dataset of the victim model and proposed a data-free attack method to generate UAP. Mopuri et al. [14] first proposed a generative method to simulate the distribution of adversarial perturbations. Li et al. [15] proved that adversarial examples can effectively attack the SAR-ATR model. Subsequently, scholars began to study UAP algorithms for SAR images. Wang et al. [16] used the method proposed in [10] to construct UAP, which effectively reduced the recognition success rate of the SAR-ATR model. Xia et al. [17] combined the principle of SAR interference to generate UAP in the signal domain. Du et al. [18] proposed an adversarial attack algorithm based on the universal local adversarial network. This algorithm only needs to perturb one-fourth of the original SAR image to achieve an attack success rate comparable to that of global perturbation. Zhou et al. [19] proposed a UAP method for SAR image adversarial attacks based on a lightweight generative model without a discriminator template. Tested on eight SAR-ATR models, the experimental results indicated that SAR-UAP obtained a high attack success rate.

Adversarial attack algorithms can be divided into white- and black-box attacks. In a white-box attack scenario, the attacker knows all the information, such as the structure, parameters, and training data of the victim model. Typical white-box attack methods include gradient-based attacks [20], [21], boundary-based attacks [22], and saliency-map-based attacks [23]. On

the contrary, in the black-box attack scenario, it is difficult for the attacker to obtain the information of the victim model. In general, black-box attacks can be divided into probabilistic label-based attacks [24], [25], [26], decision-based attacks [27], and transferred attacks [28], [29]. Among the above three black-box attacks, the first two black-box attacks usually require a large number of queries to the neural network. However, this is difficult to achieve in actual scenarios. Therefore, transferable black-box attacks are the current research focus.

Although the existing UAP algorithms for SAR images can effectively deceive the SAR-ATR model, they are fragile and inefficient in practical applications. In actual applications, the SAR-ATR model is unknown to attackers, and the black-box attack success rate of the existing universal adversarial attack algorithms is low. Therefore, there is an urgent need to develop a UAP algorithm for SAR images under black-box scenarios. In addition, the number of SAR sensors is significantly smaller than that of optical sensors, and SAR sensors are limited by airborne or spaceborne platforms. Therefore, the amount of SAR data that can be obtained in practice are limited, and the attack success rate of the existing UAP algorithms is significantly lower under small-sample conditions. Thus, it is of great significance to improve the attack success rate of universal adversarial attack algorithms under small-sample conditions.

To solve these problems, this study proposes a transferable universal adversarial network (TUAN). Based on the idea of the generative adversarial network (GAN), this method quickly maps noise to UAP in one step through the generator and then uses the attenuator to weaken the attack effectiveness of the adversarial examples. We argue that if the adversarial examples crafted by the generator are robust to the deformations produced by the attenuator, i.e., the attenuated adversarial examples remain effective to DNN models, then they can be transferred to other victim models [30].

The main contributions of this article are as follows.

1) A TUAN based on a generator and an attenuator is designed. Similar to the GAN, we leverage the game of the generator and attenuator to boost the transferability of UAP. We utilize U-Net to improve the attack success rate under small-sample conditions. Therefore, our algorithm has wide application prospects in the field of SAR-ATR attacks.
2) We design loss functions separately for the generator and the attenuator, which effectively enhance the black-box attack capability and attack stealthiness of universal adversarial examples in both the non-targeted and targeted attacks.
3) The proposed algorithm was tested on the MSTAR and SEN1-2 datasets. The experimental results show that compared with the existing universal adversarial attack algorithms, the TUAN has the strongest transferability and small-sample attack performance in non-targeted and targeted attacks.
4) We systematically evaluate the transferability of UAP among DNN-based SAR-ATR models; the experimental results showed that the SAR-ATR models have the same vulnerabilities when performing the same tasks.

The rest of this article is organized as follows. Section II introduces the relevant preparation knowledge. Section III describes the proposed method in detail. Section IV presents the experimental results. Section V discusses the results. Finally, Section VI concludes this article.

II. Preliminaries

A. UAPs for SAR Target Recognition

Suppose that χ is the SAR image dataset, $x_n \in [0,255]^{W\times H}$ is the nth SAR image example, and $f(\cdot)$ is the output of the m-class SAR-ATR model that has not passed the softmax layer. The m-dimensional vector $f(x_n) = [f(x_n)_1, f(x_n)_2, \ldots, f(x_n)_m]$ is the output result, where $f(x_n)_i \in \mathbb{R}$ is the probability that x_n is recognized by the model as category i. Let $S_p = \arg\max_i(f(x_n)_i)$ represent the model prediction class of x_n. UAP is independent of the input data, and instead of generating perturbations for a particular image, most samples in the dataset result in incorrect model identification when UAP is added

$$for''\text{most}''x_n \in \chi \text{ s.t.} \begin{cases} \arg\max_i(f(x_n+\delta)_i) \neq S_p \\ \|\delta\|_p \leq \zeta \end{cases} \tag{1}$$

$$\|\delta\|_p = \left(\sum_i |\delta_i|^p\right)^{\frac{1}{p}} \tag{2}$$

where δ is UAP and ζ controls the disturbance amplitude of UAP.

At the same time, attack methods are classified as non-targeted and targeted attacks depending on the attacker's intent and expectations. Targeted attacks require the attacker to cause the DNN models to produce the desired results, as opposed to non-targeted attacks, which merely require the attacker to misclassify the DNN model. Therefore, (1) is transformed into the following optimization problem:

$$\text{minimize}\left(\frac{\sum_{n=1}^{N} Z(\arg\max_i(f(x_n+\delta)_i) == C_{tr})}{N}\right),$$
$$\text{s.t. } \|\delta\|_p \leq \zeta \tag{3}$$

$$\text{maximize}\left(\frac{\sum_{n=1}^{N} Z(\arg\max_i(f(x_n+\delta)_i) == C_{ta})}{N}\right),$$
$$\text{s.t. } \|\delta\|_p \leq \zeta \tag{4}$$

where if the equation holds, then the discriminant function $Z(\cdot) = 1$, and conversely, $Z(\cdot) = 0$. N represents the total number of input samples, and C_{tr} and C_{ta} represent its true and target categories, respectively.

B. Transferability of Adversarial Examples

Research based on transferable black-box attacks has important value as an attacker cannot obtain information or feedback from the victim model. Fig. 1 demonstrates the transferability of adversarial examples.

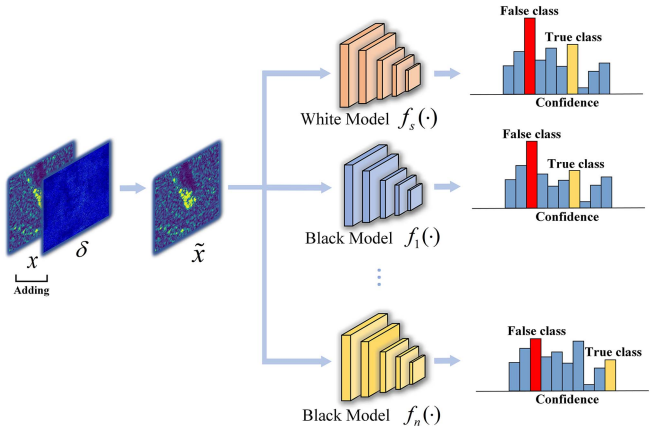


Fig. 1. Transferability of adversarial examples.

In Fig. 1, for an image multiclassification task, $f_1(\cdot), \ldots, f_n(\cdot)$, along with $f_s(\cdot)$, are the trained identification models. δ is the adversarial perturbation. x and $\tilde{x}$ are the input example and the adversarial example, respectively. Among them, $f_s(\cdot)$ is used as a white-box model to generate adversarial examples, whereas $f_1(\cdot), \ldots, f_n(\cdot)$ are black-box models. The histogram on the right-hand side of Fig. 1 shows the classification results. The greater the height of the bar, the higher the confidence. The red bars represent the classes that the neural network misclassified, and the yellow bars represent the correct classes. As shown in the figure, the adversarial example $\tilde{x}$ generated by $f_s(\cdot)$ can successfully cause $f_1(\cdot), \ldots, f_n(\cdot)$ to output incorrect classification results.

III. Methods

The framework of the TUAN is shown in Fig. 2. Based on the idea of GAN [31], the transferability of UAP is enhanced by the TUAN through the utilization of a mutual game involving the generator $G(\cdot)$ and the attenuator $A(\cdot)$. During the training process of the TUAN, the generator $G(\cdot)$ is first used to train the mapping of the normally distributed noise Z to the UAP δ, and δ is added to the input example x to obtain the adversarial example$\tilde{x}$. Then, the attenuator $A(\cdot)$ can reduce the attack effectiveness of adversarial example $\tilde{x}$. To improve the success rate of the black-box attack and the stealthiness of the attack, we designed the corresponding loss functions L_G and L_A for the generator $G(\cdot)$ and the attenuator $A(\cdot)$, respectively. If the attenuated adversarial example $\tilde{x}^{\#}$ can effectively attack the DNN victim model $f_v(\cdot)$, that is, the adversarial example $\tilde{x}$ constructed by the generator $G(\cdot)$ is robust to the deformations produced by the attenuator $A(\cdot)$ [30], then $\tilde{x}$ can successfully attack the black-box victim model. In addition, we used the U-Net [32] model as a network structure of the generator and attenuator to improve the attack success rate under small-sample conditions.

A. Network Structure of the Generator and Attenuator

The characteristics of SAR images were fully considered in the selection of the generator and attenuator. As shown in Fig. 3, the SAR image contains a target, shadow, and background. The characteristics identified by the DNN model are mainly focused on the target area. Moreover, considering the confidentiality of SAR images, attackers find it difficult to obtain SAR datasets. Therefore, the attacker must consider adversarial attacks under small-sample conditions.

Even though there are numerous improved networks [33], [34] based on U-Net, but the U-Net model has its unique advantages in SAR attacks, and it is widely used in mainstream SAR adversarial attack algorithms [18], [35], [36]. The reasons for selecting U-Net as the network architecture can be outlined as follows.

First, combined with the real-time and limited computing resources of SAR attacks, the U-Net model can be quickly and effectively deployed in actual attacks with its lightweight network structure.

Then, U-Net is a fully convolutional network that uses skip connections and feature combination to effectively capture important features under small-sample conditions. Researches show that U-Net can be trained with very few training samples but produces results with high accuracy [37], [38], [39], [40], [41], which is of great significance for solving small-sample problems.

Besides, the size of the output adversarial example must be the same as that of the original SAR image. The U-Net model can effectively ensure the size consistency of input and output.

Based on the above considerations, this study utilized U-Net as the encoder/decoder. The network structure of U-Net is shown in Fig. 4. The encoder, located on the left, functions as feature extraction, while the decoder, positioned on the right, acts as an upsampling component.

B. Loss Function of the Generator

Fig. 5 shows the loss function of the generator $G(\cdot)$. Note that, during the training phase, the white-box model was selected as the surrogate model $f_v(\cdot)$. The algorithm in this study uses a generator $G(\cdot)$ to map the normally distributed noise Z to the UAP δ. In the testing phase, the algorithm does not need to create specific perturbations for each SAR image. UAP δ can attack most of the SAR images in the testing set

$$\delta = G(Z), \quad Z \sim \mathbb{N}(0, 1). \tag{5}$$

UAP δ is then added to the input example x to obtain the adversarial example $\tilde{x}$

$$\tilde{x} = x + \delta. \tag{6}$$

Simultaneously, the adversarial example $\tilde{x}$ is passed through the attenuator $A(\cdot)$ to obtain the attenuated adversarial example $\tilde{x}^{\#}$

$$\tilde{x}^{\#} = A(\tilde{x}). \tag{7}$$

The loss function L_G designed in this article can realize non-targeted and targeted attacks, and the attacker can choose the corresponding attack mode according to actual needs. In particular, we designed different loss functions L_G for each attack mode, with each consisting of three parts.

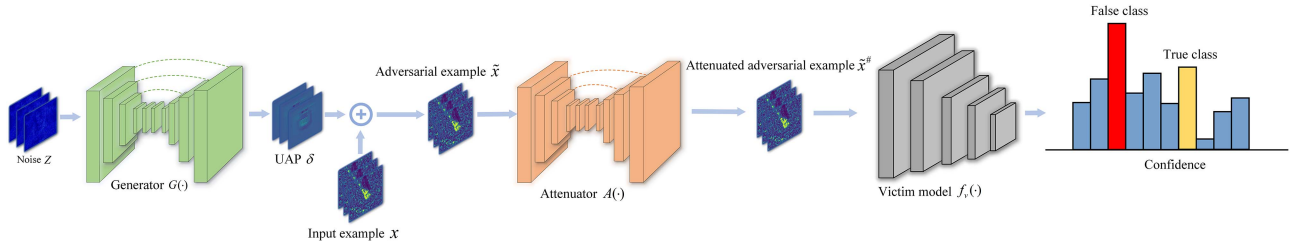


Fig. 2. Framework of the TUAN.

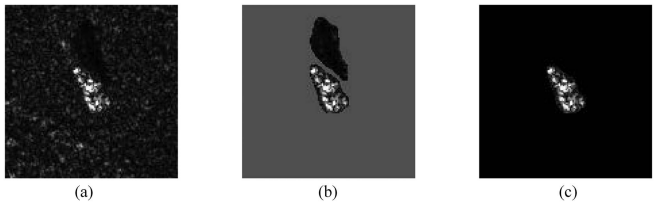


Fig. 3. Example of SAR image region analysis. (a) Original SAR images. (b) Three regions of the image. (c) Vehicle region.

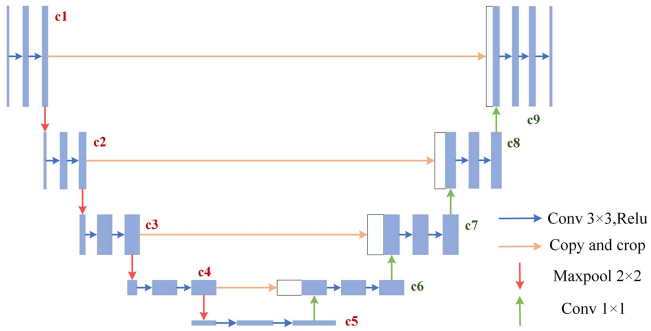


Fig. 4. U-Net network structure.

For non-targeted attacks, to effectively deceive the DNN model, it is important to decrease the confidence level that $\tilde{x}$ is identified as the true category C_{tr} and increase the confidence level that the adversarial example $\tilde{x}$ is identified as another category. Therefore, L_{G1} can be represented by the following equation:

$$L_{G1}(f_v(\tilde{x}), C_{tr}) = -\log\left(\frac{\sum_{i\neq C_{tr}} \exp(f_v(\tilde{x})_i)}{\sum_i \exp(f_v(\tilde{x})_i)}\right). \quad (8)$$

To increase the transferability of the adversarial example $\tilde{x}$, L_{G2} aims to improve the attack success rate of attenuated adversarial examples $\tilde{x}^{\#}$. Therefore, L_{G2} is expressed as follows:

$$L_{G2}(f_v(\tilde{x}^{\#}), C_{tr}) = -\log\left(\frac{\sum_{i\neq C_{tr}} \exp(f_v(\tilde{x}^{\#})_i)}{\sum_i \exp(f_v(\tilde{x}^{\#})_i)}\right). \quad (9)$$

Finally, to limit the amplitude of disturbance, we used $L_p{-}norm$ to measure the distance between the adversarial example $\tilde{x}$ and the original example x. The specific formula is as follows:

$$L_{G3}(x, \tilde{x}) = \|\tilde{x} - x\|_p = \left(\sum_i |\Delta x_i|^p\right)^{\frac{1}{p}}. \quad (10)$$

To sum up, this study used the linear weighting method to balance the relationship between L_{G1}, L_{G2}, and L_{G3}. L_G is calculated as follows:

$$L_G = w_{G1} \cdot L_{G1}(f_v(\tilde{x}), C_{tr}) + w_{G2} \cdot L_{G2}(f_v(\tilde{x}^{\#}), C_{tr}) + w_{G3} \cdot L_{G3}(x, \tilde{x}) \quad (11)$$

$$w_{G1} + w_{G2} + w_{G3} = 1 \quad (12)$$

where w_{G1}, w_{G2}, and w_{G3} are the weight coefficients of L_{G1}, L_{G2}, and L_{G3}, respectively. In the loss function designed in this article, the attacker can adjust the weight coefficient of each item according to actual needs. Therefore, the algorithm proposed in this article has good flexibility.

For targeted attacks, to designate the output of the DNN models as a certain category C_{ta}, it is necessary to increase the confidence that the adversarial example $\tilde{x}$ is identified by DNN models as a specific category C_{ta}. Thus, L_{G1} is given as follows:

$$L_{G1}(f_v(\tilde{x}), C_{ta}) = -\log\left(\frac{\exp(f_v(\tilde{x})_{C_{ta}})}{\sum_i \exp(f_v(\tilde{x})_i)}\right). \quad (13)$$

Meanwhile, it is necessary to maintain the attenuated adversarial example $\tilde{x}^{\#}$ to successfully attack the DNN model $f_v(\cdot)$. Therefore, L_{G2} can be represented as follows:

$$L_{G2}(f_v(\tilde{x}^{\#}), C_{ta}) = -\log\left(\frac{\exp(f_v(\tilde{x}^{\#})_{C_{ta}})}{\sum_i \exp(f_v(\tilde{x}^{\#})_i)}\right). \quad (14)$$

L_{G3} in the targeted attack is the same as that in the non-targeted attack, and the expression for the targeted attack L_G is as follows:

$$L_G = w_{G1} \cdot L_{G1}(f_v(\tilde{x}), C_{ta}) + w_{G2} \cdot L_{G2}(f_v(\tilde{x}^{\#}), C_{ta}) + w_{G3} \cdot L_{G3}(x, \tilde{x}). \quad (15)$$

C. Loss Function of the Attenuator

The loss function of the attenuator has been illustrated in Fig. 6. We introduce the attenuator $A(\cdot)$ to play games with the generator $G(\cdot)$ while effectively preserving the semantic information of x and weakening $\tilde{x}$. The loss function L_A comprises

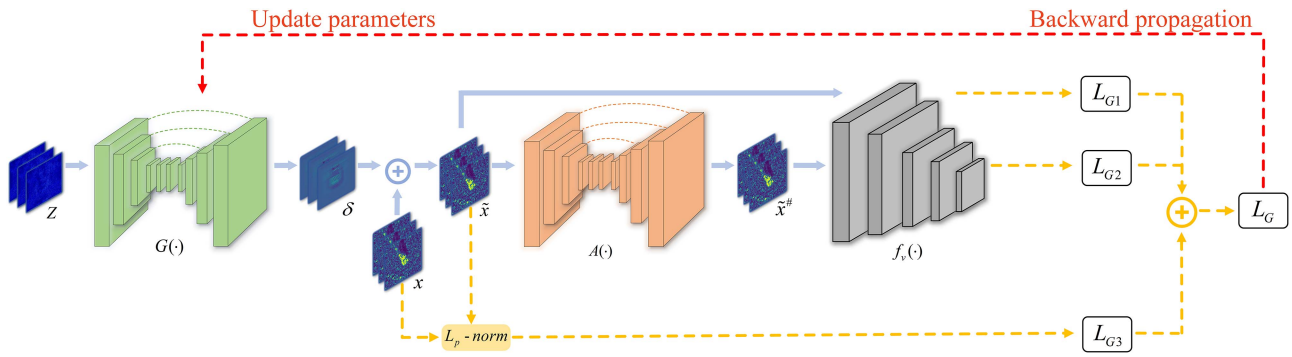


Fig. 5. Generator loss function diagram.

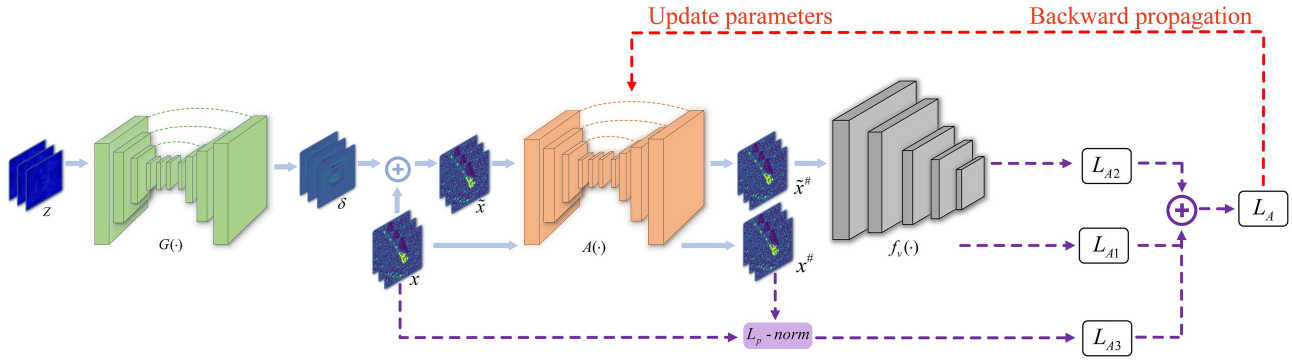


Fig. 6. Attenuator loss function diagram.

three parts. In the first part, L_{A1}, to retain the semantic information of x, the DNN model needs to increase the identification accuracy of $x^{\#}$, as follows:

$$L_{A1}(f_v(x^{\#}), C_{tr}) = -\log\left(\frac{\exp(f_v(x^{\#})_{C_{tr}})}{\sum_i \exp(f_v(x^{\#})_i)}\right). \quad (16)$$

Simultaneously, to weaken the effectiveness of the attack of $\tilde{x}$, L_{A2} aims to improve the confidence that $\tilde{x}^{\#}$ is identified as the correct category C_{tr} by the DNN model. Therefore, L_{A2} is expressed as follows:

$$L_{A2}(f_v(\tilde{x}^{\#}), C_{tr}) = -\log\left(\frac{\exp(f_v(\tilde{x}^{\#})_{C_{tr}})}{\sum_i \exp(f_v(\tilde{x}^{\#})_i)}\right). \quad (17)$$

Finally, as shown in (18), by introducing the traditional $L_p{-}norm$ to limit the deformation amplitude, the attenuator $A(\cdot)$ is prevented from producing serious image deformation

$$L_{A3}(x, x^{\#}) = \left\|x^{\#} - x\right\|_p = \left(\sum_i |\Delta x_i|^p\right)^{\frac{1}{p}}. \quad (18)$$

To sum up, similar to the calculation method of L_G, we used three weight coefficients, i.e., w_{A1}, w_{A2}, and w_{A3}, to balance the relationship between L_{A1}, L_{A2}, and L_{A3}. Therefore, $L_A$is calculated as follows:

$$L_A = w_{A1} \cdot L_{A1}(f_v(x^{\#}), C_{tr}) + w_{A2} \cdot L_{A2}(f_v(\tilde{x}^{\#}), C_{tr}) + w_{A3} \cdot L_{A3}(x, x^{\#}). \quad (19)$$

D. Complete Training Process of the TUAN

Similar to the GAN, the TUAN uses the mutual game of the generator $G(\cdot)$ and attenuator $A(\cdot)$ to improve the transferability of UAP δ. Therefore, during the training process of the TUAN, we used the alternating training method to train the generator $G(\cdot)$ and the attenuator $A(\cdot)$. For the dataset χ, assuming that the training batch size is κ, the dataset χ is divided into N batches according to the batch size κ before the start of each training session. Second, to prevent the attenuator from being too powerful for the generator to optimize the parameters, the training ratio $r \in N^*$ was set. This means that during the training process, the generator was trained r times, and the attenuator was trained once. Our complete training process for the TUAN is summarized in Algorithm 1.

IV. Experiments

A. Data Descriptions

The experiments in this study used two SAR datasets: MSTAR [42] and SEN1-2 [43]. The MSTAR dataset is extensively utilized in the domain of SAR ground target identification. As shown in Table I, the MSTAR dataset includes a total of ten categories of military objectives under standard operating conditions (SOCs). Fig. 7 shows the SAR and optical images of each target type. The SEN1-2 dataset was developed to make it easier to combine optical and SAR images. This dataset covers all the regions of the world and every weather season. In this

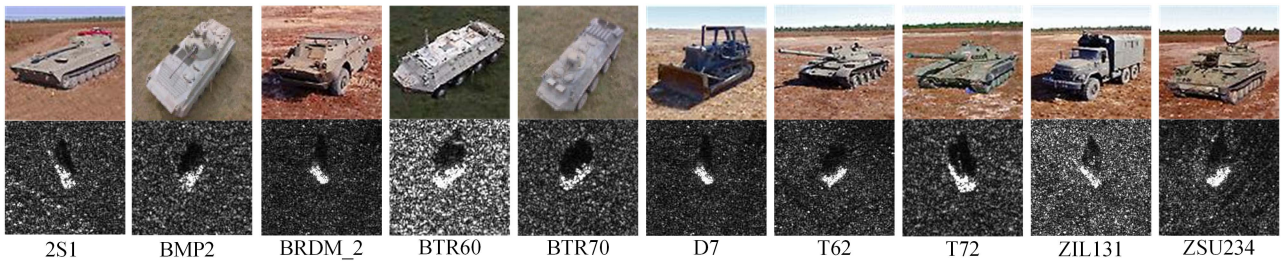


Fig. 7. Optical image (top) and the SAR image (bottom) of the MSTAR dataset.

Algorithm 1: Transferable Universal Adversarial Network.

Input: Dataset χ; surrogate model $f_v(\cdot)$; batch size κ; target class C_{ta}; training iteration number T; learning rate η; training ratio r; training loss function of the generator L_G; training loss function of the attenuator L_A

Output: The parameter θ_G of the well-trained generator.

Randomly initialize θ_G and θ_A
For $t = 1$ to T **do**
  According to κ, randomly divide χ into N batches$\{b_1, b_2, \ldots, b_N\}$
  For $n = 1$ to N **do**
    Calculate $L_G(\theta_G, \theta_A, f_v, b_n, C_{ta})$
    Update $\theta_G = \theta_G - \eta \cdot \frac{\partial}{\partial \theta_G} L_G$
    if $n\%\ r == 0$ **then**
      Calculate $L_A(\theta_G, \theta_A, f_v, b_n)$
      Update $\theta_A = \theta_A - \eta \cdot \frac{\partial}{\partial \theta_A} L_A$
    else
      $\theta_A = \theta_A$
  End For
End For

TABLE I
INFORMATION ABOUT THE MSTAR DATASET UNDER SOC

Class	Training data		Testing data	
	Depression angle	Number	Depression angle	Number
2S1	17°	299	15°	274
BMP2	17°	233	15°	196
BRDM2	17°	298	15°	274
BTR60	17°	256	15°	195
BTR70	17°	233	15°	196
D7	17°	299	15°	274
T62	17°	299	15°	273
T72	17°	232	15°	196
ZIL131	17°	299	15°	274
ZSU234	17°	299	15°	274

experiment, as shown in Fig. 8, five types of scene images taken in the summer (2017.6.1–2017.8.31) were selected. The specific information from the dataset is shown in Table II.

TABLE II
INFORMATION ABOUT THE SEN1-2 DATASET

Target class	Training data number	Testing data number
s1_47	801	801
s1_51	540	540
s1_58	450	450
s1_78	439	439
s1_106	443	443

B. Implementation Details

In terms of DNN model selection, the proposed algorithm was evaluated on six DNN models: DenseNet121 [44], GoogLeNet [45], InceptionV3 [46], MobileNet [47], ResNet50 [48], and ShuffleNet [49]. To preprocess the data, the images in the MSTAR and SEN1-2 datasets were resized to 128 × 128 pixels. At the same time, the verification dataset was randomly sampled at 10% in the training dataset. As shown in Figs. 9 and 10, the six DNN models were trained on the MSTAR and SEN1-2 datasets. The classification accuracies of the six DNN models for the MSTAR test dataset were 98.06%, 97.24%, 97.44%, 97.65%, 97.82%, and 97.77%, respectively. The classification accuracies of the six DNN models for the SEN1-2 test datasets were 94.39%, 97.87%, 99.36%, 96.11%, 99.03%, and 95.85%, respectively. In the training phase of the TUAN, as described in Sections II-B and II-C, this study adopted the traditional L_2-norm to evaluate the image distortion. The loss weight of the generator $[w_{G1}, w_{G2}, w_{G3}]$ defaulted to$[0.25, 0.25, 0.5]$, and the loss weight of the attenuator $[w_{A1}, w_{A2}, w_{A3}]$ defaulted to$[0.25, 0.25, 0.5]$. The training ratio r defaulted to 3. The number of training iteration T defaulted to 100, the batch size defaulted to 8, and the learning rate η defaulted to 0.0001.

In the comparison experiments, we used the method proposed in the literature [11], [13], [14], [16], [19], [50] as the baseline comparison method. In particular, the algorithm proposed in [16] cannot be applied to targeted attacks. The experiment used the Windows 10 operating system, PyTorch deep learning development framework, and Python as the development language. The CPU used in the experiment was an Intel Core i9-11900H, and the GPU was an NVIDIA GeForce RTX 3080.

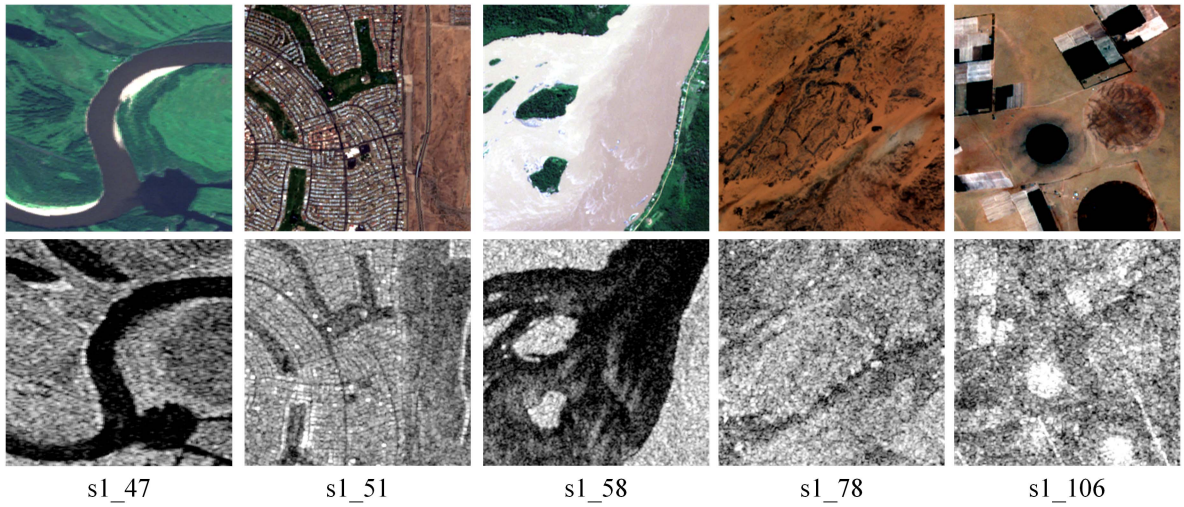


Fig. 8. Optical image (top) and the SAR image (bottom) of the SEN1-2 dataset.

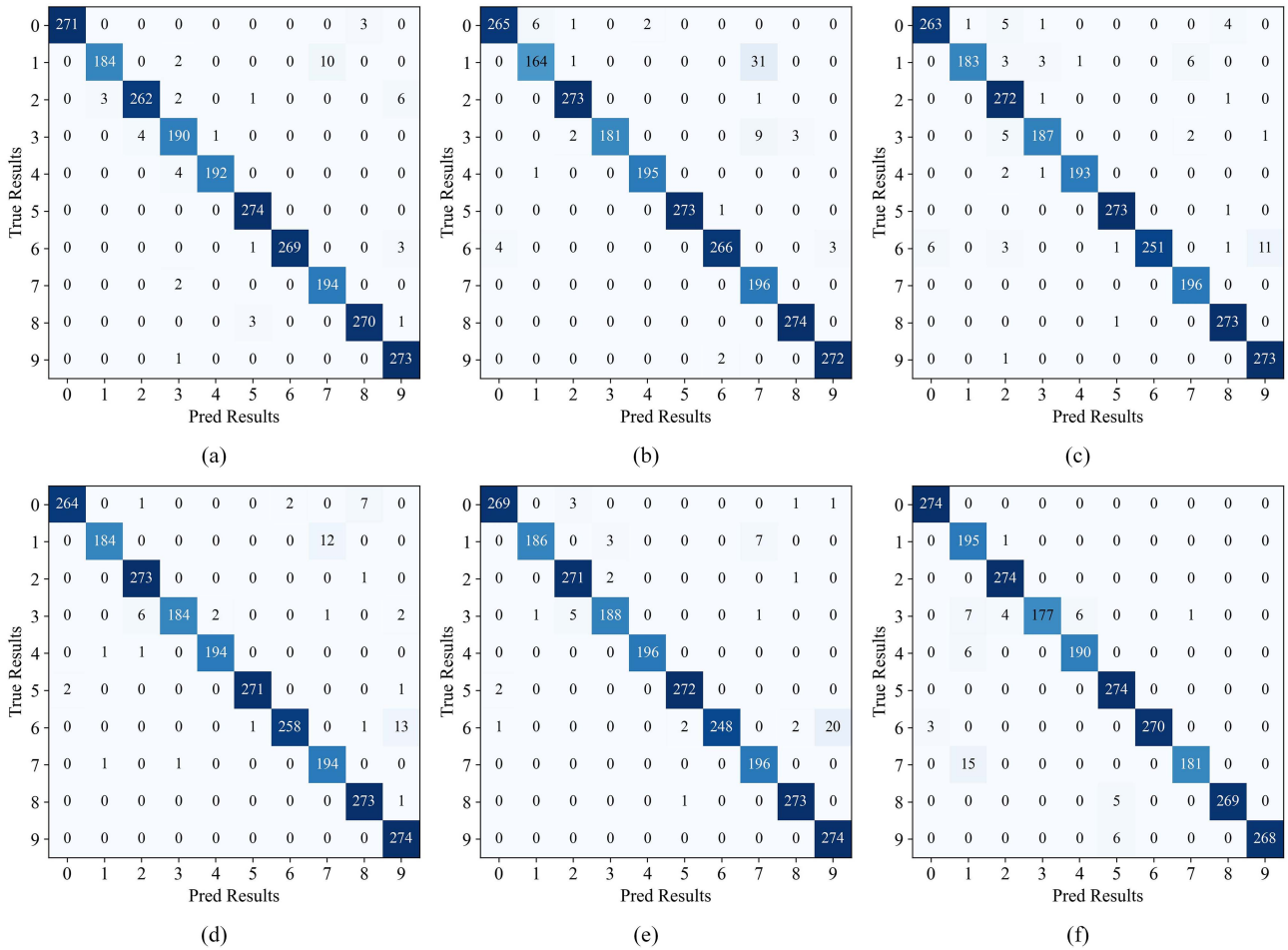


Fig. 9. Confusion matrix of the DNN models on the MSTAR dataset. (a) DenseNet121. (b) GoogLeNet. (c) InceptionV3. (d) MobileNet. (e) ResNet50. (f) ShuffleNet. Numbers from 0 to 9 indicate, respectively, the following classes: 2S1, BMP2, BRDM2, BTR60, BTR70, D7, T62, T72, ZIL131, and ZSU234.

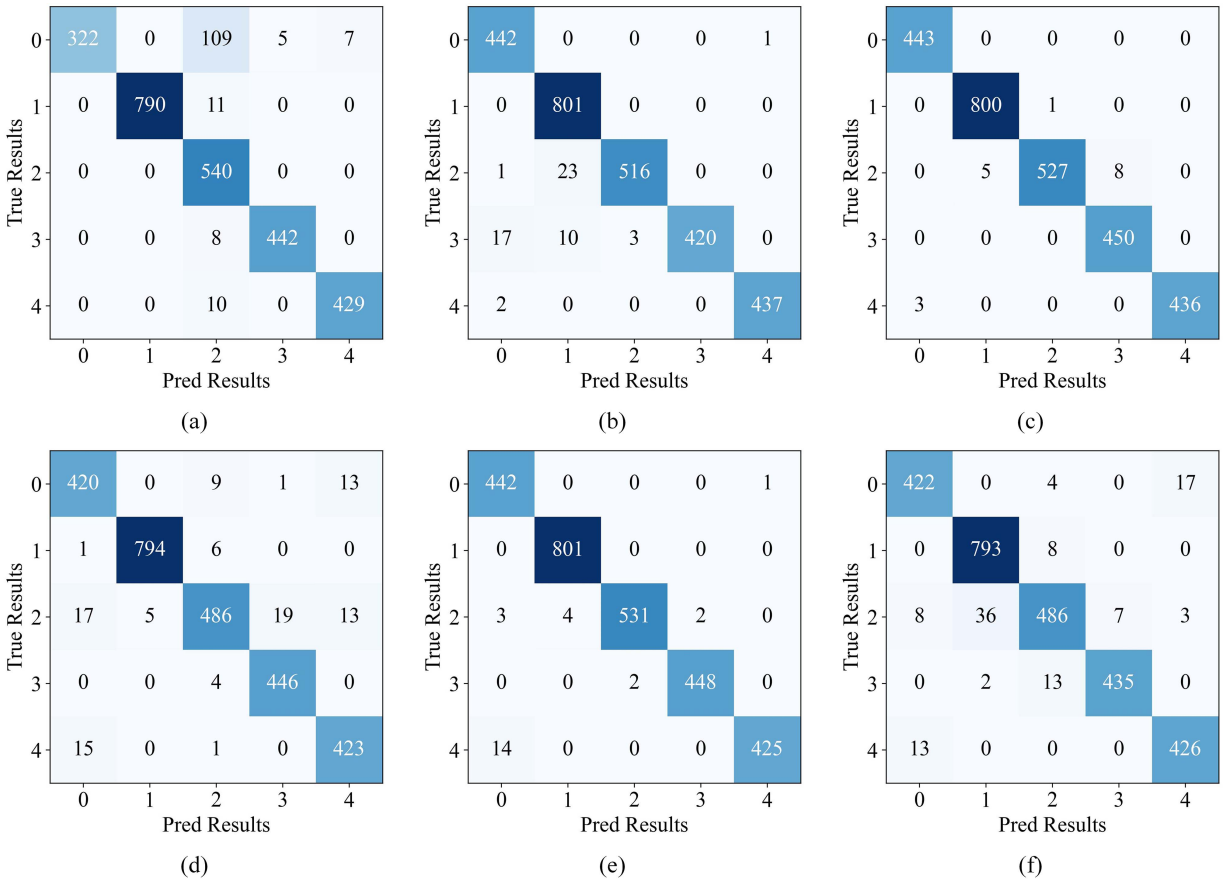


Fig. 10. Confusion matrix of the DNN models on the SEN1-2 dataset. (a) DenseNet121. (b) GoogLeNet. (c) InceptionV3. (d) MobileNet. (e) ResNet50. (f). ShuffleNet. Numbers from 0 to 4 indicate, respectively, the following classes: s1_106, s1_47, s1_51, s1_58, and s1_78.

C. Evaluation Metrics

This study examined the performance of the attack algorithm from two comprehensive perspectives: effectiveness and stealthiness of the attack.

The effectiveness of the attack is directly correlated with the accuracy of classification $\tilde{A}cc$ [18], [51]. For non-targeted attacks, $\tilde{A}cc$ demonstrates the likelihood that the DNN model will identify the adversarial example $\tilde{x}$ as the true category C_{tr}. For targeted attacks, $\tilde{A}cc$ demonstrates the likelihood that the DNN model will identify the adversarial example $\tilde{x}$ as the target category C_{ta}. Therefore, in non-targeted attacks, a lower $\tilde{A}cc$ indicates a reduced probability of the DNN model correctly identifying adversarial examples, signifying a more effective attack; in targeted attacks, a higher $\tilde{A}cc$ indicates a higher probability of the DNN model recognizing adversarial examples as the specified category C_{ta}; thus, the attack is more effective. In summary, the effectiveness of non-targeted attacks is inversely proportional to $\tilde{A}cc$, while the effectiveness of targeted attacks is directly proportional to $\tilde{A}cc$. The formula for $\tilde{A}cc$ is as follows:

$$\tilde{A}cc = \begin{cases} \sum_{n=1}^{N} \frac{Z(\arg\max_i (f_s(\tilde{x}_n)_i) == C_{tr})}{N} & \text{non-targeted attack} \\ \sum_{C_{ta}=1}^{K} \sum_{n=1}^{N} \frac{Z(\arg\max_i (f_s(\tilde{x}_n)_i) == C_{ta})}{K \times N} & \text{targeted attack} \end{cases} \tag{20}$$

where K is the number of target classes and $Z(\cdot)$ is a discriminant function.

In addition, there are three indicators similar to $\tilde{A}cc$: Acc, $Acc^{\#}$, and $\tilde{A}cc^{\#}$. Acc denotes the identification accuracy of the DNN model for the original example x, $Acc^{\#}$ denotes the identification accuracy of the DNN model for the attenuated original example $x^{\#}$, and $\tilde{A}cc^{\#}$ denotes the identification accuracy of the DNN model for the attenuated adversarial example $\tilde{x}^{\#}$. In particular, the value $\tilde{A}cc^{\#}$ indirectly reflects the transferability performance.

The second factor is the stealthiness of the attack. As indicated in (21), we employed $L_p{-}norm$ to quantify the extent of distortion present in both the input examples and adversarial examples

$$\begin{cases} \tilde{L}_P = \frac{1}{N} \sum_{n=1}^{N} \|\tilde{x}_n - x_n\|_p & \text{for the generator} \\ {L_p}^{\#} = \frac{1}{N} \sum_{n=1}^{N} \|{x^{\#}}_n - x_n\|_p & \text{for the attenuator} \end{cases} \tag{21}$$

where $\tilde{L}_p$ and ${L_p}^{\#}$ are the extent of image distortion induced by the adversarial example subsequent to its passage through the generator and the attenuator, respectively. The smaller the $\tilde{L}_p$ and ${L_p}^{\#}$, the better the stealthiness of the adversarial example.

TABLE III
NON-TARGETED ATTACK RESULTS OF THE TUAN ON THE MSTAR AND SEN1-2 DATASETS

Dataset	Surrogate	Acc	$\tilde{A}cc$	$Acc^{\#}$	$\tilde{A}cc^{\#}$	$\tilde{L}_2$	$L_2^{\#}$
MSTAR	DenseNet121	98.06%	10.88%	95.26%	65.99%	4.580	2.499
	GoogLeNet	97.24%	8.74%	93.32%	60.55%	4.070	2.354
	InceptionV3	97.44%	10.35%	92.62%	53.62%	3.375	2.765
	MobileNet	97.98%	5.44%	93.62%	36.07%	4.769	2.288
	ResNet50	97.82%	13.81%	96.74%	28.28%	4.879	1.185
	ShuffleNet	97.77%	14.72%	92.66%	32.56%	4.270	3.051
	Mean	**97.72%**	**10.66%**	**94.04%**	**46.18%**	**4.323**	**2.357**
SEN1-2	DenseNet121	94.39%	24.65%	90.53%	41.38%	5.134	3.124
	GoogLeNet	97.87%	26.49%	95.44%	48.75%	5.955	3.133
	InceptionV3	99.36%	23.72%	96.30%	44.26%	4.721	3.904
	MobileNet	98.99%	21.10%	91.77%	44.15%	5.683	2.202
	ResNet50	99.03%	13.66%	90.42%	64.38%	7.208	1.886
	ShuffleNet	95.85%	17.55%	87.83%	50.09%	7.294	4.972
	Mean	**97.58%**	**21.20%**	**92.05%**	**48.84%**	**5.999**	**3.204**

TABLE IV
TARGETED ATTACK RESULTS OF THE TUAN ON THE MSTAR AND SEN1-2 DATASETS

Dataset	Surrogate	Acc	$\tilde{A}cc$	$Acc^{\#}$	$\tilde{A}cc^{\#}$	$\tilde{L}_2$	$L_2^{\#}$
MSTAR	DenseNet121	10.00%	94.51%	91.40%	78.92%	4.210	2.632
	GoogLeNet	10.00%	95.26%	95.30%	85.20%	4.635	1.585
	InceptionV3	10.00%	98.19%	94.27%	87.72%	4.111	2.444
	MobileNet	10.00%	98.56%	94.15%	82.36%	4.037	2.479
	ResNet50	10.00%	99.01%	97.77%	90.40%	4.737	2.155
	ShuffleNet	10.00%	99.51%	97.16%	86.44%	4.283	1.943
	Mean	**10.00%**	**97.51%**	**95.01%**	**85.17%**	**4.336**	**2.206**
SEN1-2	DenseNet121	20.00%	94.95%	89.14%	67.86%	5.399	3.124
	GoogLeNet	20.00%	92.24%	95.65%	64.82%	5.369	3.133
	InceptionV3	20.00%	93.64%	97.09%	60.89%	5.154	2.361
	MobileNet	20.00%	93.12%	92.72%	43.21%	4.888	3.833
	ResNet50	20.00%	93.78%	90.42%	64.83%	7.208	1.886
	ShuffleNet	20.00%	90.96%	88.43%	44.24%	6.652	4.575
	Mean	**20.00%**	**93.12%**	**92.24%**	**57.64%**	**5.778**	**3.152**

D. Comparison of Attack Performance

This section presents an evaluation of the proposed algorithm using the MSTAR and SEN1-2 datasets. In the training phase, the algorithm in this study considered six DNN models as surrogate models and evaluated the index parameters in Section III-C after each round of training. The outcomes are displayed in Tables III and IV.

In non-targeted attacks, the identification accuracies of the six DNN models on the MSTAR and SEN1-2 datasets were mostly greater than 95%. However, on the MSTAR dataset, the identification accuracy $\tilde{A}cc$ was below 15%, and $\tilde{L}_2$ was below 5. For the SEN1-2 dataset, the identification accuracy $\tilde{A}cc$ was below 26.5%, and $\tilde{L}_2$ was below 7.5. Based on the integration of these two points, it can be inferred that the proposed algorithm can generate adversarial examples that can successfully exploit the vulnerabilities of the DNN model without being detectable by human visual perception. The attenuator performance was simultaneously evaluated during the TUAN training phase. In the MSTAR dataset, the average values of $Acc^{\#}$ and $L_2^{\#}$ for the six DNN models were 94.04% and 2.357, respectively. In the SEN1-2 dataset, the average values of $Acc^{\#}$ and $L_2^{\#}$ for the six DNN models were 92.05% and 3.204, respectively. Based on the findings above, it can be concluded that the attenuator effectively preserves a high level of identification accuracy for the original input examples while minimizing the occurrence of significant image distortions, that is, the attenuator can preserve most of the semantic information of the original input examples. $\tilde{A}cc^{\#}$ denotes the identification accuracy of the DNN model for the attenuated adversarial example $\tilde{x}^{\#}$, which indirectly reflects the transferability of adversarial examples. The average $\tilde{A}cc^{\#}$ value was 46.18% in the MSTAR dataset; this indicates that the attenuator increased the identification accuracy of the adversarial examples on the DNN model by an average of 35.52%. For the SEN1-2 dataset, the average $\tilde{A}cc^{\#}$ value was 48.84%, that is, the attenuator improved the identification accuracy of the adversarial examples on the DNN model by an

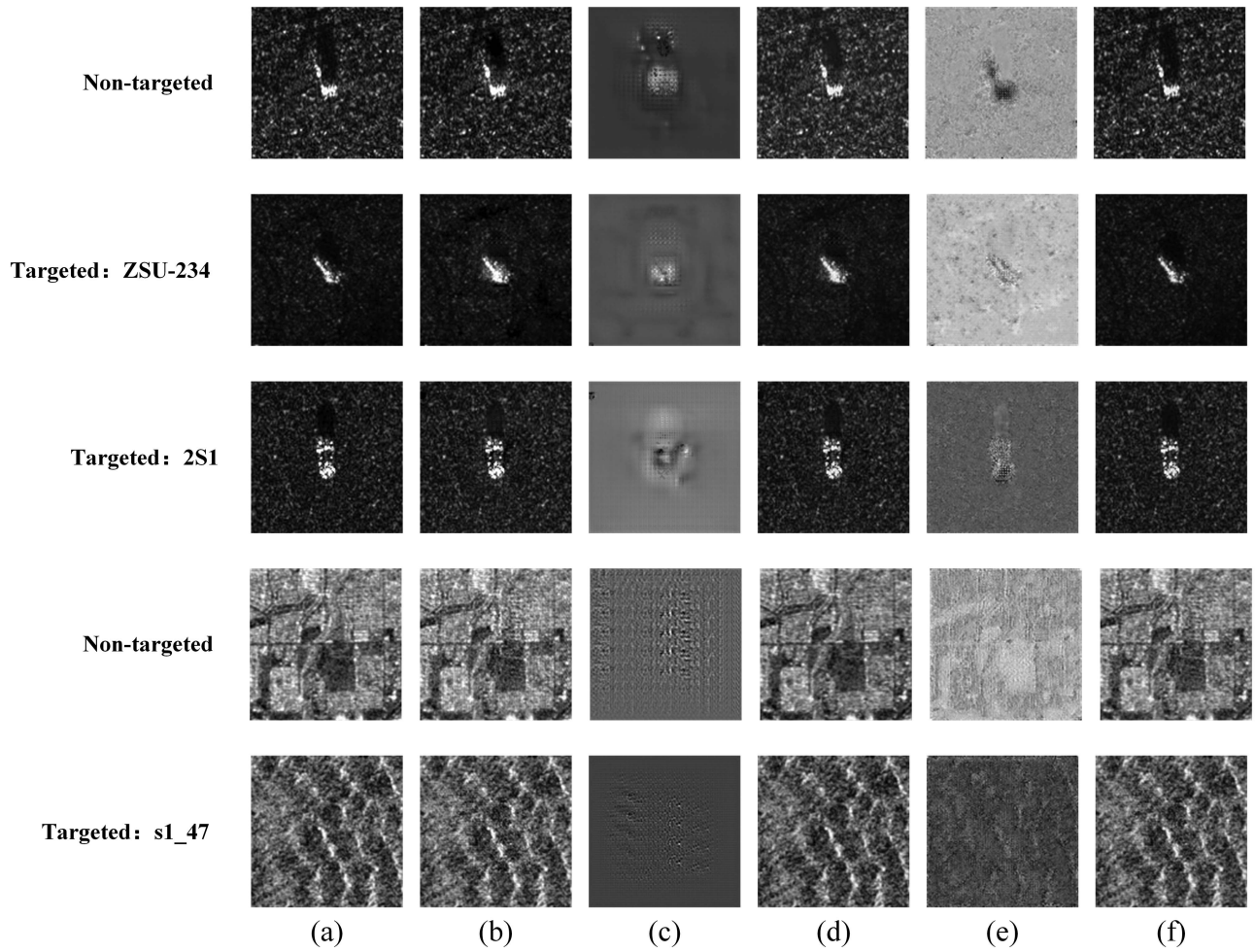


Fig. 11. Visualization of attack results against ShuffleNet. (a) Original examples. (b) Adversarial examples. (c) Adversarial perturbations. (d) Attenuated examples. (e) Deformation distortion. (f) Attenuated adversarial examples.

average of 27.64%. Hence, the attenuator can effectively reduce the attack potency of adversarial examples.

In targeted attacks, Acc represents the probability that the DNN model identifies the original input example as the target class, so it can reflect the distribution of the dataset, that is, each category on the MSTAR dataset accounted for about 10% and each category on the SEN1-2 dataset accounted for about 20%. $\tilde{A}cc$ is used to measure the probability that each DNN model recognizes the adversarial example as the target class, and $\tilde{L}_2$ indicates the degree of image distortion caused by the adversarial example. For the MSTAR dataset, the average $\tilde{A}cc$ for all the DNN models was 97.51%, and average $\tilde{L}_2$ was 4.336. On the SEN1-2 dataset, the average $\tilde{A}cc$ of all the DNN models was 93.12%, and average $\tilde{L}_2$ was 5.778. The generator in the TUAN can construct adversarial examples that effectively prompt the DNN model to classify them as the intended target class. Importantly, these adversarial examples do not result in significant image distortion. Like non-targeted attacks, in targeted attacks, the performance of attenuators was evaluated during the training process of the TUAN. For the MSTAR dataset, the mean$Acc^{\#}$, $\tilde{A}cc^{\#}$, and $\tilde{L}_2$ were 95.01%, 85.17%, and 2.206, respectively. For the SEN1-2 dataset, the mean$Acc^{\#}$, $\tilde{A}cc^{\#}$, and $\tilde{L}_2$ were 92.24%, 57.64%, and 3.152, respectively. These results demonstrate that the attenuator can effectively reduce the attack performance of adversarial examples by introducing minimal image distortion. In addition, it successfully maintains the semantic information of the input examples.

In summary, the adversarial examples constructed by the algorithm generator in this article effectively deceived the DNN model. Simultaneously, the attenuator utilized minor deformation to diminish the effectiveness of adversarial examples in terms of attack performance while maintaining the semantic information of the original examples. To ensure that the generator is better than the attenuator, this study used r to adjust the training ratio between the generator and the attenuator. Furthermore, Fig. 11 presents the visualization of the attack result graph of the TUAN, utilizing ShuffleNet as a surrogate model.

This study compared the proposed algorithm with state-of-the-art UAP algorithms on the DNN-based SAR-ATR model using the MSTAR and SEN1-2 datasets. The results are shown in Tables V and VI. It is evident that the attack performance of the proposed algorithm under the six DNN models was better than that of the other three algorithms when the degree of image distortion was comparable.

TABLE V
ATTACK PERFORMANCE OF THE TUAN (OURS), UAN [11], U-NET, RESGENERATOR [50], NAG [14], GD-UAP [13], DEEPFOOL-UAP [16], AND SAR-UAP [19] ON DNN-BASED SAR-ATR MODELS ON THE MSTAR DATASET

Method	Surrogate	Non-Targeted		Targeted	
		$\tilde{A}cc$	$\tilde{L}_2$	$\tilde{A}cc$	$\tilde{L}_2$
TUAN	DenseNet121	10.88%	4.821	94.51%	4.795
	GoogLeNet	8.74%	3.959	95.26%	4.635
	InceptionV3	10.35%	3.375	98.19%	4.111
	MobileNet	5.44%	4.181	98.56%	4.037
	ResNet50	13.81%	4.879	99.01%	4.737
	ShuffleNet	14.72%	4.270	99.51%	4.283
	Mean	**10.66%**	**4.248**	**97.51%**	**4.433**
UAN	DenseNet121	16.61%	4.582	94.39%	4.738
	GoogLeNet	19.79%	4.782	91.14%	4.469
	InceptionV3	14.63%	4.096	94.35%	4.343
	MobileNet	14.51%	4.898	96.99%	4.331
	ResNet50	15.29%	4.427	96.63%	4.465
	ShuffleNet	20.73%	4.031	88.05%	4.317
	Mean	**16.93%**	**4.469**	**93.59%**	**4.443**
U-Net	DenseNet121	11.91%	3.733	85.11%	3.928
	GoogLeNet	16.90%	1.825	81.97%	4.014
	InceptionV3	13.81%	2.656	82.86%	3.790
	MobileNet	15.79%	3.321	89.61%	4.318
	ResNet50	12.16%	3.572	89.41%	4.073
	ShuffleNet	20.82%	3.048	80.63%	3.471
	Mean	**15.23%**	**3.026**	**84.93%**	**3.932**
ResG	DenseNet121	27.62%	4.704	79.92%	4.523
	GoogLeNet	35.99%	4.764	86.27%	4.024
	InceptionV3	17.68%	4.423	83.10%	4.193
	MobileNet	23.12%	4.457	87.55%	4.606
	ResNet50	21.68%	4.668	88.17%	4.938
	ShuffleNet	33.14%	3.253	82.36%	4.838
	Mean	**26.54%**	**4.378**	**84.56%**	**4.520**
NAG	DenseNet121	20.12%	4.828	75.75%	5.444
	GoogLeNet	34.30%	4.140	77.40%	5.671
	InceptionV3	28.19%	5.123	81.61%	4.882
	MobileNet	14.51%	4.811	82.59%	5.421
	ResNet50	20.03%	4.760	75.12%	5.603
	ShuffleNet	26.55%	4.876	80.63%	4.850
	Mean	**23.95%**	**4.756**	**78.85%**	**5.312**
GD-UAP	DenseNet121	10.98%	4.157	86.06%	3.985
	GoogLeNet	16.65%	4.158	88.21%	3.987
	InceptionV3	11.90%	4.145	90.80%	3.967
	MobileNet	9.35%	4.065	89.98%	3.918
	ResNet50	13.03%	4.119	76.79%	3.964
	ShuffleNet	14.79%	4.149	87.26%	3.987
	Mean	**12.78%**	**4.132**	**86.52%**	**3.968**
DeepFool-UAP	DenseNet121	12.02%	3.385	--	--
	GoogLeNet	10.30%	4.332	--	--
	InceptionV3	11.13%	3.718	--	--
	MobileNet	10.63%	3.963	--	--
	ResNet50	15.43%	3.543	--	--
	ShuffleNet	17.06%	4.525	--	--
	Mean	**12.76%**	**3.911**	--	--
SAR-UAP	DenseNet121	14.84%	4.728	72.84%	4.401
	GoogLeNet	12.00%	4.632	91.80%	6.120
	InceptionV3	8.33%	4.623	76.09%	4.331
	MobileNet	12.70%	4.296	98.89%	5.479
	ResNet50	15.65%	4.458	94.48%	6.420
	ShuffleNet	14.18%	4.678	78.11%	3.720
	Mean	**12.95%**	**4.569**	**85.37%**	**5.079**

TABLE VI
ATTACK PERFORMANCE OF THE TUAN (OURS), UAN [11], U-NET, RESGENERATOR [50], NAG [14], GD-UAP [13], DEEPFOOL-UAP [16], AND SAR-UAP [19] ON DNN-BASED SAR-ATR MODELS ON THE SEN1-2 DATASET

Method	Surrogate	Non-Targeted		Targeted	
		$\tilde{A}cc$	$\tilde{L}_2$	$\tilde{A}cc$	$\tilde{L}_2$
TUAN	DenseNet121	24.65%	5.134	94.95%	5.399
	GoogLeNet	26.49%	5.955	92.24%	5.268
	InceptionV3	23.72%	4.721	93.64%	4.854
	Mobilenet	13.85%	4.355	97.38%	5.143
	ResNet50	13.66%	4.208	93.78%	5.114
	Shufflenet	17.55%	4.294	90.96%	4.852
	Mean	**19.99%**	**4.778**	**93.83%**	**5.105**
UAN	DenseNet121	25.48%	3.335	83.60%	4.793
	GoogLeNet	27.27%	4.006	81.72%	4.621
	InceptionV3	31.72%	6.176	87.79%	4.853
	Mobilenet	32.70%	5.915	93.15%	6.776
	ResNet50	20.31%	5.611	85.88%	5.072
	Shufflenet	18.22%	5.968	94.97%	5.673
	Mean	**25.95%**	**5.169**	**87.85%**	**5.298**
U-Net	DenseNet121	29.31%	4.319	78.94%	5.511
	GoogLeNet	27.65%	4.982	82.29%	5.697
	InceptionV3	31.01%	5.115	84.65%	5.600
	Mobilenet	25.24%	4.652	84.37%	4.458
	ResNet50	31.43%	4.116	84.46%	4.826
	Shufflenet	31.64%	4.636	88.96%	4.823
	Mean	**29.38%**	**4.637**	**83.95%**	**5.153**
ResG	DenseNet121	26.84%	4.636	86.64%	4.968
	GoogLeNet	36.82%	4.259	84.06%	4.452
	InceptionV3	28.32%	4.324	85.30%	4.979
	Mobilenet	30.19%	5.422	86.57%	4.398
	ResNet50	41.23%	5.423	84.96%	5.403
	Shufflenet	33.45%	5.144	89.75%	5.898
	Mean	**32.81%**	**4.868**	**86.21%**	**5.016**
NAG	DenseNet121	25.06%	4.828	87.13%	4.958
	GoogLeNet	41.53%	3.862	86.37%	3.524
	InceptionV3	32.25%	4.375	82.34%	4.262
	Mobilenet	26.86%	3.597	78.98%	5.437
	ResNet50	32.36%	6.171	86.59%	4.891
	Shufflenet	19.87%	4.860	75.27%	5.113
	Mean	**29.66%**	**4.616**	**82.78%**	**4.698**
GD-UAP	DenseNet121	35.98%	3.836	71.53%	5.547
	GoogLeNet	30.69%	4.652	67.76%	5.502
	InceptionV3	18.93%	4.683	79.98%	5.572
	Mobilenet	24.54%	4.730	77.96%	5.621
	ResNet50	17.21%	4.820	81.93%	4.688
	Shufflenet	36.78%	4.648	72.20%	5.565
	Mean	**27.36%**	**4.562**	**75.23%**	**5.416**
Deepfool-UAP	DenseNet121	26.72%	4.232	--	--
	GoogLeNet	35.71%	4.602	--	--
	InceptionV3	28.98%	5.218	--	--
	Mobilenet	21.74%	4.913	--	--
	ResNet50	18.33%	5.380	--	--
	Shufflenet	18.55%	4.835	--	--
	Mean	**25.01%**	**4.863**	--	--
SAR-UAP	DenseNet121	25.72%	4.036	90.09%	6.169
	GoogLeNet	23.64%	4.591	81.18%	4.080
	InceptionV3	22.37%	4.636	91.32%	4.904
	Mobilenet	18.03%	4.582	72.35%	4.683
	ResNet50	17.17%	4.687	83.80%	4.530
	Shufflenet	33.00%	4.151	89.92%	4.283
	Mean	**23.32%**	**4.447**	**84.78%**	**4.775**

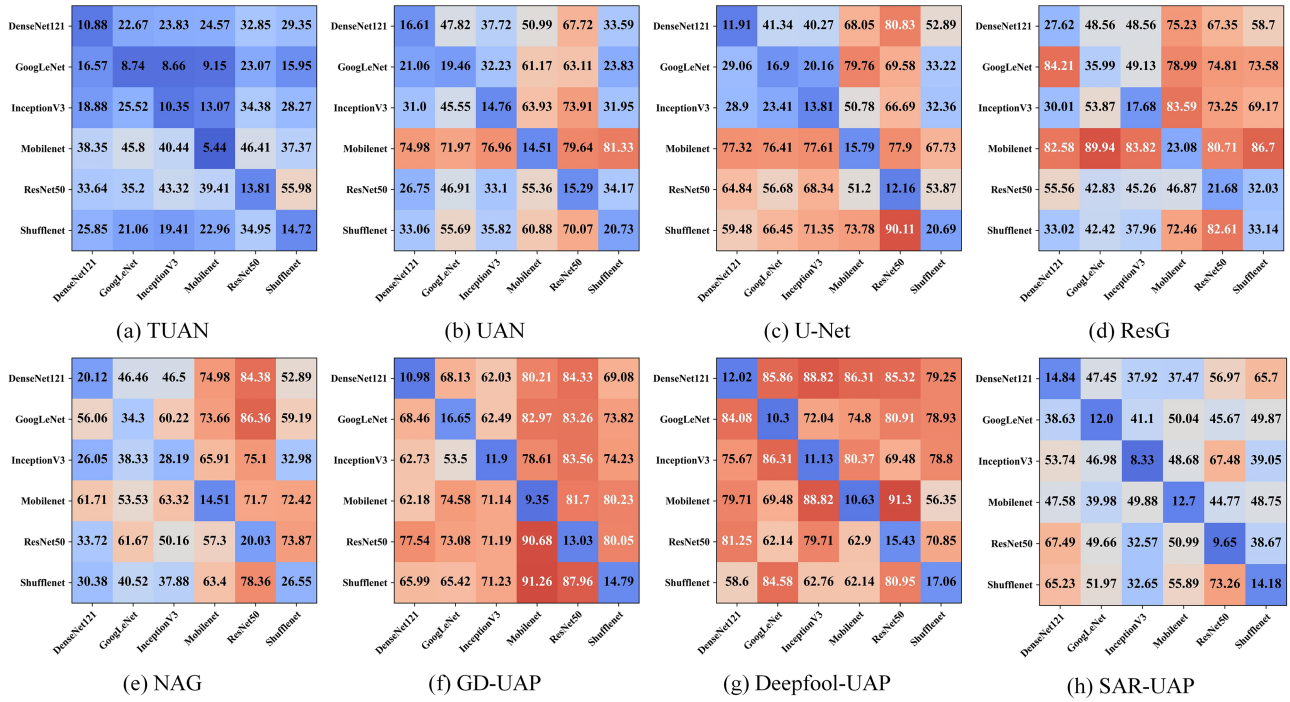


Fig. 12. Transferability of adversarial examples under non-targeted attacks on the MSTAR dataset. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) DeepFool-UAP. (h) SAR-UAP.

E. Comparative Analysis of Transferability

This section focuses on assessing the transferability of adversarial examples on the MSTAR and SEN1-2 datasets. In the experiment, all the DNN models were attacked with adversarial examples. To evaluate the transferability of adversarial examples, the identification results were assessed on different DNN models. Figs. 12 and 13 depict the non-targeted and targeted attacks conducted on the MSTAR dataset, respectively; the non-targeted and targeted attacks on the SEN1-2 dataset are shown in Figs. 14 and 15, respectively.

In non-targeted attacks, the proposed algorithm used six DNN models as surrogate models. In the MSTAR dataset, the highest identification accuracies of the victim model were 32.85%, 23.07%, 34.38%, 46.41%, 55.98%, and 34.95%. For the SEN1-2 dataset, the highest identification accuracies of the victim model were 72.53%, 78.26%, 83.09%, 71.49%, 78.38%, and 65.52%. As for baseline algorithms, on the MSTAR dataset, the highest identification accuracies of the victim model were 88.82%, 86.36%, 86.31%, 91.30%, 90.68%, and 91.26%. For the SEN1-2 dataset, the highest identification accuracies of the victim model were 94.61%, 89.03%, 90.83%, 89.71%, 89.84%, and 89.36%. By analyzing the data in Figs. 12 and 14, for each surrogate model, the adversarial examples created by the method in this study had a higher attack success rate against the six DNN models than the baseline algorithm, that is, they had the highest transferability.

In targeted attacks, similar to non-targeted attacks, the algorithms in this study used six DNN models as the surrogate models. In the MSTAR dataset, the lowest probabilities for the victim model to identify adversarial examples as the target category were 56.05%, 57.01%, 53.92%, 31.04%, 34.68%, and 49.43%. In the SEN1-2 dataset, the lowest probabilities for the victim model to identify adversarial examples as the target category were 43.73%, 31.09%, 24.73%, 30.60%, 32.17%, and 31.87%. Simultaneously, on the MSTAR dataset, the lowest probabilities for the victim model to identify adversarial examples generated by the baseline algorithm as the target category were 7.88%, 3.49%, 11.93%, 11.45%, 9.36%, and 10.18%. In the SEN1-2 dataset, the lowest probabilities of the victim model identifying adversarial examples generated by the baseline algorithm as the target category were 17.82%, 11.90%, 10.03%, 13.81%, 13.18%, and 14.75%. In contrast to the baseline algorithm, the proposed algorithm consistently identified the adversarial examples produced by the surrogate model as the target category with the highest probability, that is, in a targeted attack, the proposed algorithm had the best transferability.

F. Adversarial Attacks Under Small-Sample Conditions

The above experiments are based on the fact that an attacker can obtain all the arbitrary training images. However, due to the professionalism and confidentiality of SAR images, this is difficult to obtain SAR data. Therefore, an attacker must consider attacking the victim model with a limited number of training images.

In this section, we consider an extreme case in which the attacker can obtain only 50 samples (five for each class) in the MSTAR dataset. The 50 examples were randomly selected from the complete training dataset. Tables VII and VIII show the non-targeted and targeted attack results of the eight attack algorithms for different scale datasets, respectively. The results

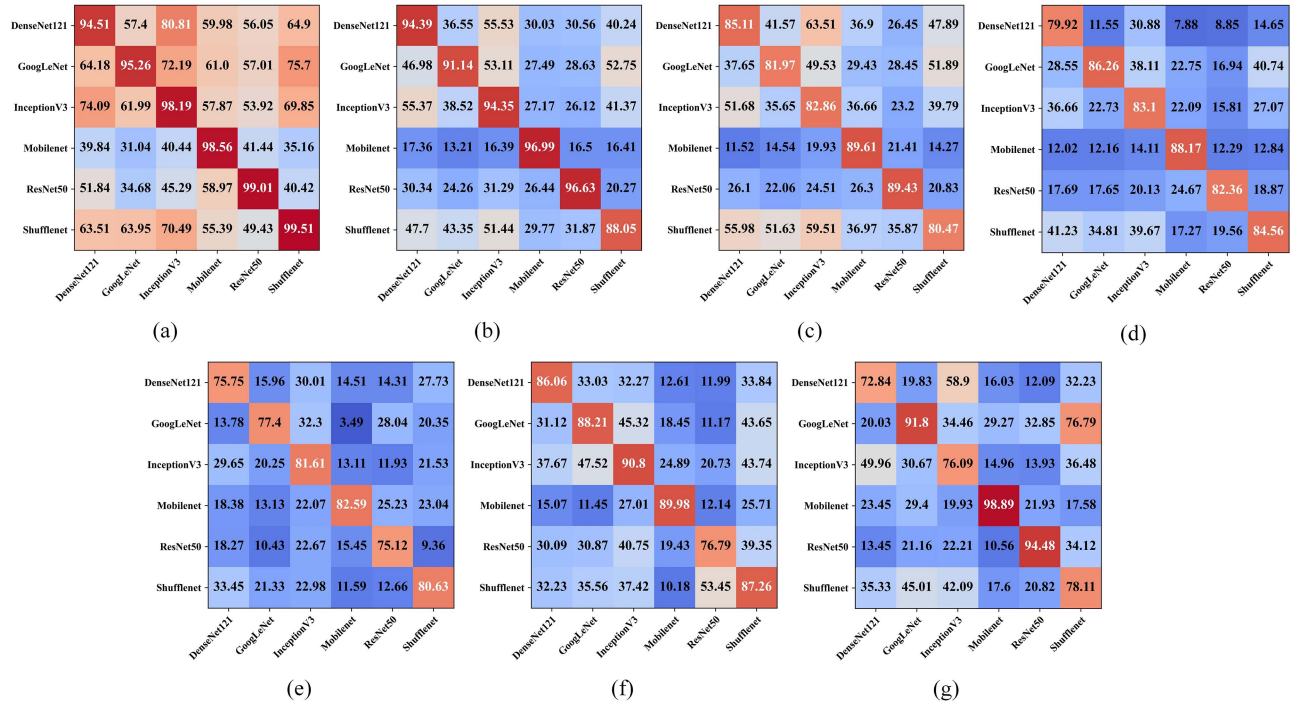


Fig. 13 Transferability of adversarial examples under targeted attacks on the MSTAR dataset. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) SAR-UAP.

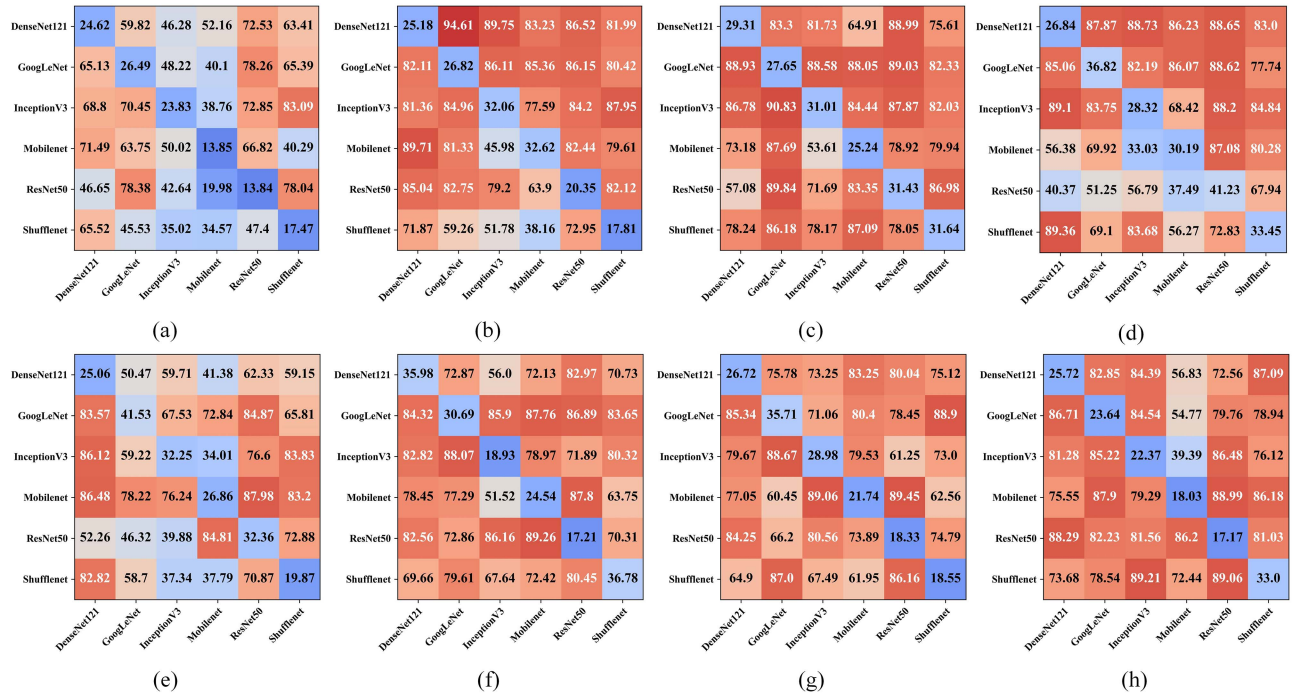


Fig. 14. Transferability of adversarial examples under non-targeted attacks on the SEN1-2 dataset. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) DeepFool-UAP. (h) SAR-UAP.

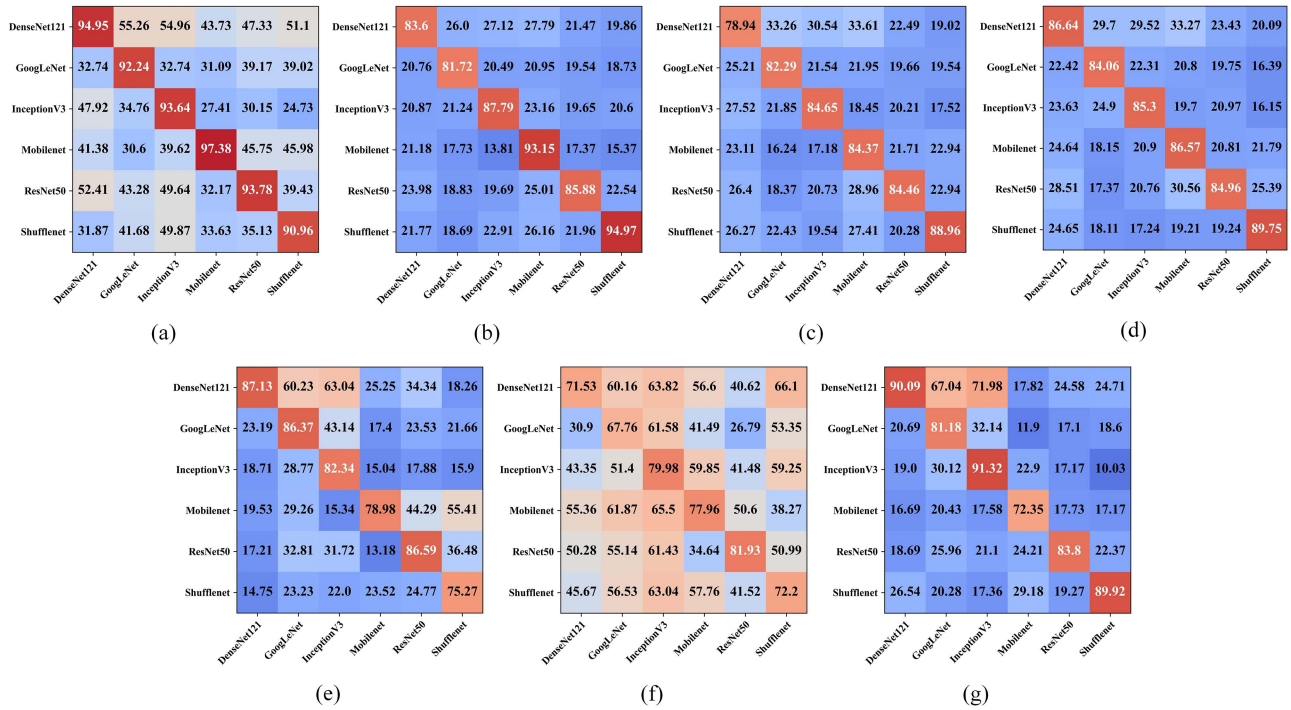


Fig. 15. Transferability of adversarial examples under targeted attacks on the SEN1-2 dataset. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) SAR-UAP.

TABLE VII
NON-TARGETED ATTACK RESULTS

Method	Dataset	Acc	$\tilde{A}cc$	$Acc^{\#}$	$\tilde{A}cc^{\#}$	$\tilde{L}_2$	$L_2^{\#}$
TUAN	Full Dataset	97.72%	10.66%	93.79%	46.18%	4.273	2.329
	Subset	97.72%	15.45%	90.02%	63.02%	4.561	3.579
	Gap	--	+4.79%	-3.77%	+16.84%	+0.288	+1.250
UAN	Full Dataset	97.72%	25.95%	--	--	5.169	--
	Subset	97.72%	35.60%	--	--	6.345	--
	Gap	--	+9.65%	--	--	+1.179	--
U-Net	Full Dataset	97.72%	15.23%	--	--	3.026	--
	Subset	97.72%	20.80%	--	--	3.641	--
	Gap	--	+5.57	--	--	+0.615	--
ResG	Full Dataset	97.72%	26.54%	--	--	4.378	--
	Subset	97.72%	39.89%	--	--	7.472	--
	Gap	--	+13.35%	--	--	+3.094	--
NAG	Full Dataset	97.72%	23.95%	--	--	4.756	--
	Subset	97.72%	37.16%	--	--	5.934	--
	Gap	--	+13.21%	--	--	+1.178	--
GD-UAP	Full Dataset	97.72%	12.78%	--	--	4.132	--
	Subset	97.72%	21.77%	--	--	5.636	--
	Gap	--	+8.99%	--	--	+1.504	--
Deepfool-UAP	Full Dataset	97.72%	12.76%	--	--	3.911	--
	Subset	97.72%	19.89%	--	--	5.683	--
	Gap	--	+7.13	--	--	+1.772	--
SAR-UAP	Full Dataset	97.72%	12.95%	--	--	4.569	--
	Subset	97.72%	24.63%	--	--	6.106	--
	Gap	--	+11.68%	--	--	+1.537	--

TABLE VIII
TARGETED ATTACK RESULTS

Method	Dataset	Acc	$\tilde{A}cc$	$Acc^{\#}$	$\tilde{A}cc^{\#}$	$\tilde{L}_2$	$L_2^{\#}$
TUAN	Full Dataset	10.00%	97.51%	95.39%	85.17%	4.361	1.121
	Subset	10.00%	85.61%	78.25%	48.61%	4.697	2.599
	Gap	--	-11.90%	-17.14%	-36.56%	+0.336	+1.478
UAN	Full Dataset	10.00%	87.85%	--	--	5.298	--
	Subset	10.00%	52.70%	--	--	6.521	--
	Gap	--	-35.15%	--	--	+1.223	--
U-Net	Full Dataset	10.00%	83.95%	--	--	5.153	--
	Subset	10.00%	74.26%	--	--	4.659	--
	Gap	--	-9.69%	--	--	-0.494	--
ResG	Full Dataset	10.00%	84.56%	--	--	4.520	--
	Subset	10.00%	44.04%	--	--	6.804	--
	Gap	--	-40.52%	--	--	+2.284	--
NAG	Full Dataset	10.00%	78.85%	--	--	5.312	--
	Subset	10.00%	60.47%	--	--	6.625	--
	Gap	--	-18.38%	--	--	+1.131	--
GD-UAP	Full Dataset	10.00%	86.52%	--	--	3.968	--
	Subset	10.00%	56.59%	--	--	5.712	--
	Gap	--	-29.93%	--	--	+1.744	--
SAR-UAP	Full Dataset	10.00%	85.37%	--	--	5.079	--
	Subset	10.00%	59.48%	--	--	6.346	--
	Gap	--	-25.89%	--	--	+1.267	--

showed that the attack performances of UAN, ResG, NAG, GD-UAP, DeepFool-UAP, and SAR-UAP on the small-sample dataset were significantly degraded; however, the adversarial examples generated by the TUAN and U-Net were not greatly affected.

The rationale behind these findings is that both the TUAN and U-Net employ the U-Net model as a generator. With the structural characteristics of the U-Net model decoder, the generator can effectively integrate the features of other network layers. Therefore, even on a small-sample dataset, it can fully learn the distribution characteristics of the data, thereby generating aggressive SAR image adversarial examples.

G. Influence of Parameters

This section evaluates and analyzes the impact of different parameter settings on TUAN attack performance on the MSTAR dataset. Specifically, Sections III-G1 and III-G2 discuss and analyze the influence of the generator $G(\cdot)$ and attenuator $A(\cdot)$ weight coefficients on attack performance. Section III-G3 evaluates the impact of different training ratios r on attack performance.

1) Loss Weight Coefficient of the Generator: As mentioned in Section III-B, in the above experiments, we set the loss weight $[w_{G1}, w_{G2}, w_{G3}]$ of the generator to$[0.25, 0.25, 0.5]$. As mentioned in Section II-B, w_{G1} and w_{G2} were used to improve the white-box attack performance and black-box attack performance of adversarial examples, respectively, that is, the ability to attack the surrogate model and the ability to attack other victim models, both of which are considered equally important in this study. Therefore, in this section, we explore the effects on the TUAN performance when $w_{G3} = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$, and 0.9 under the assumption of $w_{G1} = w_{G2}$.

Fig. 16(a) and (d) shows the attack success rate on the surrogate model, i.e., the white-box attack performance. For non-targeted attacks, $\tilde{A}cc$ increased as w_{G3} increased. For targeted attacks, $\tilde{A}cc$ decreased as w_{G3} increased. To analyze the impact of w_{G3} on transferability, Fig. 16(c) and (f) uses the adversarial examples generated in DenseNet as an example to show the black-box attack performance. Similar to the trend shown in Fig. 16(a) and (d), for non-targeted attacks, $\tilde{A}cc$ increased as w_{G3} increased. For targeted attacks, $\tilde{A}cc$ decreased as w_{G3} increased. Fig. 16(b) and (e) shows the effects of w_{G3} on $\tilde{L}_2$. For non-targeted attacks, $\tilde{L}_2$ decreased as w_{G3} increased, while for targeted attacks, $\tilde{L}_2$ decreased as w_{G3} increased.

In summary, as w_{G3} increased, the attack performance gradually weakened and the attack stealthiness gradually increased. This can be explained as when w_{G3} increased, w_{G1} and w_{G2} decreased accordingly, and the TUAN paid more attention to the stealthiness of the attack during the process of training the generator. This weakened the effectiveness of the attack. When attackers need to improve the stealth performance of adversarial examples, w_{G3} should be appropriately increased. Conversely, when attackers focus more on the effectiveness of attacks, w_{G3} should be reduced.

2) Loss Weight Coefficient of the Attenuator: This section explores the effects of different weight loss coefficients on the performance of the TUAN. As described in Section II-C, the attenuator loss weight coefficients w_{A1} and w_{A2} were used to

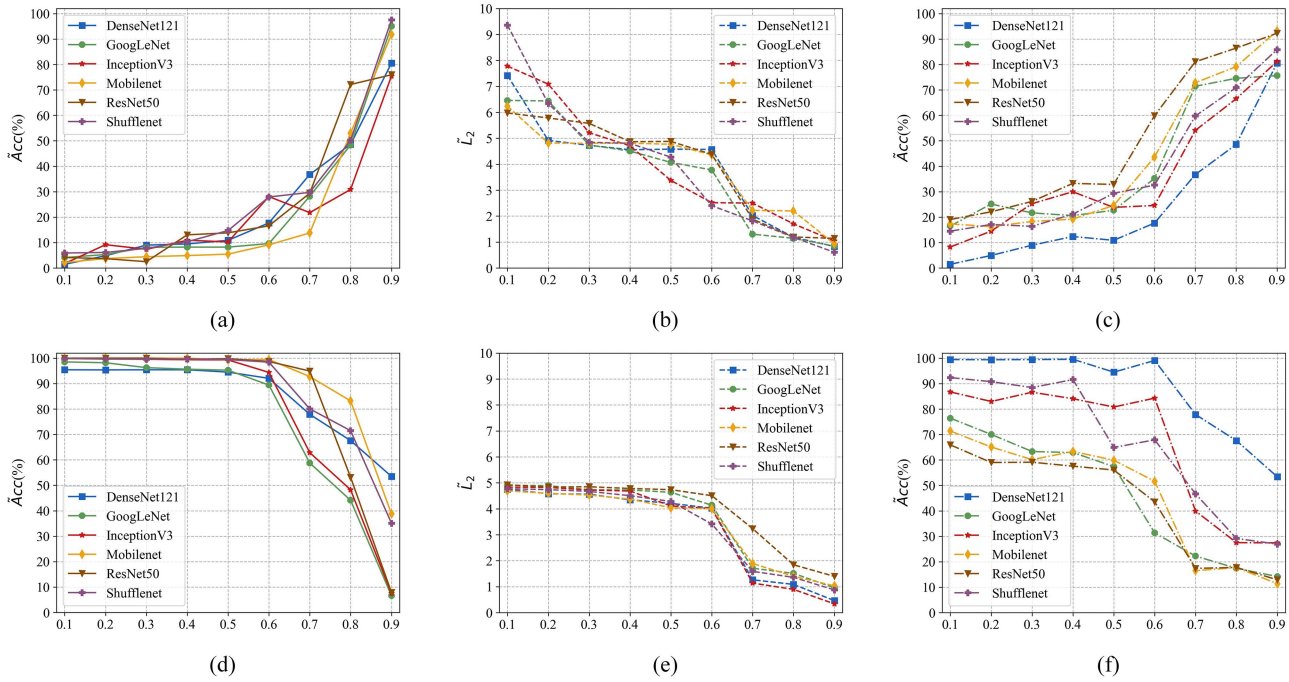


Fig. 16. Effect of the generator $G(\cdot)$ weight coefficients on TUAN performance. (a)–(c) Non-targeted attack results. (d)–(f) Targeted attack results.

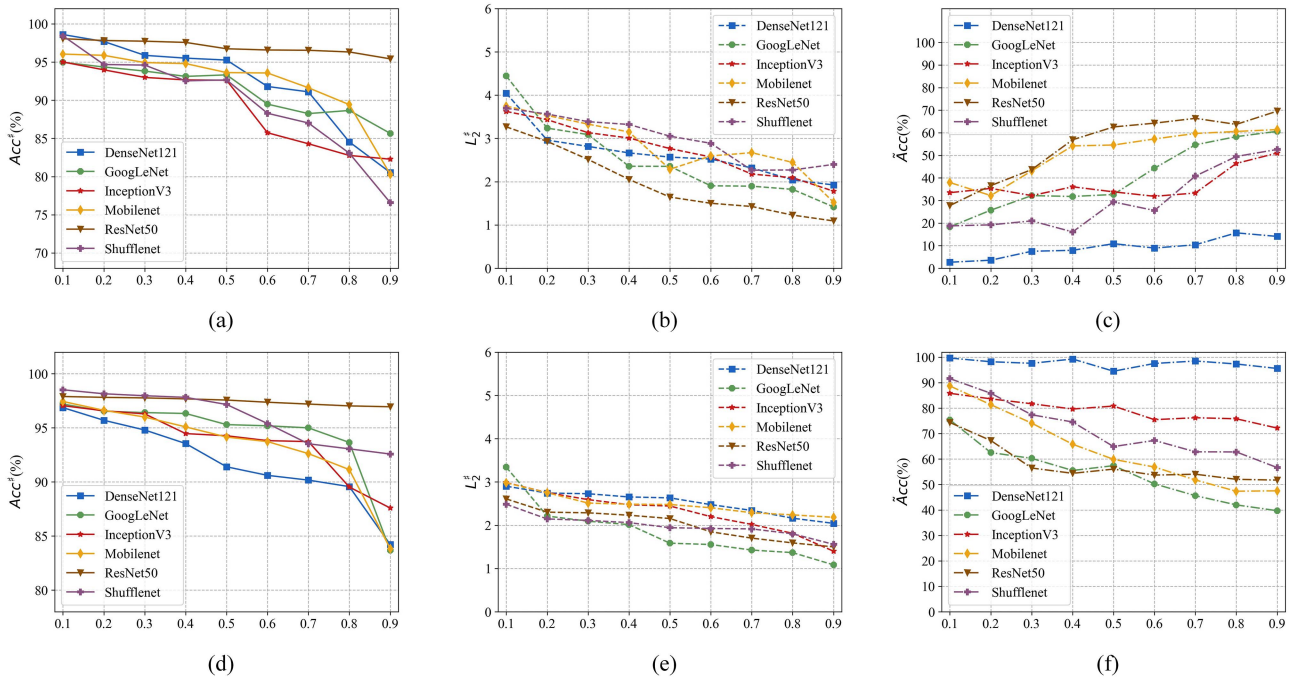


Fig. 17. Effect of the attenuator $A(\cdot)$ weight coefficient on TUAN performance. (a)–(c) Non-targeted attack results. (d)–(f) Targeted attack results.

weaken the effectiveness of $\tilde{x}$ and preserve the semantic information of x, respectively. The algorithm in this article treats both as equally important. Therefore, under the assumption$w_{A1} = w_{A2}$, this section evaluates the impact on the attack performance when w_{A3} is equal to 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. The results are shown in Fig. 17. In both the attack modes, $Acc^{\#}$ and $L_2^{\#}$ decreased as w_{A3} increased. Therefore, as w_{A3} increased, the weakening performance of the attenuator on the adversarial example $\tilde{x}$ decreased and the preserved semantic information decreased; however, the degree of deformation of the image by the attenuator decreased. To investigate the impact on the transferability of the algorithm, an analysis was conducted based on the findings presented in Fig. 17(c) and (f). The DenseNet model produced adversarial examples that attacked six models, and the transferability decreased with increasing w_{A3}.

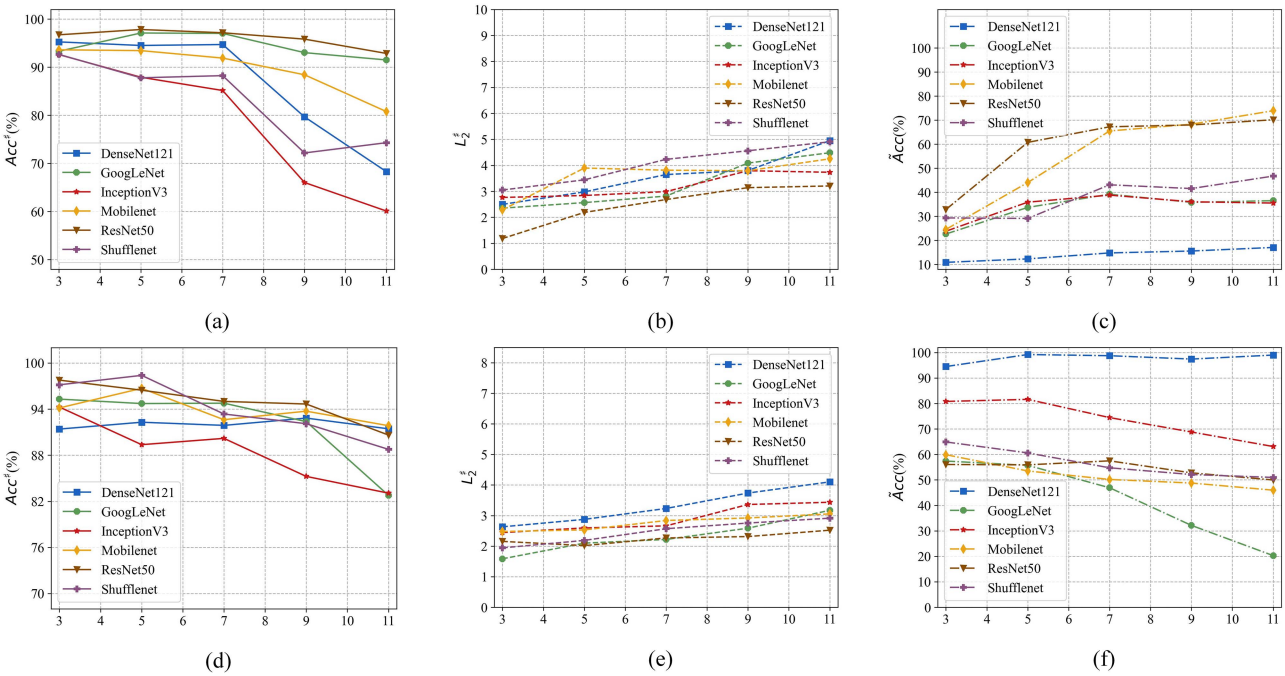


Fig. 18. Effect of training ratio r on TUAN performance. (a)–(c) Non-targeted attack results. (d)–(f) Targeted attack results.

TABLE IX
ATTACK RESULTS UNDER DIFFERENT DISTANCE METRICS

Mode	Surrogate	$\tilde{A}cc$	
		L_2 - Norm	L_∞ - Norm
Non-Targeted	DenseNet121	10.88%	22.34%
	GoogLeNet	8.74%	22.96%
	InceptionV3	10.35%	21.97%
	Mobilenet	5.44%	20.24%
	ResNet50	13.81%	28.48%
	Shufflenet	14.72%	15.99%
	Mean	**10.66%**	**22.00%**
Targeted	DenseNet121	94.51%	73.56%
	GoogLeNet	95.26%	95.09%
	InceptionV3	98.19%	95.30%
	Mobilenet	98.56%	96.21%
	ResNet50	99.01%	83.20%
	Shufflenet	99.51%	83.89%
	Mean	**97.51%**	**87.88%**

3) Training Ratio: The training ratio r was used to control the attenuator training time. This section investigates the impact of various training ratios r on TUAN performance. Fig. 18 displays the experimental findings. With an increase in the training ratio r in the two attack modes, $Acc^{\#}$ gradually decreased, $L_2^{\#}$ gradually increased, and the transfer performance gradually weakened. This is because, similar to the analysis in the previous section, when the number of generator training sets is fixed, the greater training ratios r correspond to lesser attenuator participation in the training. Therefore, when the training ratio r is excessively large, it can fail to provide sufficient training time for the attenuator to weaken the attack effectiveness of the adversarial example. When the training ratio r is too small, the combination of the attenuator and surrogate models is too powerful, and the adversarial examples constructed by the generator cannot be successfully attacked.

4) Type of L_P-Norm: The above experiments employed the L_2-norm as a metric for quantifying the level of image distortion.

In this section, we add the $L_\infty-norm$ to compare the attack effect of the TUAN. Table IX presents the attack results for the six surrogate models. To evaluate the transferability of

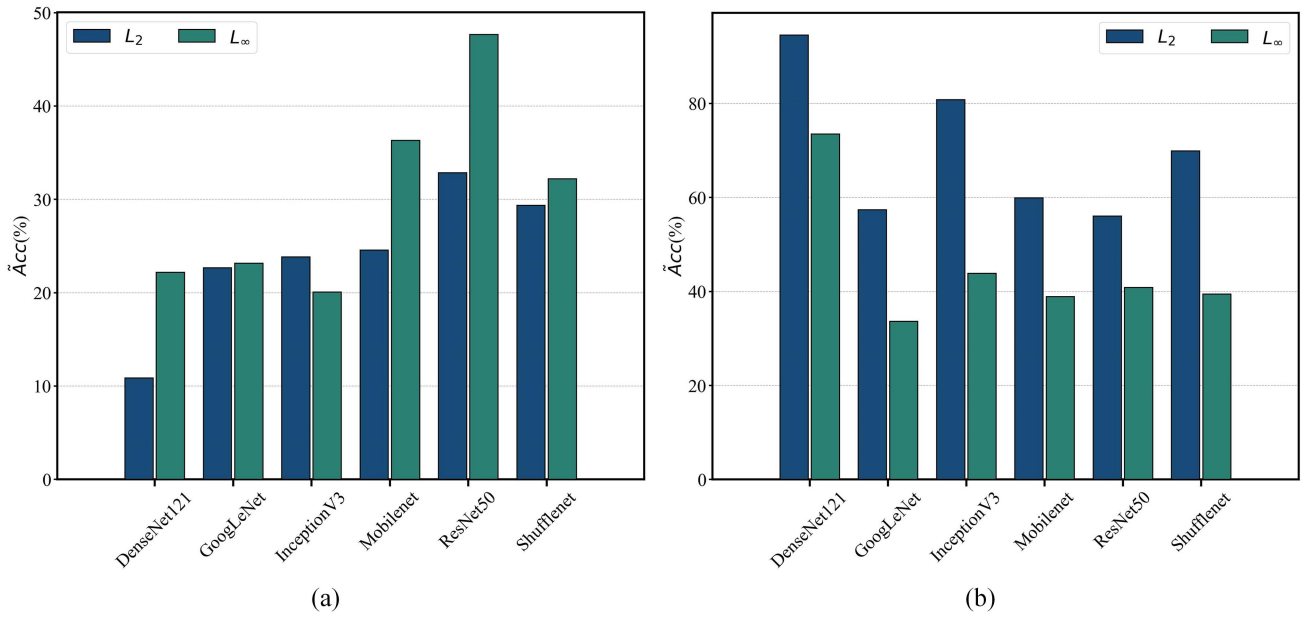


Fig. 19. Transferability under different distance metrics. (a) Non-targeted results. (b) Targeted results.

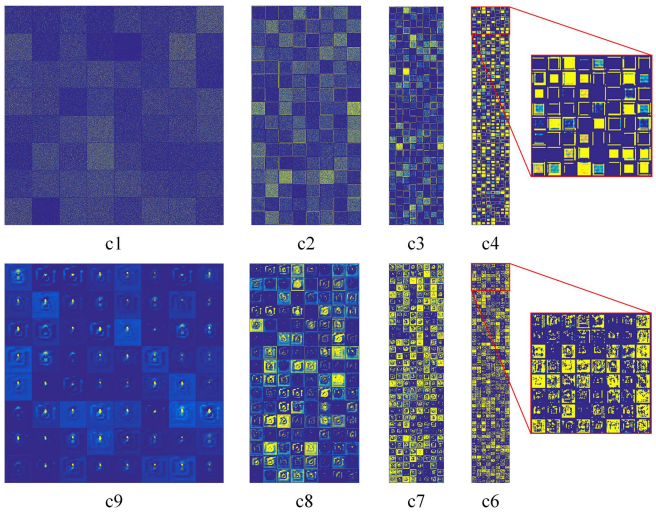


Fig. 20. Feature maps learned by the generator (U-Net).

the algorithm, Fig. 19 shows the attack results of the adversarial examples generated by DenseNet training on the other victim models. The attack effectiveness of the TUAN under the L_2-norm was more effective than that under the $L_\infty-norm$. Therefore, when an attacker trains the network model, this study recommends using the L_2-norm as a distance measure.

H. Display of the Process of Generating Adversarial Examples

To visualize how the generator $G(\cdot)$ maps from random noise to UAPs, this section presents the output feature maps of each layer of the generator on the MSTAR dataset. Based on the analysis of Fig. 20, it can be inferred that as the network layer became more proximate to the input random noise, the resulting feature map exhibited random fluctuations in pixel values; when the network layer was closer to the output layer of the decoder, the high pixel value points of the feature map output were mainly concentrated on the target area of SAR images.

I. Misclassified Category Distributions of the Non-Targeted Adversarial Attack

The existing studies have shown that on the MSTAR dataset, the white-box attack algorithm exhibits attack selectivity under non-targeted attacks [15], [52]. This section explores the distribution of incorrectly labeled categories of adversarial examples generated by DenseNet training on six models under non-targeted attacks. The results are presented in Fig. 21 . Through observation, the adversarial examples generated by DenseNet training were mainly identified as two to four categories on the other five types of victim models, which shows that the adversarial examples can lead to similar misidentification of other victim models. This phenomenon confirmed that different DNN models have similar decision boundaries [53].

J. Display of the Target Adversarial Attack

t-Distributed stochastic neighbor embedding (t-SNE) was used to map the high-dimensional features of the SAR images extracted by the DNN model in a 2-D space. As shown in Fig. 22, DenseNet was used as the surrogate model. Fig. 22(a) shows the classification results of DenseNet on the clean original MSTAR dataset, and Fig. 22(b)–(f) shows the attack result diagrams, where the specified targets are classified as 0, 2, 4, 6, and 8. In each subgraph, this section names the ten types of targets as 0–9 in sequence, and each type of target corresponds to a different color. Taking Fig. 22(c) as an example, most of the categories in

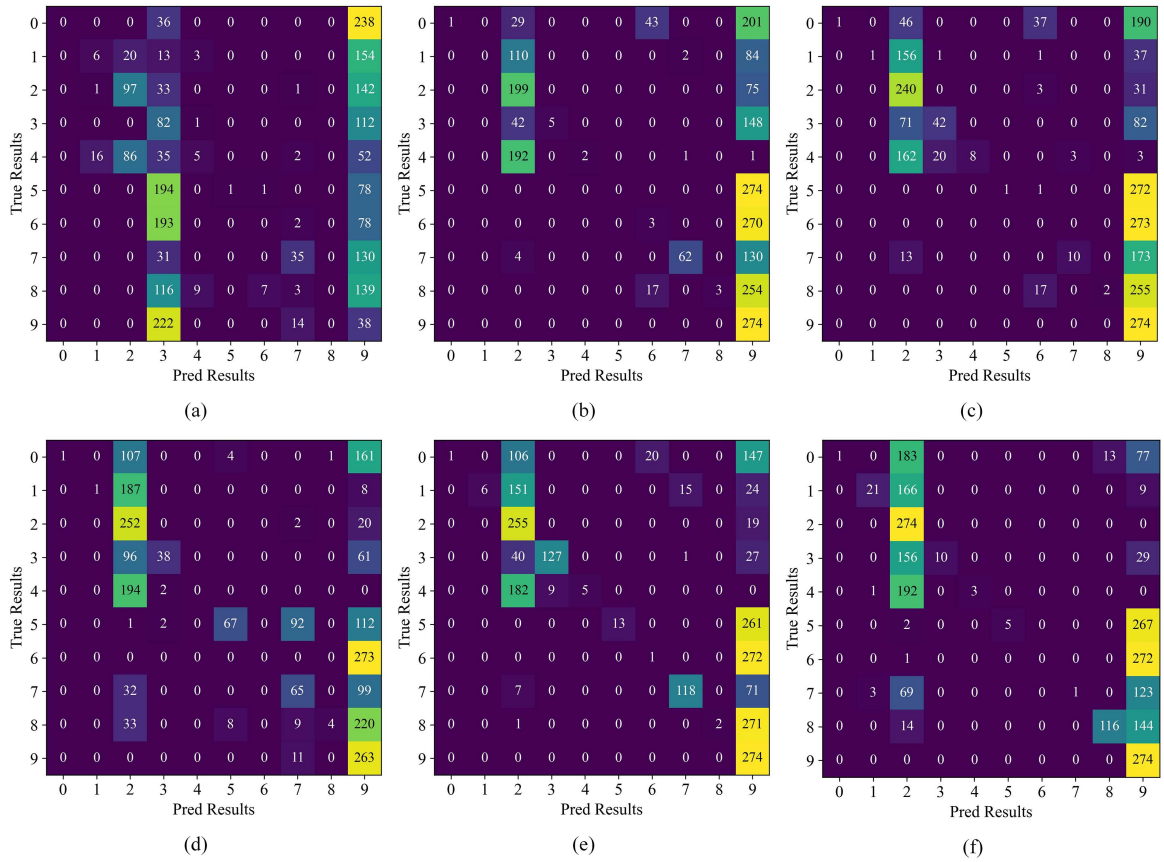


Fig. 21. Distribution of adversarial categories of the TUAN attack. (a) DenseNet121. (b) GoogLeNet. (c) InceptionV3. (d) MobileNet. (e) ResNet50. (f) ShuffleNet. Numbers from 0 to 9 indicate, respectively, the following classes: 2S1, BMP2, BRDM2, BTR60, BTR70, D7, T62, T72, ZIL131, and ZSU234.

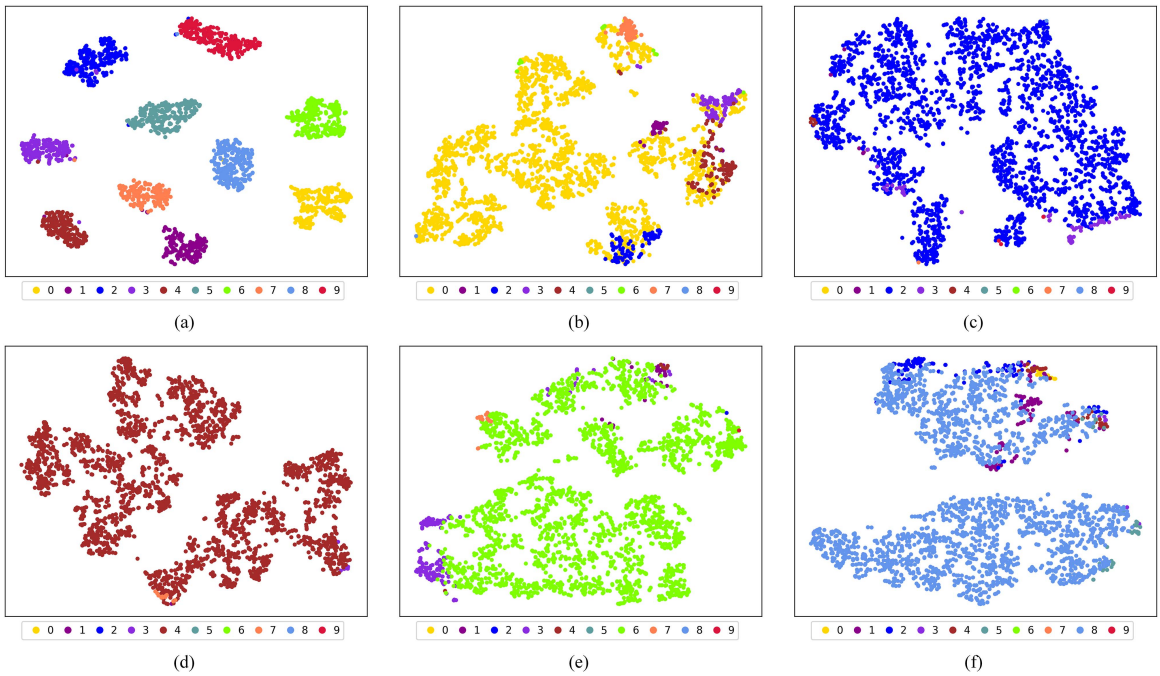


Fig. 22. t-SNE visualization. (a) Original classification results. (b)–(f) Targeted attack results. Each color denotes a category.

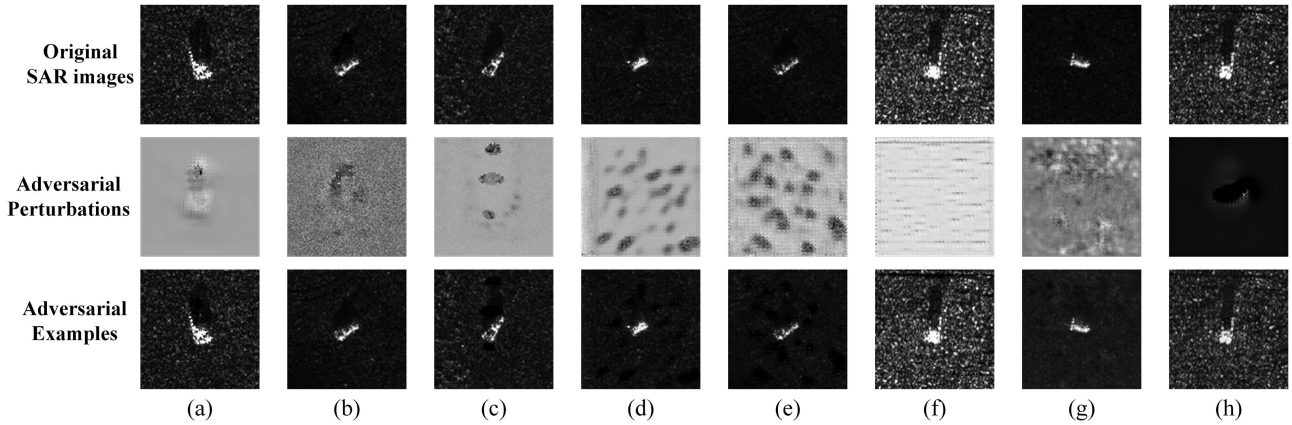


Fig. 23. Visualization of adversarial examples under non-targeted attacks. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) DeepFool-UAP. (h) SAR-UAP.

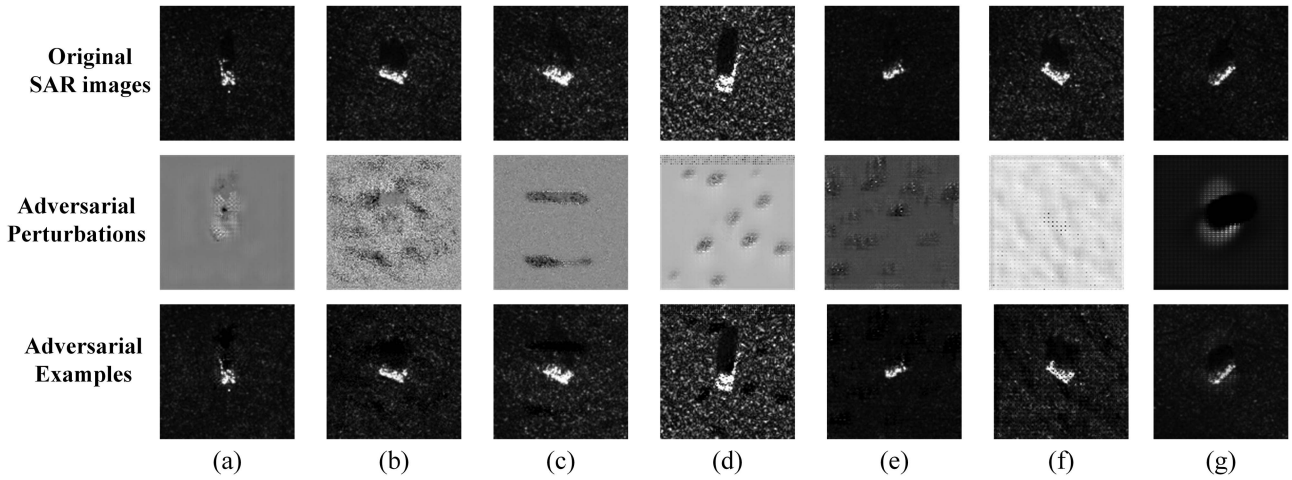


Fig. 24. Visualization of adversarial examples under targeted attack. (a) TUAN. (b) UAN. (c) U-Net. (d) ResG. (e) NAG. (f) GD-UAP. (g) SAR-UAP.

the figure were identified as specified target category 2, that is, the DNN model recognized the adversarial examples as specified categories.

K. Visualization of Adversarial Examples

DenseNet was employed as a surrogate model for the MSTAR dataset. Figs. 23 and 24 visualize adversarial examples generated by different methods. It is evident that our algorithm mostly produces adversarial perturbations that are concentrated in the target area, regardless of whether the attack is non-targeted or targeted. Nevertheless, the disturbances yielded by baseline algorithms exhibit a broad spatial distribution and mostly manifest as distinct entities.

First, the adversarial perturbations generated by the algorithm in this study were more attack specific than those generated by the baseline method, as shown in Fig. 25, which uses GradCAM++ [54] to visualize the output feature map weights of the last convolutional layer of DenseNet, where the darker region represents a greater contribution to the neural network identification process. Expectedly, when the DNN model recognizes correctly, the weight of the feature map is mainly concentrated in the target area. However, when the DNN model obtains

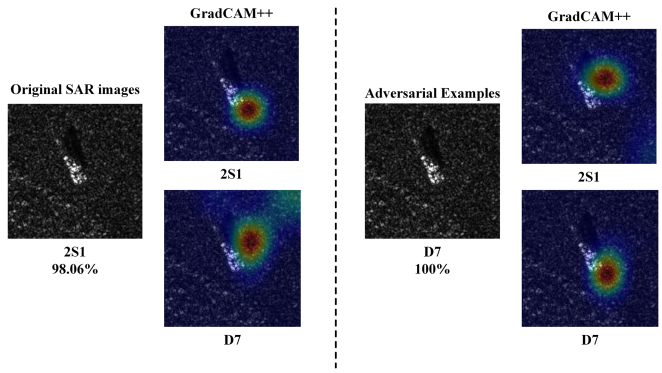


Fig. 25. Identification result map generated by GradCAM++.

wrong results in identifying adversarial examples, the weight of the feature map is mainly concentrated in the background area. Therefore, from a feature extraction standpoint, the adversarial perturbations primarily focus on the target area rather than the background clutter area. Second, from the perspective of physical realization, the adversarial disturbance generated by the algorithm is mainly concentrated in the target area; therefore, it can provide convenient conditions for the practical application deployment of SAR target identification network adversarial attacks.

V. Discussion

The in silico results presented above indicated that the proposed TUAN can effectively attack SAR images. The experimental findings presented in Section IV-E demonstrated that the suggested method outperforms the baseline approach in terms of transferability, thereby enhancing the potential for increasing the transferability of adversarial examples. This is because the TUAN uses a mutual game between the generator and the attenuator to improve the transferability of adversarial examples. Briefly, throughout the training process, the generator simulates an adversarial attack against a hard-to-attack victim model using a combination of an attack surrogate model and an attenuator. Therefore, once training is completed, the generator can yield a UAP with strong transferability through one-step mapping. In terms of small-sample attacks, the experimental results of Section IV-F showed that the TUAN can maintain good attack effectiveness and stealthiness in the absence of samples. This is because the generator U-Net has a strong data distribution learning ability due to its unique network structure design; thus, it can still effectively attack the DNN model even on small samples. In terms of parameter setting, Section IV-G evaluated the weight coefficient of the generator, the weight coefficient of the attenuator, and the influence of the training ratio on the TUAN on the MSTAR dataset. To obtain a better attack performance, attackers can flexibly adjust the parameters during the training phase according to actual needs. Section IV-H visualized the process of the generator mapping random noise to the UAP step-by-step because the generator can effectively establish the mapping from random noise to UAP through extensive training. Section IV-I explored the misclassification of adversarial examples generated by the surrogate model on other victim models under non-targeted attacks, and the results showed that DNN models with different network structures show similar misclassification results. This is because DNN models have similar decision boundaries even if they are structurally different. Section IV-J visualized the classification results of targeted attacks, proving that the proposed algorithm had a strong attack performance in targeted attacks. Section IV-K showed adversarial example images of different algorithms, demonstrating that the perturbation region generated by the TUAN was more focused on the target region than the baseline algorithm. This was because the features of the target region have the most remarkable impact on the DNN model identification process; therefore, to improve the attack effectiveness of adversarial examples, the attenuator forces the generator to produce adversarial perturbations for the target region.

VI. Conclusion

This study proposed the TUAN for DNN-based SAR-ATR models. First, the TUAN used the mutual game of the generator and attenuator to improve the transferability of universal adversarial examples. Second, with the unique network structure characteristics of U-Net, the TUAN effectively improved the attack performance of universal adversarial examples under small-sample conditions. The in silico results showed that the TUAN had better attack performance in both the non-targeted and targeted attacks than baseline algorithms. In addition, UAPs generated by the TUAN were mainly concentrated in the target region, which provides theoretical support for future physical practical deployment.

In future research, we plan to design network structure of the generator and attenuator to further improve the performance of black-box attack and small-sample attack.

References

[1] F. Zhang, X. Yao, H. Tang, Q. Yin, Y. Hu, and B. Lei, "Multiple mode SAR raw data simulation and parallel acceleration for Gaofen-3 mission," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 6, pp. 2115–2126, Jan. 2018, doi: 10.1109/JSTARS.2017.2787728.

[2] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, and I. Hajnsek, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013, doi: 10.1109/MGRS.2013.2248301.

[3] W. M. Brown, "Synthetic aperture radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-3, no. 2, pp. 217–229, Mar. 1967, doi: 10.1109/TAES.1967.5408745.

[4] Y. Wu et al., "RORNet: Partial-to-partial registration network with reliable overlapping representations," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, doi: 10.1109/TNNLS.2023.3286943.

[5] Y. Wu, X. Hu, Y. Zhang, M. Gong, W. Ma, and Q. Miao, "SACF-Net: Skip-attention based correspondence filtering network for point cloud registration," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3585–3595, Aug. 2023.

[6] Z. Wen, Y. Yu, and Q. Wu, "Multimodal discriminative feature learning for SAR ATR: A fusion framework of phase history, scattering topology, and image," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2023, Art. no. 5200414, doi: 10.1109/TGRS.2023.3340651.

[7] J. Yu, Z. Yu, L. Yu, P. Cheng, J. Chen, and C. Chi, "A comprehensive survey on SAR ATR in deep-learning era," *Remote Sens*, vol. 15, no. 5, Feb. 2023, Art. no. 1454.

[8] C. Wu, J. Yang, Y. Shang, and J. Pei, "Dynamically weighted prototypical learning method for few-shot SAR ATR," *IEEE Geosci. Remote Sens. Lett.*, vol. 21, 2024, Art. no. 4004705.

[9] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2014.

[10] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 86–94, doi: 10.1109/CVPR.2017.17.

[11] J. Hayes and G. Danezis, "Learning universal adversarial perturbations with generative models," in *Proc. IEEE Secur. Privacy Workshops*, San Francisco, CA, USA, 2018, pp. 43–49, doi: 10.1109/SPW.2018.00015.

[12] K. R. Mopuri, U. Garg, and R. V. Badu, "Fast Feature fool: A data independent approach to universal adversarial perturbations," 2017. [Online]. Available: https://arxiv.org/abs/1707.05572

[13] K. R. Mopuri, A. Ganeshan, and R. V. Babu, "Generalizable data-free objective for crafting universal adversarial perturbations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, pp. 2452–2465, Oct. 2019.

[14] K. R. Mopuri, U. Ojha, U. Garg, and R. V. Babu, "NAG: Network for adversary generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 742–751, doi: 10.1109/CVPR.2018.00084.

[15] H. Li et al., "Adversarial examples for CNN-based SAR image classification: An experience study," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1333–1347, 2021.

[16] L. Wang, X. Wang, S. Ma, and Y. Zhang, "Universal adversarial perturbation of SAR images for deep learning based target classification," in *Proc. IEEE 4th Int. Conf. Electron. Technol.*, Chengdu, China, 2021, pp. 1272–1276, doi: 10.1109/ICET51757.2021.9450944.
[17] W. Xia, Z. Liu, and Y. Li, "SAR-PeGA: A generation method of adversarial examples for SAR image target recognition network," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 2, pp. 1910–1920, Apr. 2023, doi: 10.1109/TAES.2022.3206261.
[18] M. Du, D. Bi, M. Du, X. Xu, and Z. Wu, "ULAN: A universal local adversarial network for SAR target recognition based on layer-wise relevance propagation," *Remote Sens.*, vol. 15, no. 1, Dec. 2022, Art. no. 21, doi: 10.3390/rs15010021.
[19] J. Zhou, H. Sun, and G. Kuang, "Template-based universal adversarial perturbation for SAR target classification," in *Proc. 8th China High Resolution Earth Observ. Conf.*, Beijing, China, 2022, pp. 351–360, doi: 10.1007/978-981-19-8202-6_32.
[20] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Artif. Intell. Saf. Secur.*, 2018, pp. 99–112.
[21] J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, 2015.
[22] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2574–2582, doi: 10.1109/CVPR.2016.282.
[23] 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, doi: 10.1109/EuroSP.2016.36.
[24] 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.
[25] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *Proc. 10th ACM Workshop Artif. Intell. Secur.*, Dallas, TX, USA, 2017, pp. 15–26, doi: 10.1145/3128572.3140448.
[26] M. Du, D. Bi, M. Du, and Z. Wu, "Local aggregative attack on SAR image classification models," in *Proc. IEEE 6th Adv. Inf. Technol., Electron. Autom. Control Conf.*, Beijing, China, 2022, pp. 1519–1524.
[27] M. I. J. Chen and M. J. Wainwright, "HopSkipJumpAttack: A query-efficient decision-based attack," in *Proc. IEEE Symp. Secur. Privacy*, San Francisco, CA, USA, 2020, pp. 1277–1294, doi: 10.1109/SP40000.2020.00045.
[28] C. Xie et al., "Improving transferability of adversarial examples with input diversity," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, 2019, pp. 15–20, doi: 10.1109/CVPR.2019.00284.
[29] G. Lin, Z. Pan, and X. Zhou, "Boosting adversarial transferability with shallow-feature attack on SAR images," *Remote Sens.*, vol. 15, no. 10, Apr. 2023, Art. no. 2699.
[30] W. Wu, Y. Su, M. R. Lyu, and I. King, "Improving the transferability of adversarial samples with adversarial transformations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Nashville, TN, USA, 2021, pp. 9024–9033.
[31] J. Goodfellow et al., "Generative adversarial networks," *Commun. ACM*, , vol. 63, pp. 139–144, 2020, doi: 10.1145/3422622.
[32] A. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, Munich, Germany, 2015, Art. no. 9351.
[33] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-net architecture for medical image segmentation," in *Proc. 4th Int. Workshop Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support*, Granada, Spain, 2018, pp. 3–11.
[34] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, 2016, pp. 694–711.
[35] C. Du and Z. Lei, "Adversarial attack for SAR target recognition based on UNet-generative Adversarial network," *Remote Sens*, vol. 13, no. 21, Oct. 2021, Art. no. 4358.
[36] C. Du, C. Huo, Z. Lei, C. Bo, and Y. Yan, "Fast C&W: A Fast adversarial attack algorithm to fool SAR target recognition with deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2021, Art. no. 4010005.
[37] Y. Han and J. C. Ye, "Framing U-Net via deep convolutional framelets: Application to sparse-view CT," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1418–1429, Jun. 2018.
[38] M. Z. Alom, C. Yakopcic, M. Hasan, T. M. Taha, and V. K. Asari, "Recurrent residual U-Net for medical image segmentation," *J. Med. Imag.*, vol. 6, 2019, Art. no. 014006.
[39] H. Xie, Q. Xu, Y. Cheng, X. Yin, and Y. Jia, "Reconstruction of subsurface temperature field in the South China Sea from satellite observations based on an attention U-Net model," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4209319.
[40] S. Wei, H. Zhang, C. Wang, Y. Wang, and L. Xu, "Multi-temporal SAR data large-scale crop mapping based on U-Net model," *Remote Sens*, vol. 11, no. 1, Art. no. 68, Jan. 2019.
[41] Z. Guo, L. Wu, Y. Huang, Z. Guo, J. Zhao, and N. Li, "Water-body segmentation for SAR images: Past, current, and future," *Remote Sens*, vol. 14, no. 7, Art. no. 1752, Mar. 2022.
[42] E. R. Keydel, S. W. Lee, and J. T. Moore, "MSTAR extended operating conditions: A tutorial," *Proc. SPIE*, vol. 2757, pp. 228–242, Jun. 1996, doi: 10.1117/12.242059.
[43] M. Schmitt, L. H. Hughes, and X. X. Zhu, "The SEN1-2 dataset for deep learning in SAR-optical data fusion," 2018, [Online]. Available: https://arxiv.org/abs/1807.01569
[44] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.
[45] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
[46] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 2818–2826.
[47] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017. [Online]. Available: https://arxiv.org/abs/1704.04861
[48] X. Z. He, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 770–778.
[49] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 6848–6856.
[50] A. Poursaeed, I. Katsman, B. Gao, and S. Belongie, "Generative adversarial perturbations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 4422–4431, doi: 10.1109/CVPR.2018.00465.
[51] H. Sun, J. Chen, L. Lei, K. Ji, and G. Kuang, "Adversarial robustness of deep convolutional neural network-based image recognition models: A review," *J. Radars*, vol. 10, no. 4, pp. 571–594, Aug. 2021, doi: 10.12000/JR21048.
[52] Z. Zhang, S. Liu, X. Gao, and Y. Diao, "An empirical study towards SAR adversarial examples," in *Proc. Int. Conf. Image Process., Comput. Vis. Mach. Learn.*, Xi'an. China, 2022, pp. 127–132, doi: 10.1109/ICICML57342.2022.10009880.
[53] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," 2017. [Online]. Available: https://arxiv.org/abs/1611.02770
[54] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Lake Tahoe, NV, USA, 2018, pp. 839–847, doi: 10.1109/WACV.2018.00097.

Xuanshen Wan was born in Taizhou, Zhejiang, China, in 1999. He is working toward the M.S. degree in information and communication engineering with the School of Data and Target Engineering, Information Engineering University, Zhengzhou, China.

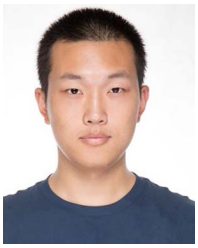
His research interests include synthetic aperture radar adversarial attack and synthetic aperture radar automatic target recognition.

Wei Liu received the B.S., M.S., and Ph.D. degrees in information and communication engineering from Information Engineering University, Zhengzhou, China, in 2001, 2003, and 2016, respectively.

He is currently an Associate Professor with Information Engineering University. His research interests include pattern recognition, remote sensing information processing, and deep learning.

Chaoyang Niu received the B.S. and M.S. degrees in information engineering from Zhengzhou Information Technology Institute, Zhengzhou, China, in 2003 and 2006, respectively, and the Ph.D. degree in signal and information processing from the Zhengzhou Institute of Surveying and Mapping, Zhengzhou, in 2011.

In 2016, he became an Associate Professor with Data and Target Engineering Institute, Information Engineering University, Zhengzhou. His research interests include pattern recognition, unmanned aerial vehicle remote sensing, and optical and radar imagery processing.



Wanjie Lu received the B.S. degree in photogrammetry and remote sensing and the Ph.D. degree in surveying and mapping from Information Engineering University, Zhengzhou, China, in 2016 and 2020, respectively.

He is currently a Lecturer with the Data and Target Engineering Institute, Information Engineering University. His research interests include unmanned aerial vehicle remote sensing image processing, deep learning algorithm, and spatial information service.

Meng Du received the B.S. and M.S. degrees in space engineering and information and communication engineering from the National University of Defense Technology, Changsha, China, in 2020 and 2023, respectively. He is currently working toward the Ph.D. degree with Information Engineering University, Zhengzhou, China.

His research interests include synthetic aperture radar adversarial attack and synthetic aperture radar automatic target recognition.

Yuanli Li received the B.S. degree in electronic and information engineering from Henan Polytechnic University, Jiaozuo, China, in 2022. She is currently working toward the M.S. degree in information and communication engineering with Information Engineering University, Zhengzhou, China.

Her research interests include semantic descriptions of synthetic aperture radar images.