Article

ULAN: A Universal Local Adversarial Network for SAR Target Recognition Based on Layer-wise Relevance Propagation

Meng Du *, Daping Bi, Mingyang Du, Xinsong Xu and Zilong Wu

College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China; bdpeei@163.com (D.B.); dumingyang17@nudt.edu.cn (M.D.); xuxinsong17@nudt.edu.cn (X.X.); wuzilong@nudt.edu.cn (Z.W.)

* Correspondence: dumeng_nudt@163.com

Abstract: Recent studies have proven that synthetic aperture radar (SAR) automatic target recognition (ATR) models based on deep neural networks (DNN) are vulnerable to adversarial examples. However, existing attacks are easily failed in the case where adversarial perturbations cannot be fully fed to victim models. We call this situation *perturbation offset*. Moreover, since background clutter takes up most of the areas in SAR images and has low relevance to recognition results, fooling models with global perturbations is quite inefficient. This paper proposes a semi-whitebox attack network, called *Universal Local Adversarial Network* (ULAN), to generate universal adversarial perturbations (UAP) for the target regions of SAR images. In the proposed network, we calculate the model's attention heatmaps through layer-wise relevance propagation (LRP), which is used to locate the target regions of SAR images that have high relevance to recognition results. In particular, we utilize a generator based on the U-Net to learn the mapping from noise to UAPs and craft adversarial examples by adding the generated local perturbations to target regions. Experiments indicate that the proposed method fundamentally prevents perturbation offset and achieves comparable attack performance to conventional global UAPs by perturbing only a quarter or less of SAR image areas.

Keywords: deep neural network (DNN); synthetic aperture radar automatic target recognition (SAR-ATR); universal adversarial perturbation (UAP); U-Net; attention heatmap; layer-wise relevance propagation (LRP)

1. Introduction

Synthetic aperture radar (SAR) is widely used in military and civilian fields for its ability to image targets with high resolution under all-time and all-weather conditions [1–3]. However, unlike natural images, it is difficult for humans to intuitively understand SAR images without resorting to interpretation techniques. The most popular interpretation method at present is the SAR automatic target recognition (SAR-ATR) technology based on deep neural networks (DNNs) [4–8]. With its powerful representation capabilities, the DNN outperforms traditional supervised methods in image classification tasks. Yet, some researchers have proved that DNN-based SAR target recognition models are vulnerable to adversarial examples [9].

Szegedy et al. [10] first propose the concept of adversarial examples, that is, a well-designed tiny perturbation can lead to the misclassification of a well-trained recognition model. This discovery makes adversarial attacks become one of the biggest threats to artificial intelligence (AI) security. So far, researchers have proposed a series of adversarial attack methods, which can be divided into two categories from the perspective of prior knowledge: white-box attacks and black-box attacks. In white-box conditions, the attacker has high access to the victim model, which means that the attacker can utilize lots of prior information to craft adversarial examples. The typical white-box methods are gradient-based attacks [11,12], boundary-based attacks [13], saliency map-based attacks [14], etc. Conversely, in black-box conditions, the biggest challenge for attackers is that they can

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

only access the output information of the victim model, or even less. The representative black-box methods are probability label-based attacks [15,16], decision-based attacks [17], and transferability-based attacks [18], etc. While the above methods achieve fantastic attack performance, all of them fool DNNs with data-dependent perturbations, i.e., each input corresponds to a different adversarial perturbation, which is hard to satisfy in real-world deployments. Moosavi et al. [19] first propose a universal adversarial perturbation (UAP) that can deceive DNNs independently of the input data. Subsequently, the work in [20] designs a universal adversarial network to learn the mapping from noise to UAPs and demonstrates the transferability of UAPs across different network structures. Mopuri et al. [21] argue that it is difficult for attackers to obtain the training dataset of the victim model, so to reduce the dependence on the dataset, they propose a data-free method to generate UAPs by destroying the features extracted by convolutional layers. Another data-free work [22] uses class impressions to simulate real data distribution, generating UAPs with high transferability. In the field of remote sensing, Xu et al. [23] are the first to investigate the adversarial attack and defense in safety-critical remote sensing tasks. Meanwhile, they also propose the Mixup-Attack [24] to craft universal adversarial examples for remote sensing data. Furthermore, researchers [25] have successfully attacked an advanced YOLOv2 detector in the real world with just a printed patch. Thus, a further study on adversarial examples, especially UAPs, is necessary for both attackers and defenders.

With the wide application of DNNs in the field of SAR-ATR, researchers embark on the adversarial examples of SAR images. In terms of data-dependent perturbations, Li et al. [26] use the FGSM and BIM algorithms to produce abundant adversarial examples for the CNN-based SAR image classification model and comprehensively analyze various factors affecting the attack success rate. The work in [27] presents a Fast C&W algorithm for real-time attacks that introduces an encoder network to generate adversarial examples through one-step forward mapping of SAR images. To enhance the universality of adversarial perturbations, Wang et al. [28] utilize the method proposed in [19] to craft UAPs for SAR images and achieve high attack success rates. In addition, the latest research [29] has broken through the limitations of the digital domain and implemented the UAP of SAR images in the signal domain by transmitting a two-dimensional jamming signal.

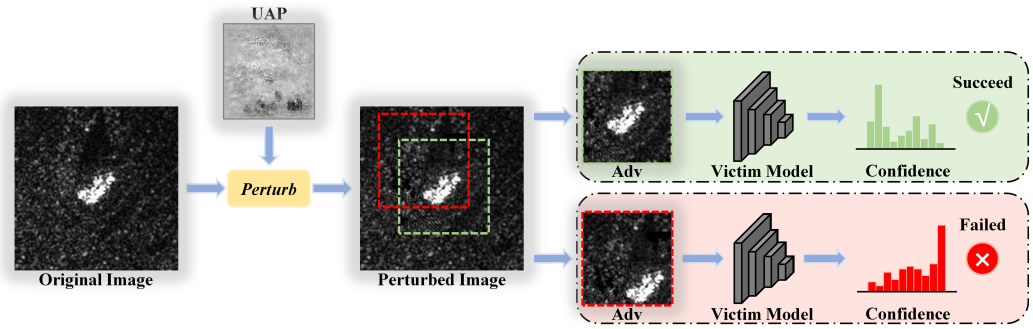

Figure 1. Given an original SAR image in the dataset and a well-designed UAP, we display the adversarial attacks with (bottom) and without (top) perturbation offset. Since we can only perturb a limited region, the perturbation size is smaller than the image size but identical to the input size of the victim model. Here, we attack the model by perturbing the green box region, that is, adding the UAP to this region. The adversarial attack without perturbation offset means that the perturbed (green box) region must be exactly fed to the model. However, suppose the model takes as input the red box region that has an offset from the perturbed region, the incomplete adversarial perturbation is likely to make the attack fail.

Although the above methods perform well in fooling SAR target recognition models, they are vulnerable and inefficient in practical applications. Specifically, existing attack methods are working on the assumption that the adversarial perturbations can be fully fed to the victim model, while it is not always held in practice, i.e., in many cases the

 doi:10.20944/preprints202211.0243.v1

perturbations fed to the model are incomplete, resulting in the failure of the adversarial attacks. We attribute the failure to the vulnerability of adversarial attacks and call this situation *perturbation offset*. For ease of understanding, we detail a specific example in Figure 1. On the other hand, we calculate the model's attention heatmaps [31] through layer-wise relevance propagation (LRP) [32], which is used to analyze the relevance of each pixel in the SAR image to the recognition results. The pixel-wise attention heatmaps can be found in Section 4.3. The fact is that the background regions of SAR images have little relevance to the model's outputs, and the features that greatly impact the recognition results are mainly concentrated in the target regions. However, existing attack methods fool DNN models by global perturbations so that massive time and computing resources are allocated to design perturbations for low-relevance background regions, which is undoubtedly inefficient. Therefore, the vulnerability and inefficiency of adversarial attacks are pending to be solved in real-world implementations.

In this paper, we propose a semi-whitebox [33] attack network — called *Universal Local Adversarial Network* (ULAN) to generate UAPs for target regions of SAR images. Specifically, we first calculate the model's attention heatmaps through LRP to locate the target regions in SAR images that have high relevance to the recognition results. Then, we utilize a U-Net [30] to learn the mapping from noise to UAPs and craft the adversarial examples by adding the generated local perturbations to the target regions. In this way, attackers can focus perturbations on the high-relevance target regions, which significantly improves the efficiency of adversarial attacks. Meanwhile, the proposed method also ensures that the well-designed perturbations can be fully fed to the victim model along with the targets such that perturbation offset is fundamentally prevented.

The main contributions of this paper are summarized as follows.

(1) We are the first to evaluate the adversarial attacks against DNN-based SAR-ATR models in the case of perturbation offset and analyze the relevance of each pixel in SAR images to the recognition results. Our research reveals the vulnerability and inefficiency of existing adversarial attacks in SAR target recognition tasks.

(2) A semi-whitebox attack network is proposed to generate UAPs for the target regions of SAR images. Once the proposed network is trained, it can real-time attack the victim model without requiring access to the model itself anymore, and thus possesses high potential in practical applications.

(3) Experiments on the moving and stationary target acquisition and recognition (MSTAR) dataset show that the proposed method not only prevents perturbation offset effectively, but also achieves comparable attack performance to the conventional global UAPs by perturbing only a quarter or less of the SAR image area. Furthermore, we evaluate the attack performance of the ULAN under small sample conditions, and the result shows that given five images per class, our method can cause a misclassification rate over 70%.

The rest of this paper is organized as follows. Section 2 introduces the relevant preparation knowledge. In Section 3, we describe the proposed method in detail. The experiment results are shown in Section 4. The conclusion is given in Section 5.

2. Preliminary

2.1. Universal Adversarial Perturbations for SAR Target Recognition

Suppose $x_n \in [0,255]^{W\times H}$ is an 8-bit gray-scale image from the SAR image dataset $\mathcal{X}$, and $f(\cdot)$ is a DNN-based k-class SAR target recognition model without a softmax output layer. Given a sample x_n as input to $f(\cdot)$, the output is a k-dimensional logits vector $f(x_n)=[f(x_n)_1, f(x_n)_2, \cdots, f(x_n)_k]$, where $f(x_n)_i \in \mathbb{R}$ denotes the score of x_n belonging to class i. Let $C_p = \arg\max_i(f(x_n)_i)$ represent the predicted class of the model for x_n.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

Universal adversarial perturbations (UAP) can fool the model independently of the input data as follows:

$$\text{for "most" } x_n \in \mathcal{X} \quad \text{s.t.} \begin{cases} \arg\max_i (f(x_n+\delta)_i) \neq C_p \\ \|\delta\|_p \leq \xi \end{cases} \tag{1}$$

where δ is a UAP, the L_p-norm is defined as $\|\delta\|_p = (\sum_i |\delta_i|^p)^{\frac{1}{p}}$, and ξ controls the magnitude of δ. Meanwhile, adversarial attacks can be divided into non-targeted and targeted attacks in terms of attack modes. As the name suggests, the former just makes DNN models misclassify, while the latter induces models to output specified results. From a military perspective, targeted attacks are more challenging and threatening than non-targeted attacks. In other words, UAPs can reduce the probability that DNN models correctly recognize samples in non-targeted attack scenarios; conversely, they increase the probability of models identifying samples as target classes in targeted attack scenarios. Therefore, we transform (1) into the following optimization problems:

- for the non-targeted attack:

$$minimize\left(\frac{\sum_{n=1}^{N} D(\arg\max_i (f(x_n+\delta)_i) == C_{tr})}{N}\right), \quad \text{s.t.} \|\delta\|_p \leq \xi \tag{2}$$

- for the targeted attack:

$$maximize\left(\frac{\sum_{n=1}^{N} D(\arg\max_i (f(x_n+\delta)_i) == C_{ta})}{N}\right), \quad \text{s.t.} \|\delta\|_p \leq \xi \tag{3}$$

where the discriminant function $D(\cdot)$ equals one if the equation holds; otherwise equals zero. N is the total number of images in the dataset. C_{tr} and C_{ta} represent the true and target classes of the input data. Obviously, the above optimization problems are exactly the opposite of a DNN's training process, and the corresponding loss functions will be given in the next chapter.

2.2. Attention Heatmaps

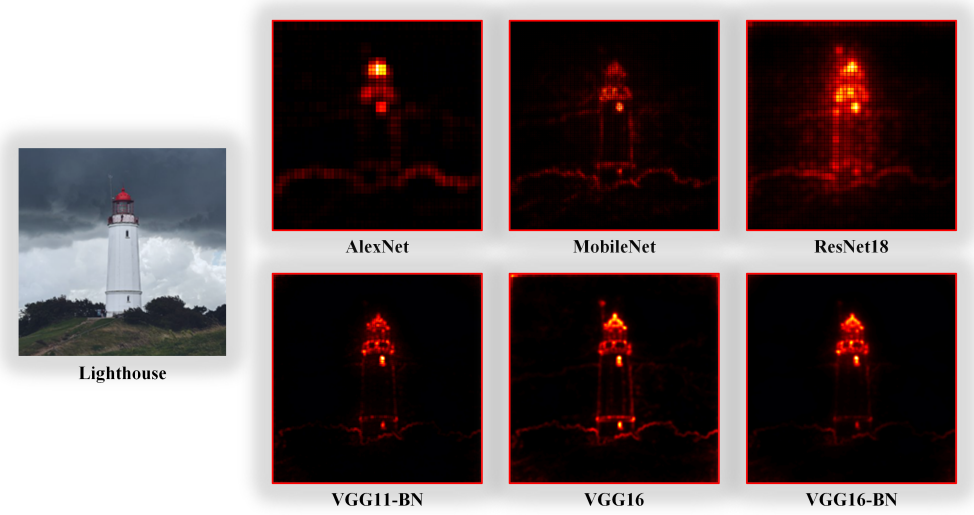

Figure 2. Attention heatmaps for AlexNet [34], MobileNet [35], ResNet18 [36], VGG11-BN, VGG16, and VGG16-BN [37].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

When humans make judgments, they can reasonably allocate their attention to different features of an object and get the desired semantic information efficiently. Coincidentally, recent studies have shown that DNNs have similar characteristics when making decisions [31]. For example, in image classification tasks, the pixels surrounding target regions tend to have a much greater impact on the classification results than others. Researchers typically utilize attention heatmaps to visualize the contribution of each pixel to the network output.

Nowadays, lots of algorithms have been proposed to calculate the DNN's attention heatmaps. In this paper, we employ layer-wise relevance propagation (LRP) [32] to obtain the pixel-wise attention heatmaps, which is actually a backward visualization method [38–40] that obtains the heatmap by calculating the relevance between adjacent layers from outputs to inputs. Figure 2 displays the heatmaps of six DNNs calculated by LRP. As we can see, the hotspots are mainly concentrated in the target regions, and the heatmaps of different DNNs have similar structures, i.e., attention heatmaps may be the semantic features shared by DNNs. Destroying the common semantic feature of DNNs is a promising idea to enhance the transferability of adversarial examples. We will detail the principle of LRP in Section 3.2.

3. The Proposed Universal Local Adversarial Network (ULAN)

The framework of the universal local adversarial network (ULAN) is shown in Figure 3. To describe the training process of the ULAN more clearly, we divide it into four steps. The first step uses a generator to learn the mapping from normal distribution noise into universal adversarial perturbations (UAPs). Next, the second step calculates the pixel-wise attention heatmaps of the surrogate model through layer-wise relevance propagation (LRP). Then, the third step utilizes UAPs and attention heatmaps to craft adversarial examples of SAR images. Finally, the fourth step computes the training loss and updates the generator's parameters through backward propagation. Note that the victim model is a white-box in the training phase, but in the testing phase it is a black-box, and thus we calculate the heatmap of the surrogate model as an alternative to the victim network's heatmap. This chapter will introduce each of the above steps in detail.

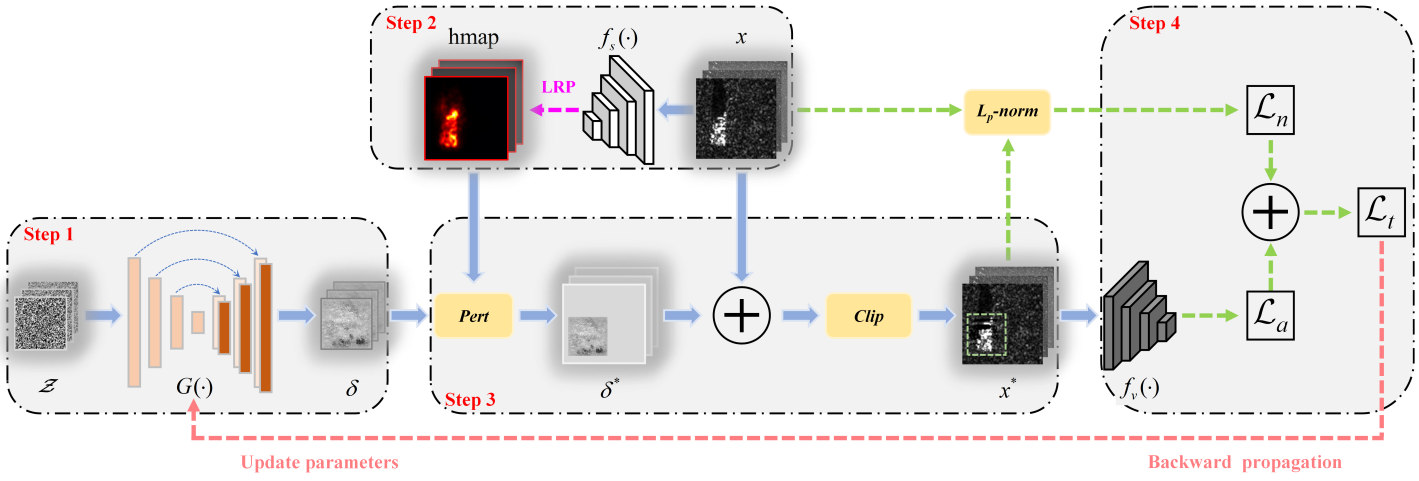

Figure 3. Framework of the ULAN. The UAP δ generated by the generator $G(\cdot)$ is a local perturbation, and its size is much smaller than the SAR image x. Attackers utilize the attention heatmap (hmap) of the surrogate model $f_s(\cdot)$ to locate the target region, i.e., the green box region, and obtain the adversarial example by adding δ to the target region. The victim model $f_v(\cdot)$ takes the adversarial example as input and outputs the attack loss L_a, plus the norm loss L_n to form the total loss L_t, which is used to update the parameters of $G(\cdot)$.

3.1. Structure of Generative Network

In order to craft UAPs independently of the input data, this paper trains a generative network $G(\cdot)$ to transform the normal distribution noise $\mathcal{Z}$ into a UAP δ as follows:

$$\delta = G(\mathcal{Z}), \quad \mathcal{Z} \sim \mathcal{N}(0,1) \tag{4}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

where $\mathcal{Z}$ and δ have the same size, denoted as $w \times h$. Meanwhile, we set the size of SAR images to $W \times H$. Since the generated δ is a local perturbation, the relationship between $w \times h$ and $W \times H$ is that $w \times h \ll W \times H$.

The characteristics of SAR images should be taken into account when choosing the generative network. First of all, a SAR image mainly consists of the target and background clutter. Yet, the features that have great impact on the recognition results are mainly concentrated in the target region, which only occupies a tiny part of the SAR image. Second, compared to natural images, the professionalism and confidentiality of SAR images make them challenging to access. This means that we need to consider adversarial attacks under small sample conditions, so a lightweight generator is necessary to prevent network overfitting. In summary, this paper takes the U-Net as the generator to craft UAPs.

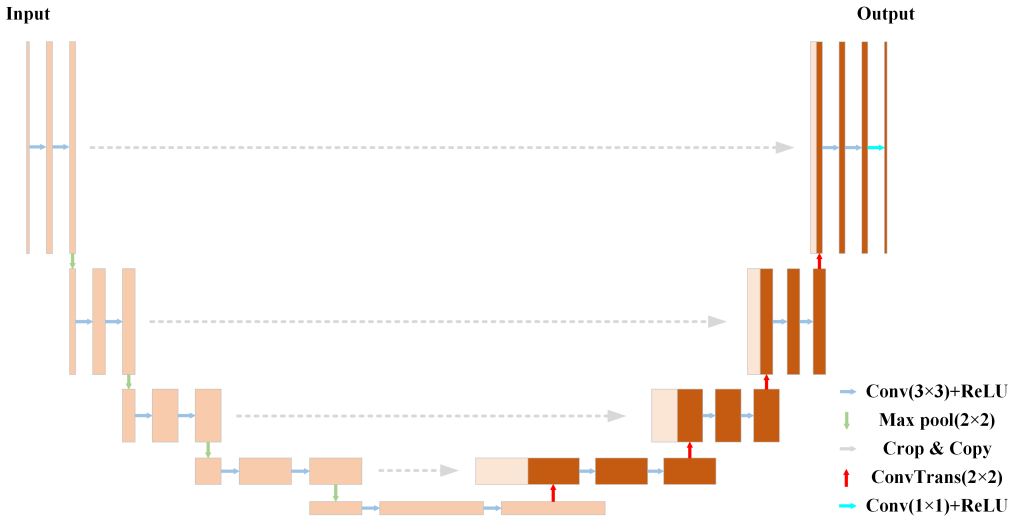

Figure 4. The structure of the U-Net.

Table 1. The network parameters. Here we set $w \times h$ to 32×32 and abbreviate the combination of two convolutional layers as DoubleConv. The parameters of the convolutional layer represent the number of input and output channels and the kernel size, respectively. The parameter of the max-pooling layer represents the kernel size.

Layer	Shape
Input	$1 \times 32 \times 32$
DoubleConv$(1, 64, 3)$+Max pool(2)	$64 \times 16 \times 16$
DoubleConv$(64, 128, 3)$+Max pool(2)	$128 \times 8 \times 8$
DoubleConv$(128, 256, 3)$+Max pool(2)	$256 \times 4 \times 4$
DoubleConv$(256, 512, 3)$+Max pool(2)	$512 \times 2 \times 2$
DoubleConv$(512, 1024, 3)$	$1024 \times 2 \times 2$
ConvTrans$(1024, 512, 2)$+DoubleConv$(1024, 512, 3)$	$512 \times 4 \times 4$
ConvTrans$(512, 256, 2)$+DoubleConv$(512, 256, 3)$	$256 \times 8 \times 8$
ConvTrans$(256, 128, 2)$+DoubleConv$(256, 128, 3)$	$128 \times 16 \times 16$
ConvTrans$(128, 64, 2)$+DoubleConv$(128, 64, 3)$	$64 \times 32 \times 32$
Conv$(64, 1, 1)$	$1 \times 32 \times 32$

Figure 4 shows the detailed U-Net structure. It is first proposed to segment biomedical images [30] and mainly consists of an encoder and a decoder. The encoder extracts features by down-sampling the input data, while the decoder recovers the data by up-sampling feature maps. The biggest difference between the U-Net and other common encoder-decoder models is that the former introduces a skip connection operation to fuse features from different layers. Specifically, both the encoder and the decoder consist of four sub-blocks. The encoder block contains two 3×3 convolutional layers and a 2×2 max-pooling

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

layer, while the decoder block contains a 2×2 transposed convolutional layer and two 3×3 convolutional layers. Note that the last layer of the decoder utilizes a 1×1 convolutional layer to make the number of input and output channels identical. The network parameters are given in Table 1.

3.2. Layer-wise Relevance Propagation (LRP)

To analyze the relevance of each pixel in SAR images to the recognition results, we need obtain the DNN model's attention heatmaps first. In this paper, we use layer-wise relevance propagation (LRP) [32] to calculate the pixel-wise attention heatmaps of the surrogate model $f_s(\cdot)$. For an easy explanation, we suppose $f_s(\cdot)$ is an l-layer DNN without the softmax output layer. Figure 5 illustrates the network's forward propagation and LRP.

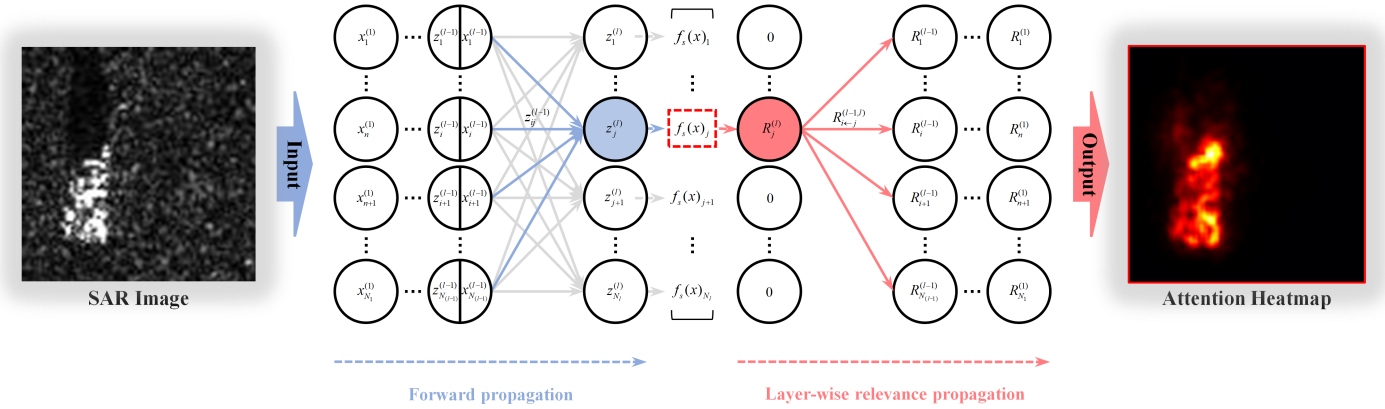

Figure 5. Forward propagation (left) and LRP (right) of the surrogate model $f_s(\cdot)$.

The left of Figure 5 shows a standard forward propagation, which takes a SAR image x as input and outputs a logits vector $f_s(x)$. A common mapping from one layer to the next one can be expressed as follows:

$$x_i^{(l-1)} = \sigma(z_i^{(l-1)}) \tag{5}$$

$$z_{ij}^{(l-1)} = w_{ij}^{(l-1)} x_i^{(l-1)} \tag{6}$$

$$z_j^{(l)} = \sum_i z_{ij}^{(l-1)} + b_j^{(l)} \tag{7}$$

where $z_i^{(l-1)}$ and $x_i^{(l-1)}$ denote the pre-activation and post-activation of the corresponding node (superscript and subscript denote layer and node indices, respectively), $\sigma(\cdot)$ is an activation function, $w_{ij}^{(l-1)}$ and $z_{ij}^{(l-1)}$ can be understood as the weight and local pre-activation between nodes $x_i^{(l-1)}$ and $z_j^{(l)}$, and $b_j^{(l)}$ is a bias term. The activation function $\sigma(\cdot)$ is usually nonlinear, such as the hyperbolic tangent *tanh* or the rectification function *ReLU*, which can enhance the network's representation capacity. Note that the input and output layers typically don't include activation functions, and the output $f_s(x)=[f_s(x)_1, f_s(x)_2, \cdots, f_s(x)_{N_l}]$ is a logits vector without softmax operations.

As for LRP, given a target class output $f_s(x)_j$ as input, its output is a pixel-wise attention heatmap reflecting the image regions most relevant to $f_s(x)_j$. Specifically, we sequentially decompose the relevance of each node for the target class output $f_s(x)_j$ from the neural network's output layer to input layer. Meanwhile, the backward propagation of the relevance must satisfy the following conservation property:

$$f_s(x)_j = R_j^{(l)} = \sum_i R_i^{(l-1)} = \cdots = \sum_n R_n^{(1)} \tag{8}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

A common decomposition is to allocate the relevance according to the ratio of local to global pre-activations in the forward propagation, as follows:

$$R_{i\leftarrow j}^{(l-1,l)} = \frac{z_{ij}^{(l-1)}}{z_j^{(l)}} \cdot R_j^{(l)} \tag{9}$$

where $R_{i\leftarrow j}^{(l-1,l)}$ denotes the relevance assigned from node $R_j^{(l)}$ to node $R_i^{(l-1)}$. This decomposition can approximately satisfy the conservation property in (8):

$$\begin{aligned} \sum_i R_{i\leftarrow j}^{(l-1,l)} &= R_j^{(l)} \cdot (1 - \frac{b_j^{(l)}}{z_j^{(l)}}) \\ &\approx R_j^{(l)} \end{aligned} \tag{10}$$

Additional, considering that if $z_j^{(l)}$ goes to zero, then $R_{i\leftarrow j}^{(l-1,l)}$ will close to infinity, so (9) can be modified by introducing a stable term $\epsilon \geq 0$ as follows:

$$R_{i\leftarrow j}^{(l-1,l)} = \frac{z_{ij}^{(l-1)}}{z_j^{(l)} + \epsilon \cdot sign(z_j^{(l)})} \cdot R_j^{(l)} \tag{11}$$

In summary, we can calculate the relevance of each node for the target class output through the following recursion formula and backward pass the relevance until reaching the input layer.

$$R_i^{(l-1)} = \sum_j \frac{z_{ij}^{(l-1)}}{z_j^{(l)} + \epsilon \cdot sign(z_j^{(l)})} \cdot R_j^{(l)} \tag{12}$$

3.3. Adversarial Examples of SAR Images

To add the local perturbations generated in Section 3.1 to the target regions of SAR images, we determine the perturbation location through the attention heatmaps calculated by Section 3.2. Therefore, we take the attention heatmap centroid as the perturbation center and design a perturbation function to craft the adversarial examples.

First of all, the coordinates of the image centroid can be calculated by the following formula [41]:

$$(u_c, v_c) = (\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}}) \tag{13}$$

where M_{00} is the zero-order moment of the image, M_{10} and M_{01} are the first-order moments of the image. Here involves the calculation of higher-order moments, which are generally defined as:

$$M_{\alpha\beta} = \int\int u^\alpha v^\beta f(u,v) du dv \tag{14}$$

For a digital image, we regard the coordinates of the pixel as a two-dimensional random variable (u, v), and the value of each pixel is regarded as the density of the point. Thus, a gray-scale image can be represented by a two-dimensional gray-scale density function $V(u, v)$, and its higher-order moments can be expressed as:

$$M_{\alpha\beta} = \sum_u \sum_v V(u,v) \cdot u^\alpha \cdot v^\beta \tag{15}$$

Note that the premise here is a two-dimensional gray-scale image, so we convert the attention heatmap *hmap* to a single-channel gray-scale image first, and then preprocess it with Gaussian blur and binarization algorithms [42].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

Then, we take the attention heatmap centroid as the perturbation center, so the pixel coordinates corresponding to $\delta(0,0)$, i.e., the perturbation origin, can be derived as:

$$(u_o, v_o) = (u_c + \Delta u - \left\lfloor \frac{w}{2} \right\rfloor, v_c + \Delta v - \left\lfloor \frac{h}{2} \right\rfloor) \tag{16}$$

where w and h are the width and height of δ, $\left\lfloor \frac{w}{2} \right\rfloor$ and $\left\lfloor \frac{h}{2} \right\rfloor$ represent the displacement difference between the perturbation center and the perturbation origin in the horizontal and vertical directions, and $\lfloor \cdot \rfloor$ means rounding down. Meanwhile, this paper adds a two-dimensional random noise $(\Delta u, \Delta v) \sim \mathcal{U}(-5,5)$ on the centroid coordinates to improve the generalization of our attack.

Next, we add the UAP δ to the perturbed region through the following perturbation function. Let $Pert(u_o, v_o, \delta, W, H)$ be a function that takes as input the perturbation origin coordinates (u_o, v_o), a UAP δ, and the size of SAR images $W \times H$, and outputs an adversarial perturbation $\delta^* \in \mathbb{R}^{W \times H}$ of the same size as SAR images, defined as:

$$\underset{\substack{0 \le u \le W-1 \\ 0 \le v \le H-1}}{\delta^*(u,v)} = \begin{cases} \delta(u - u_o, v - v_o) & \text{, if} \begin{cases} u_o \le u \le u_o + w - 1 \\ v_o \le v \le v_o + h - 1 \end{cases} \\ 0 & \text{, otherwise} \end{cases} \tag{17}$$

In brief, the adversarial perturbation $\delta^* = Pert(u_o, v_o, \delta, W, H)$ equals zero at all pixels except the pixels in the perturbed region.

Finally, the adversarial example x^* can be expressed as:

$$x^* = Clip_{[0,255]}(x + \delta^*) \tag{18}$$

The clipping operation restricts the pixel values of x^* to the interval of $[0, 255]$, ensuring that x^* is still an 8-bit gray-scale image.

3.4. Design of Loss Functions

To effectively fool the DNN model with a minor perturbation, we design a loss function $\mathcal{L}_t$ consisting of an attack loss $\mathcal{L}_a$ and a norm loss $\mathcal{L}_n$. This section will introduce them in detail.

For the non-targeted attack: In this paper, we design an attack loss $\mathcal{L}_a$ on the basis of the following standard cross-entropy loss.

$$loss(f_v(x), C_{tr}) = -\log\left(\frac{\exp(f_v(x)_{C_{tr}})}{\sum_j \exp(f_v(x)_j)}\right) \tag{19}$$

where $f_v(x)$ is the logits output of the victim model. The above formula actually contains the following softmax operation:

$$\text{softmax}(f_v(x)_i) = \left(\frac{\exp(f_v(x)_i)}{\sum_j \exp(f_v(x)_j)}\right) \in [0,1] \tag{20}$$

Obviously, the cross-entropy loss in (19) has been widely used in network training to improve the DNN model's classification accuracy by increasing the confidence of true classes. Instead, according to Formula 2, the non-targeted attack can minimize the classifi-

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

cation accuracy by decreasing the confidence of true classes, i.e., increasing the confidence of others, and thus, the attack loss $\mathcal{L}_a$ can be expressed as:

$$\begin{aligned}\mathcal{L}_a(f_v(x^*), C_{tr}) &= -\log\left(\frac{\sum_{j\neq C_{tr}} \exp(f_v(x^*)_j)}{\sum_j \exp(f_v(x^*)_j)}\right) \\ &= -\log\left(1 - \frac{\exp(f_v(x^*)_{C_{tr}})}{\sum_j \exp(f_v(x^*)_j)}\right)\end{aligned} \tag{21}$$

Meanwhile, a norm loss $\mathcal{L}_n$ is introduced to limit the perturbation magnitude. We use the traditional L_p-norm to measure the degree of image distortion as follows:

$$\begin{aligned}\mathcal{L}_n(x, x^*) &= \|x^* - x\|_p \\ &= (\sum_i |\Delta x_i|^p)^{\frac{1}{p}}\end{aligned} \tag{22}$$

Then, we apply the linear weighted sum method to balance the relationship between $\mathcal{L}_a$ and $\mathcal{L}_n$, so the total loss $\mathcal{L}_t$ can be represented as:

$$\begin{aligned}\mathcal{L}_t &= \mathcal{L}_a(f_v(x^*), C_{tr}) + \omega \cdot \mathcal{L}_n(x, x^*) \\ &= \omega \cdot \|x^* - x\|_p - \log\left(1 - \frac{\exp(f_v(x^*)_{C_{tr}})}{\sum_j \exp(f_v(x^*)_j)}\right)\end{aligned} \tag{23}$$

where $\omega \geq 0$ is a constant that measures the relative importance of the attack effectiveness and the attack stealthiness.

For the targeted attack: According to Formula 3, the targeted attack is to maximize the probability that the victim model recognizes samples as target classes. In other words, we need to increase the confidence of target classes. Thus, contrary to the non-targeted attack, the attack loss $\mathcal{L}_a$ can be expressed as:

$$\mathcal{L}_a(f_v(x^*), C_{ta}) = -\log\left(\frac{\exp(f_v(x^*)_{C_{ta}})}{\sum_j \exp(f_v(x^*)_j)}\right) \tag{24}$$

The norm loss $\mathcal{L}_n$ is the same as (22), so the total loss $\mathcal{L}_t$ of the targeted attack can be derived as follows:

$$\begin{aligned}\mathcal{L}_t &= \mathcal{L}_a(f_v(x^*), C_{ta}) + \omega \cdot \mathcal{L}_n(x, x^*) \\ &= \omega \cdot \|x^* - x\|_p - \log\left(\frac{\exp(f_v(x^*)_{C_{ta}})}{\sum_j \exp(f_v(x^*)_j)}\right)\end{aligned} \tag{25}$$

4. Experiments

4.1. Dataset and Implementation Details

4.1.1. Dataset

The moving and stationary target acquisition and recognition (MSTAR) dataset [43] published by the U.S. Defence Advanced Research Projects Agency (DARPA) is employed in our experiments. MSTAR is collected by the high-resolution spotlight SAR and contains SAR images of Soviet military vehicle targets at different azimuth and depression angles. All the experiments are performed under the standard operating condition (SOC), which includes ten ground target classes, such as self-propelled howitzer (2S1); infantry fighting vehicle (BMP2); armored reconnaissance vehicle (BRDM2); wheeled armored transport vehicle (BTR60, BTR70); bulldozer (D7); main battle tanks (T62, T72); cargo truck (ZIL131); self-propelled artillery (ZSU234). The training dataset contains 2747 images collected at 17° depression angle, and the testing dataset contains 2426 images captured at 15° depression

 doi:10.20944/preprints202211.0243.v1

angle. More details about the dataset are shown in Table 2, and Figure 6 shows the optical images and corresponding SAR images of ten ground target classes.

Table 2. Details of MSTAR under SOC, including target class, serial, depression angle, and number of training and testing images.

Target Class	Serial	Training Data		Testing Data	
		Depression Angle	Number	Depression Angle	Number
2S1	b01	17°	299	15°	274
BMP2	9566	17°	233	15°	196
BRDM2	E-71	17°	298	15°	274
BTR60	k10yt7532	17°	256	15°	195
BTR70	c71	17°	233	15°	196
D7	92v13015	17°	299	15°	274
T62	A51	17°	299	15°	273
T72	132	17°	232	15°	196
ZIL131	E12	17°	299	15°	274
ZSU234	d08	17°	299	15°	274

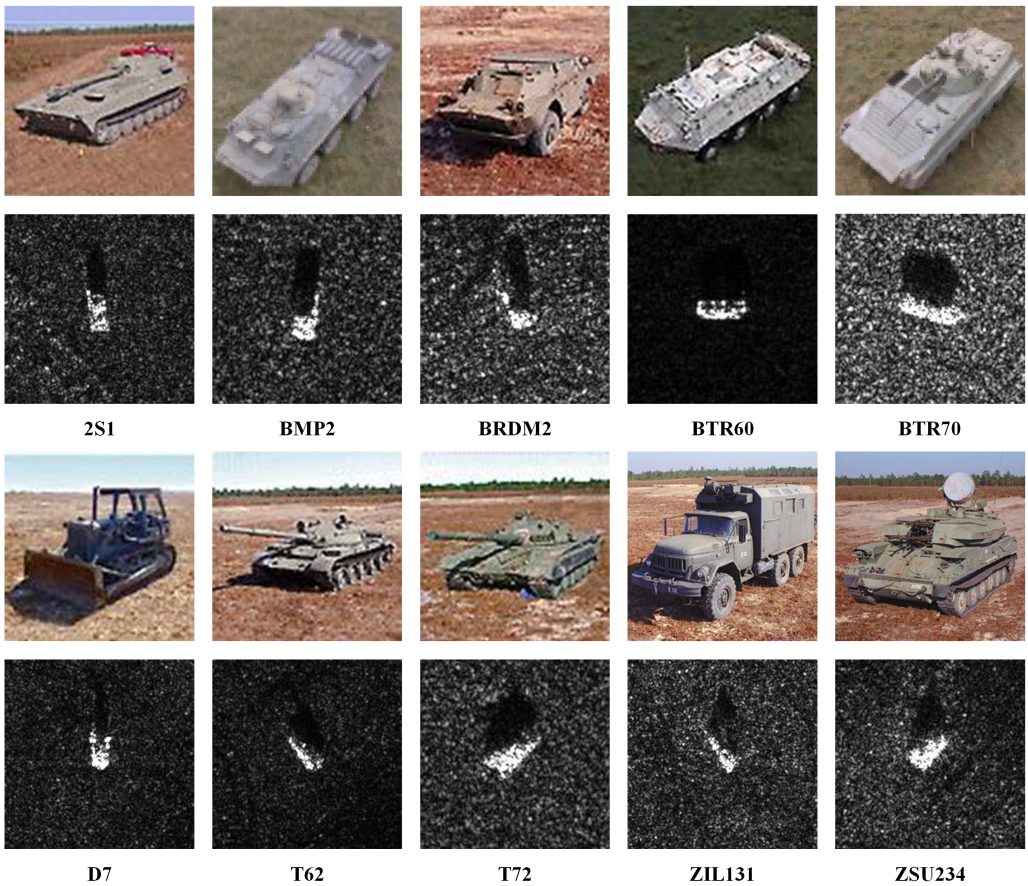

Figure 6. Optical images (top) and SAR images (bottom) of ten ground target classes.

4.1.2. Implementation Details

Due to the different sizes of SAR images in MSTAR, we first center-crop the image to 128×128. Meanwhile, in practice, the target is not necessarily located in the center of the SAR image. Thus, we random-crop the cropped image to 88×88 again, and finally normalize it to $\mathcal{N}(0,1)$.

For the victim models, we adopt six common DNNs, A-ConvNets-BN [44], VGG16-BN [37], GoogLeNet [45], InceptionV3 [46], ResNet50 [36] and ResNeXt50 [47], which are trained on the MSTAR dataset and have a classification accuracy of over 97%. The surrogate

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

model employs a well-trained VGG16-BN network to approximate the pixel-wise attention heatmap of the victim model. During the training phase, we form the validation dataset by uniformly sampling 10% data from the training dataset, and use the Adam optimizer [48] with the learning rate 0.001, the training epoch 15, and the training batch size 32. The size of UAPs defaults to 44×44, the norm type defaults to L_2-norm, and the weight coefficient ω defaults to 0.5.

Considering that most of the current research aims to craft global adversarial perturbations for SAR images, few scholars focus on universal or local perturbations. Therefore, in the comparative experiments, we take the methods proposed in [20,49] as baselines to compare with the ULAN. Note that baseline methods generate global UAPs for SAR images, while our method only needs to perturb local regions. All codes are written in Pytorch and the experimental environment consists of Windows 10; GPU (NVIDIA GeForce RTX 2080 Ti); and CPU (3.6GHz Intel(R) Core(TM) i9-9900K).

4.2. Evaluation Metrics

This paper takes into account two factors to comprehensively evaluate the performance of adversarial attacks: the attack effectiveness and the attack stealthiness. In the experiments, we craft adversarial examples for all samples in the SAR image dataset, so the victim model's classification accuracy directly reflects the attack effectiveness of UAPs:

$$Acc = \begin{cases} \dfrac{\sum_{n=1}^{N} D(\arg\max_i(f(x_n+\delta)_i)==C_{tr})}{N} & \text{Non-targeted Attack} \\ \dfrac{\sum_{C_{ta}=1}^{k}\sum_{n=1}^{N} D(\arg\max_i(f(x_n+\delta)_i)==C_{ta})}{k\times N} & \text{Targeted Attack} \end{cases} \tag{26}$$

where C_{tr} and C_{ta} represent the true and target classes of the input data, k is the number of target classes, and $D(\cdot)$ is a discriminant function. The non-targeted attack effect is inversely proportional to the classification accuracy, while the targeted attack performance is proportional to the Acc metric. Moreover, to verify the reliability of attacks, we also compare the confidence level of target classes before and after the attack.

When evaluating the attack stealthiness, in addition to using the L_p-norm to measure the degree of image distortion, we also introduce the structural similarity (SSIM) [50], a metric more in line with human visual perception, for a more objective evaluation, defined as:

$$SSIM(a,b) = \frac{(2\mu_a\mu_b + C_1)(2\sigma_{ab} + C_2)}{(\mu_a^2 + \mu_b^2 + C_1)(\sigma_a^2 + \sigma_b^2 + C_2)} \tag{27}$$

where a and b are the images to be compared, μ_a, μ_b and σ_a, σ_b are the mean and standard deviation of the corresponding image, σ_{ab} is the covariance, and C_1, C_2 are the constants used to keep the metric stable. The value of the SSIM ranges from -1 to 1, and the higher the SSIM, the more imperceptible the adversarial perturbation.

4.3. Attention Heatmaps for DNN-based SAR Target Recognition Models

For the six victim models mentioned in Section 4.1.2, given ten SAR images from different target classes as input, they all correctly classify the targets with high confidence. Then, we calculate pixel-wise attention heatmaps for the victim models by LRP, as shown in Figure 7. The result is similar to the natural image in Figure 2, i.e., the pixels that have a great impact on the SAR image classifiers are mainly concentrated in the target regions. Furthermore, we find that the attention heatmaps of different models have similar structures, which proves the feasibility of our method. Specifically, since the victim model is a black-box in the testing phase, attackers are unable to directly obtain its attention heatmaps through LRP. However, due to the similarity of attention heatmaps between different DNN models, we can calculate a white-box surrogate model's attention heatmap as an alternative. Meanwhile, since the attention heatmap of VGG16-BN best matches the target shape and has the clearest boundary, the surrogate model adopts a well-trained VGG16-BN network to approximate the attention heatmap of the victim model.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

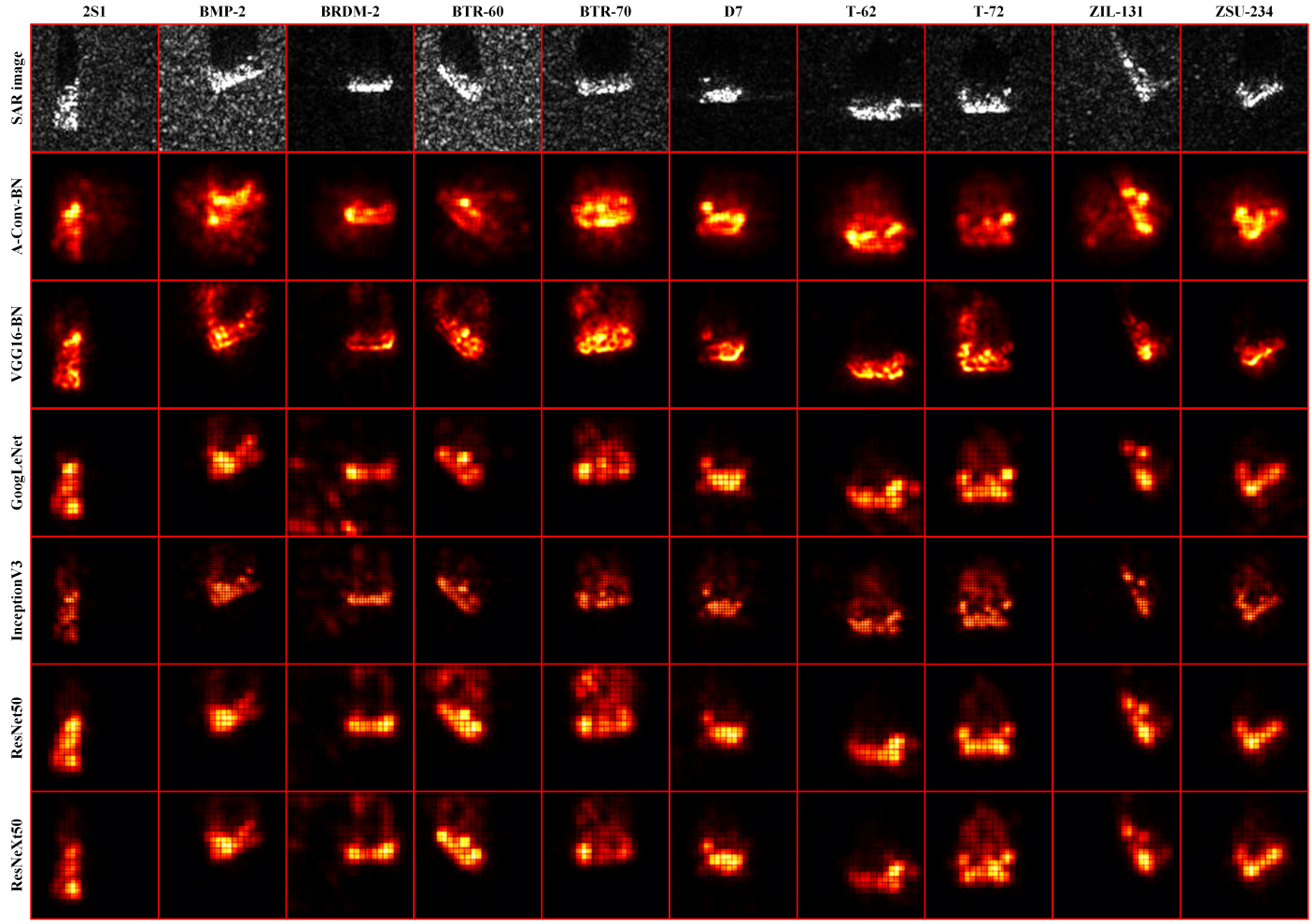

Figure 7. Pixel-wise attention heatmaps for DNN-based SAR-ATR models. The true class of the SAR image is listed at the top, and the DNN structure is shown on the left.

4.4. Adversarial Attacks without Perturbation Offset

In this experiment, we evaluate the non-targeted and targeted attack performance of each method without perturbation offset. Specifically, we first crop the SAR image to 88×88 as mentioned in Section 4.1.2, and then craft adversarial examples by adding the well-designed perturbations to the cropped images, which ensures that the perturbations can be fully fed to the victim model. Note that the structures and parameters of the model are known in the training phase, while these details are unavailable in the testing phase. Moreover, we emphasize that the UAPs generated by baseline methods cover the global SAR images, but our method only needs to perturb target regions. The results of the non-targeted and targeted attacks are shown in Table 3 and Table 4, respectively. There are four metrics in the table to evaluate the attack performance: the classification accuracy and target class confidence before and after the attack, the L_2-norm of image distortion, and the SSIM between clean and adversarial examples.

In the non-targeted attack, the classification accuracy of each DNN model on the testing dataset exceeds 95%, and the target class confidence is over 0.9. However, after the attack, the model's classification accuracy decreases significantly, the maximum reduction reaches 85%, and the minimum exceeds 60%; the drop in target class confidence varies from 0.6 to 0.85. From the perspective of attack effectiveness, the UAN performs the best, followed by the ULAN and U-Net, and the worst is the ResNet Generator. Yet, the biggest drawback of baseline methods is that they need to perturb the global regions of size 88×88, but our method perturbs the target regions of size 44×44. Even though the ULAN only

 doi:10.20944/preprints202211.0243.v1

perturbs a quarter of the SAR image area, it achieves comparable attack performance to the global UAPs. We speculate the reason is that the features within target regions have stronger relevance with the recognition results than others, so a focused perturbation on the target region is more efficient than the global perturbation. In terms of the attack stealthiness, Table 3 lists the L_2-norm value of image distortion caused by each method and the SSIM between the adversarial examples and clean SAR images. An interesting phenomenon is that sometimes the ULAN causes a larger image distortion but still performs better on the SSIM metric than baseline methods. We attribute this to the fact that the human eye is more sensitive to large-range minor perturbations than small-range focused ones, resulting in the superior performance of our method on the SSIM metric. It also illustrates that local perturbations can enhance the imperceptibility of adversarial attacks.

Table 3. Non-targeted attacks of the ULAN (ours), UAN [20], U-Net, and ResNet Generator [49] against DNN models on the MSTAR dataset. We report attack results on the testing dataset.

Victim	Method	Acc			Confidence			L_2-norm	SSIM
		Clean	Adv	Gap	Clean	Adv	Gap		
A-Conv-BN	ULAN	98.19%	31.53%	-66.66%	0.93	0.31	-0.62	2.03	0.96
	UAN	98.23%	29.06%	-69.17%	0.94	0.29	-0.65	2.54	0.93
	U-Net	98.52%	28.07%	-70.45%	0.94	0.28	-0.66	2.33	0.95
	ResG	98.06%	35.04%	-63.02%	0.93	0.33	-0.60	2.07	0.94
VGG16-BN	ULAN	96.17%	16.94%	-79.23%	0.95	0.17	-0.78	2.45	0.95
	UAN	95.75%	10.47%	-85.28%	0.95	0.11	-0.84	3.63	0.86
	U-Net	95.59%	12.94%	-82.65%	0.94	0.13	-0.81	2.68	0.93
	ResG	95.63%	18.51%	-77.12%	0.95	0.18	-0.77	4.34	0.82
GoogLeNet	ULAN	97.28%	16.90%	-80.38%	0.96	0.17	-0.79	3.11	0.95
	UAN	97.11%	11.91%	-85.20%	0.96	0.12	-0.84	3.68	0.88
	U-Net	97.32%	15.87%	-81.45%	0.97	0.17	-0.80	2.87	0.93
	ResG	97.32%	19.62%	-77.70%	0.97	0.20	-0.77	2.67	0.94
InceptionV3	ULAN	92.91%	23.00%	-69.91%	0.91	0.23	-0.68	2.30	0.96
	UAN	92.87%	14.59%	-78.28%	0.92	0.15	-0.77	2.64	0.93
	U-Net	93.16%	22.59%	-70.57%	0.92	0.21	-0.71	2.30	0.95
	ResG	93.45%	21.48%	-71.97%	0.92	0.21	-0.71	2.66	0.93
ResNet50	ULAN	96.17%	16.08%	-80.09%	0.96	0.16	-0.79	3.65	0.94
	UAN	96.21%	14.39%	-81.82%	0.96	0.14	-0.82	5.57	0.73
	U-Net	95.67%	19.95%	-75.72%	0.95	0.20	-0.75	3.57	0.91
	ResG	96.08%	35.00%	-61.08%	0.96	0.35	-0.61	3.70	0.91
ResNeXt50	ULAN	96.37%	17.35%	-79.02%	0.96	0.18	-0.78	3.84	0.94
	UAN	96.78%	10.96%	-85.82%	0.96	0.11	-0.85	4.56	0.82
	U-Net	96.58%	13.27%	-83.31%	0.96	0.14	-0.82	3.43	0.91
	ResG	96.70%	17.52%	-79.18%	0.96	0.18	-0.78	3.19	0.92

In the targeted attack, we regard the target category as the correct class, so the classification accuracy of DNN models on the testing dataset reflects the data distribution, i.e., each category accounts for about one-tenth of the total dataset. According to Table 4, adversarial examples lead to a sharp rise in the model's classification accuracy, the maximum increase reaches 84%, and the minimum exceeds 70%; the rise of target class confidence varies from 0.67 to 0.83. It means that the generated UAPs can induce DNN models to output specified results with high confidence. Meanwhile, for the same victim model, the ULAN is slightly inferior to the UAN and U-Net on the attack effectiveness but performs much better than baseline methods on the attack stealthiness. Thus, we believe that given a fixed SSIM value, the ULAN can achieve the best attack performance.

To visualize the adversarial examples generated by different methods, we take the VGG16-BN-based SAR-ATR model as the victim network, and display the adversarial examples for the non-targeted and targeted attacks in Figure 8 and Figure 9, respectively. We list the prediction and confidence output by the victim model at the top of each adversarial example, and the bottom of each figure shows the sizes of the corresponding image and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

perturbation. As we can see, the UAPs generated by baseline methods fully cover the SAR images fed to the model, while the ULAN can locate and perturb the target (green box) region effectively. Meanwhile, according to Figure 8 and Figure 9, there are apparent shadow and texture traces in the adversarial examples crafted by baseline methods, which also suggests that the global perturbations are more perceptible than the local ones. In summary, compared to baseline methods, our method can achieve good attack performance with smaller perturbed regions and lower perceptions.

Table 4. Targeted attacks of the ULAN (ours), UAN [20], U-Net, and ResNet Generator [49] against DNN models on the MSTAR dataset. We report attack results on the testing dataset.

Victim	Method	Acc			Confidence			L_2-norm	SSIM
		Clean	Adv	Gap	Clean	Adv	Gap		
A-Conv-BN	ULAN	9.99%	85.45%	+75.46%	0.10	0.81	+0.71	4.28	0.90
	UAN	9.97%	90.73%	+80.76%	0.10	0.87	+0.77	3.56	0.88
	U-Net	9.99%	91.63%	+81.64%	0.10	0.88	+0.78	3.71	0.87
	ResG	9.98%	90.23%	+80.25%	0.10	0.87	+0.77	3.94	0.87
VGG16-BN	ULAN	9.98%	90.21%	+80.23%	0.10	0.89	+0.79	4.71	0.90
	UAN	10.02%	93.84%	+83.82%	0.10	0.93	+0.83	4.99	0.80
	U-Net	10.02%	94.15%	+84.13%	0.10	0.93	+0.83	5.09	0.82
	ResG	10.05%	88.19%	+78.14%	0.10	0.86	+0.76	7.25	0.69
GoogLeNet	ULAN	10.02%	81.65%	+71.63%	0.10	0.80	+0.70	4.47	0.92
	UAN	10.03%	90.70%	+80.67%	0.10	0.90	+0.80	4.80	0.85
	U-Net	10.00%	91.33%	+81.33%	0.10	0.89	+0.79	4.64	0.88
	ResG	10.02%	77.74%	+67.72%	0.10	0.77	+0.67	4.99	0.83
InceptionV3	ULAN	10.05%	80.06%	+70.01%	0.10	0.79	+0.69	4.08	0.93
	UAN	9.98%	91.10%	+81.12%	0.10	0.90	+0.80	4.87	0.84
	U-Net	9.95%	91.77%	+81.82%	0.10	0.90	+0.80	4.91	0.87
	ResG	9.90%	84.27%	+74.37%	0.10	0.82	+0.72	4.99	0.84
ResNet50	ULAN	9.95%	85.31%	+75.36%	0.10	0.84	+0.74	5.54	0.90
	UAN	10.09%	90.41%	+80.32%	0.10	0.90	+0.80	5.46	0.80
	U-Net	10.01%	87.58%	+77.57%	0.10	0.87	+0.77	5.34	0.83
	ResG	10.04%	88.08%	+78.04%	0.10	0.87	+0.77	6.70	0.75
ResNeXt50	ULAN	9.98%	86.53%	+76.55%	0.10	0.86	+0.76	5.15	0.90
	UAN	9.98%	91.77%	+81.79%	0.10	0.91	+0.81	5.60	0.79
	U-Net	10.00%	91.88%	+81.88%	0.10	0.91	+0.81	5.28	0.83
	ResG	9.98%	83.89%	+73.91%	0.10	0.83	+0.73	6.73	0.75

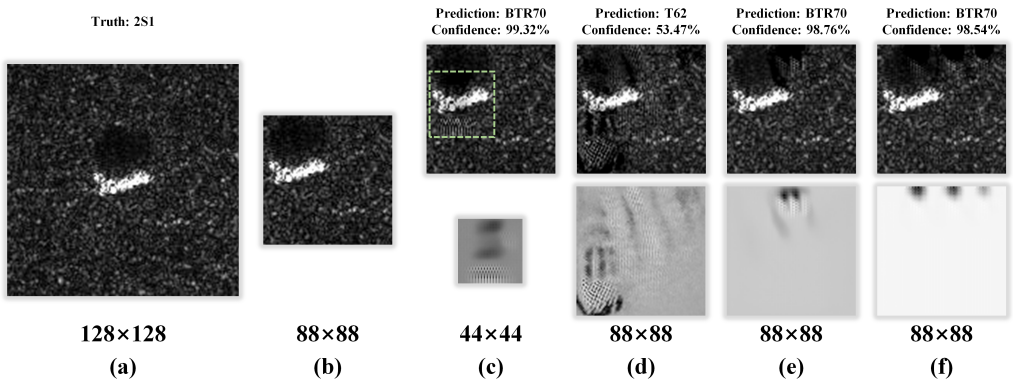

Figure 8. (a) The original SAR image in MSTAR. (b) The clean SAR image fed to the model. The first row shows the adversarial examples for non-targeted attacks, and the second row shows the UAPs generated by different methods, corresponding to ULAN (c), UAN (d), U-Net (e), and ResNet Generator (f), respectively. We list the prediction and confidence output by the victim model at the top of each adversarial example, and the bottom of the figure shows the sizes of the corresponding image and perturbation.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

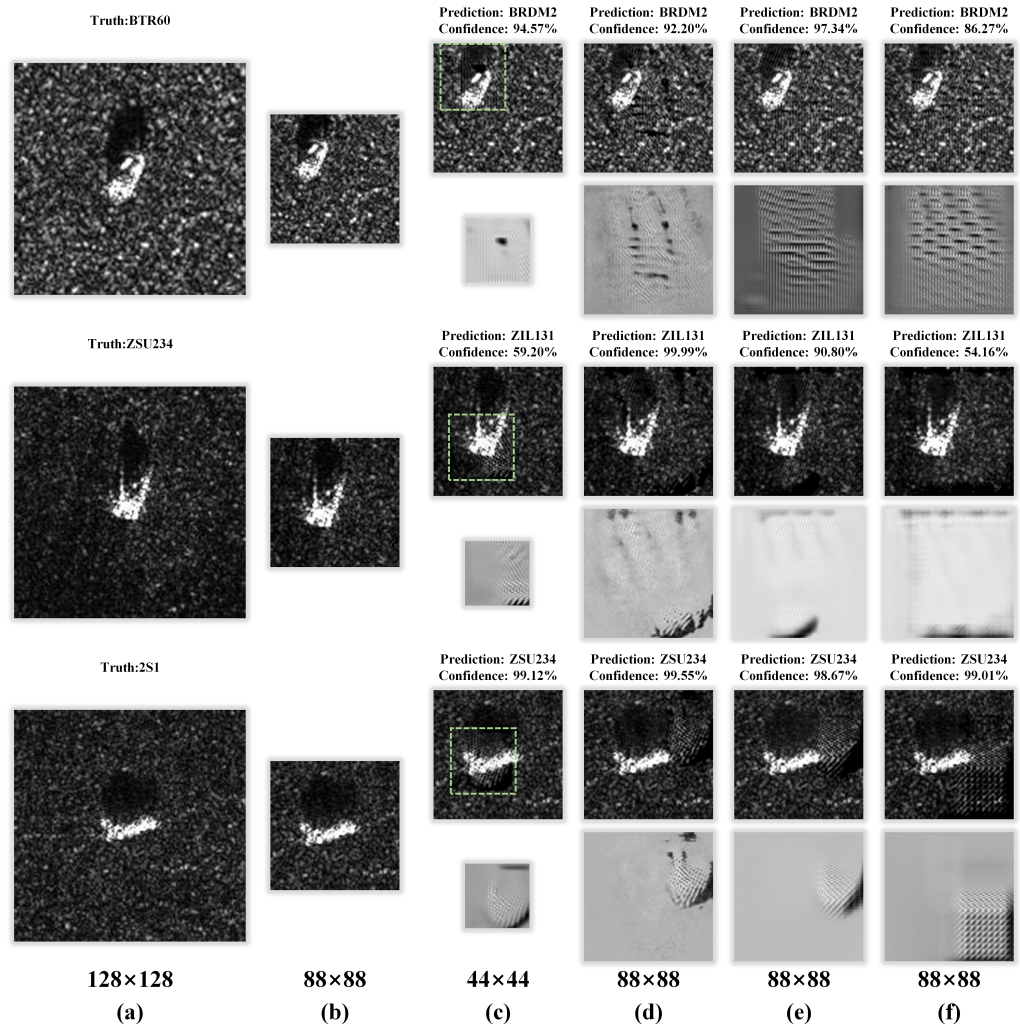

Figure 9. (a) The original SAR image in MSTAR. (b) The clean SAR image fed to the model. From top to bottom, the corresponding target classes are BRDM2, ZIL131, and ZSU234. For each target class, the first row shows the adversarial examples for targeted attacks, and the second row shows the UAPs generated by different methods, corresponding to ULAN (c), UAN (d), U-Net (e), and ResNet Generator (f), respectively. We list the prediction and confidence output by the victim model at the top of each adversarial example, and the bottom of the figure shows the sizes of the corresponding image and perturbation.

4.5. Adversarial Attacks with Perturbation Offset

We now evaluate the adversarial attacks in the case of perturbation offset. Specifically, we first recover the adversarial examples generated in Section 4.4 to 128×128, and next random-crop the recovered images to 88×88 again, such that the perturbation offset condition is constructed. The results of non-targeted and targeted attacks in the case of perturbation offset are shown in Table 5 and Table 6, respectively.

The experimental results suggest that perturbation offset severely impacts the attack performance of baseline methods. In non-targeted attacks, the classification accuracy of DNN models rises sharply, the maximum increase reaches 40%, and the minimum exceeds 10%; the rise of target class confidence varies from 0.10 to 0.38. A similar situation also occurs in targeted attacks, where the UAPs generated by baseline methods are likely to be ineffective in the case of perturbation offset. The decrease of the classification accuracy varies from 30% to 48%, and the drop in target class confidence varies from 0.31 to 0.49. In contrast, the attack performance of our method is hardly affected under the same experimental condition. The detailed experimental data is displayed in Table 5 and Table 6.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

Table 5. Non-targeted attacks against DNN models in the case of perturbation offset. We report attack results on the testing dataset.

Victim	Method	Acc			Confidence		
		No-offset	Offset	Gap	No-offset	Offset	Gap
A-Conv-BN	ULAN	31.53%	32.09%	+0.56%	0.31	0.33	+0.02
	UAN	29.06%	59.65%	+30.59%	0.29	0.53	+0.24
	U-Net	28.07%	51.48%	+23.41%	0.28	0.47	+0.19
	ResG	35.04%	60.10%	+25.06%	0.33	0.55	+0.22
VGG16-BN	ULAN	16.94%	17.31%	+0.37%	0.17	0.18	+0.01
	UAN	10.47%	26.26%	+15.79%	0.11	0.26	+0.15
	U-Net	12.94%	46.25%	+33.31%	0.13	0.46	+0.33
	ResG	18.51%	29.39%	+10.88%	0.18	0.29	+0.11
GoogLeNet	ULAN	16.90%	18.92%	+2.02%	0.17	0.19	+0.02
	UAN	11.91%	40.40%	+28.49%	0.12	0.39	+0.27
	U-Net	15.87%	43.57%	+27.70%	0.17	0.43	+0.26
	ResG	19.62%	48.76%	+29.14%	0.20	0.48	+0.28
InceptionV3	ULAN	23.00%	23.50%	+0.50%	0.23	0.24	+0.01
	UAN	14.59%	36.31%	+21.72%	0.15	0.35	+0.20
	U-Net	22.59%	46.17%	+23.58%	0.21	0.44	+0.23
	ResG	21.48%	39.20%	+17.72%	0.21	0.38	+0.17
ResNet50	ULAN	16.08%	16.24%	+0.16%	0.16	0.16	+0.00
	UAN	14.39%	23.87%	+9.48%	0.14	0.24	+0.10
	U-Net	19.95%	50.62%	+30.67%	0.20	0.50	+0.30
	ResG	35.00%	51.65%	+16.65%	0.35	0.51	+0.16
ResNeXt50	ULAN	17.35%	17.44%	+0.09%	0.18	0.18	+0.00
	UAN	10.96%	41.59%	+30.63%	0.11	0.41	+0.30
	U-Net	13.27%	46.21%	+32.94%	0.14	0.46	+0.32
	ResG	17.52%	57.05%	+39.53%	0.18	0.56	+0.38

Table 6. Targeted attacks against DNN models in the case of perturbation offset. We report attack results on the testing dataset.

Victim	Method	Acc			Confidence		
		No-offset	Offset	Gap	No-offset	Offset	Gap
A-Conv-BN	ULAN	85.45%	82.23%	-3.22%	0.81	0.79	-0.02
	UAN	90.73%	43.80%	-46.93%	0.87	0.42	-0.45
	U-Net	91.63%	45.49%	-46.14%	0.88	0.43	-0.45
	ResG	90.23%	44.28%	-45.95%	0.87	0.42	-0.45
VGG16-BN	ULAN	90.21%	86.72%	-3.49%	0.89	0.86	-0.03
	UAN	93.84%	56.72%	-37.12%	0.93	0.56	-0.37
	U-Net	94.15%	58.46%	-35.69%	0.93	0.58	-0.35
	ResG	88.19%	51.23%	-36.96%	0.86	0.51	-0.35
GoogLeNet	ULAN	81.65%	78.71%	-2.94%	0.80	0.78	-0.02
	UAN	90.70%	49.18%	-41.52%	0.90	0.49	-0.41
	U-Net	91.33%	44.81%	-46.52%	0.89	0.44	-0.45
	ResG	77.74%	46.76%	-30.98%	0.77	0.46	-0.31
InceptionV3	ULAN	80.06%	73.54%	-6.52%	0.79	0.72	-0.07
	UAN	91.10%	41.56%	-49.54%	0.90	0.41	-0.49
	U-Net	91.77%	44.46%	-47.31%	0.90	0.44	-0.46
	ResG	84.27%	42.35%	-41.92%	0.82	0.41	-0.41
ResNet50	ULAN	85.31%	82.32%	-2.99%	0.84	0.81	-0.03
	UAN	90.41%	43.48%	-46.93%	0.90	0.43	-0.47
	U-Net	87.58%	40.07%	-47.51%	0.87	0.40	-0.47
	ResG	88.08%	46.32%	-41.76%	0.87	0.46	-0.41
ResNeXt50	ULAN	86.53%	82.82%	-3.71%	0.86	0.82	-0.04
	UAN	91.77%	50.22%	-41.55%	0.91	0.50	-0.41
	U-Net	91.88%	50.59%	-41.29%	0.91	0.50	-0.41
	ResG	83.89%	41.28%	-42.61%	0.83	0.41	-0.42

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

In summary, the global UAPs generated by baseline methods are vulnerable to perturbation offset. They might be ineffective unless the victim model accurately takes the perturbed region as input. However, the local perturbations generated by the ULAN only cover the target regions of SAR images so that they can be fully fed to the model along with the targets regardless of the input regions, which fundamentally prevents perturbation offset.

4.6. Adversarial Attacks under Small Sample Conditions

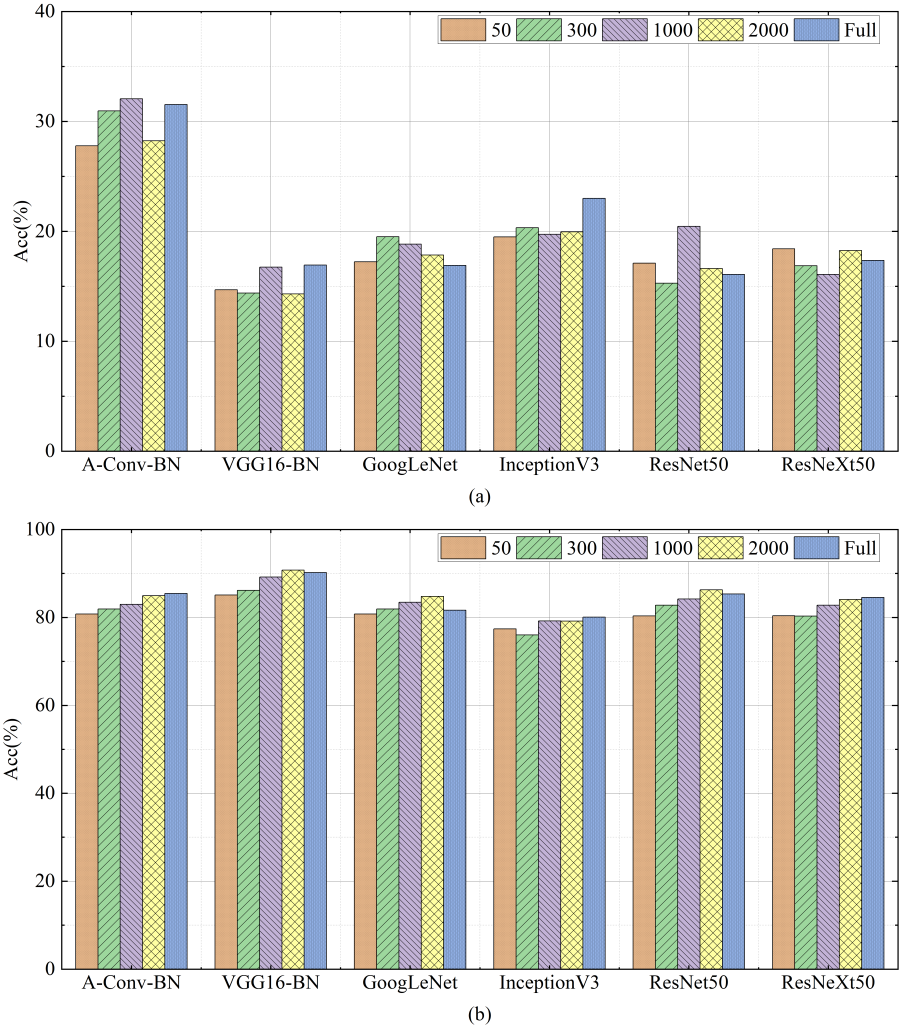

Figure 10. The attack results of the ULAN against DNN models on MSTAR. (a): Non-targeted attacks; (b): Targeted attacks. We vary the number of images the ULAN is trained on, and report results on the testing dataset.

So far, we have assumed attackers share full access to any images used to train the victim model. However, the professionalism and confidentiality of SAR images make them challenging to access in practice. In other words, it is difficult for attackers to obtain sufficient data to support the training of the ULAN. Therefore, we now evaluate the attack performance of our method under stronger assumptions of attacker access to training data.

Figure 10 shows the non-targeted (a) and targeted (b) attack results of the ULAN trained on subsets of the MSTAR training dataset. Specifically, we uniformly sample 50, 300, 1000, and 2000 images from the full training dataset to form the subsets and evaluate the attack performance of the ULAN trained on subsets against different DNNs. As we can see, for the same victim model, the difference in the attack performance of the ULAN trained

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

on 50 images (5 per class) and the full training dataset is less than 5% − in other words, there is virtually no fluctuation in the attack performance when the amount of training data changes. The possible reason why the proposed method maintains good performance even with few training samples might be due to the skip connection structure of the network and the fixation structure of the SAR image. The decoder of the ULAN fuses the features from different layers through the skip connection structure, which can help the generator learn the data distribution sufficiently. Moreover, the low dependence on the training data also attributes to the fixation structure of the SAR image itself such that its semantic features are easier extracted and represented than natural images. Thus, the proposed method can work well under small sample conditions.

4.7. Influence of Parameters

This section evaluates the attack performance of the ULAN trained on different parameter settings, providing guidance for attackers to achieve superior attack performance. The parameters mainly include the perturbation size $w \times h$, the weight coefficient ω, and the type of L_p-norm.

4.7.1. Perturbation Size $w \times h$

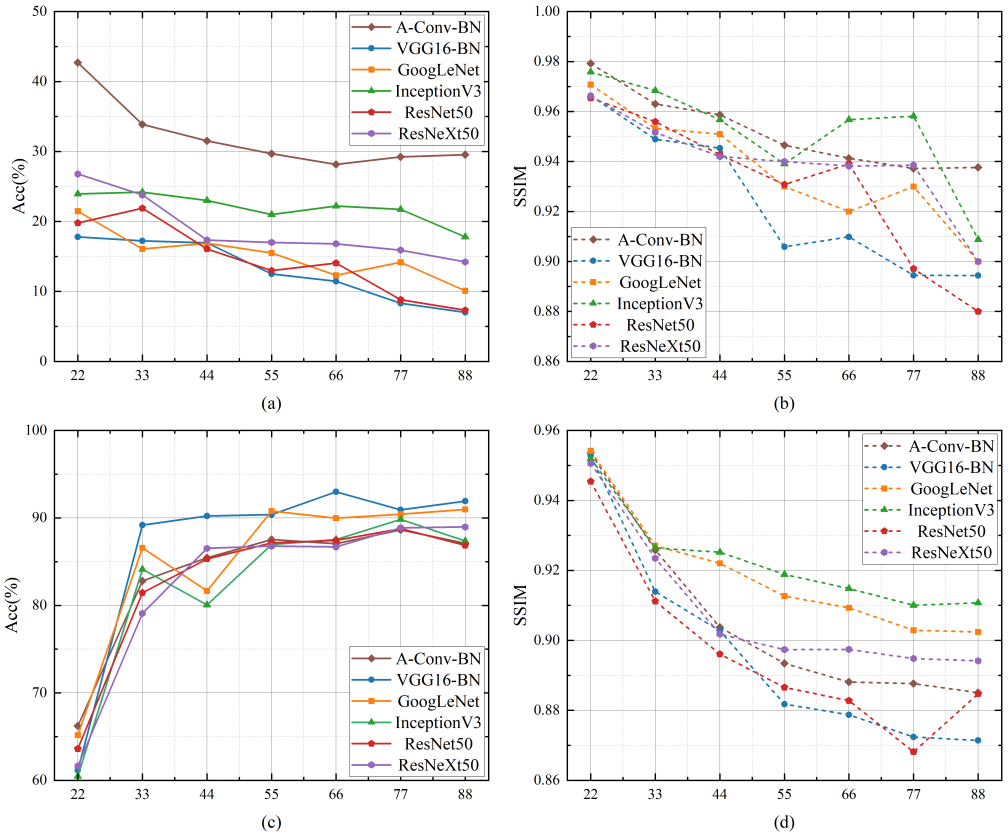

Figure 11. The influence of the perturbation size $w \times h$ on the attack performance. The Acc and SSIM metrics of non-targeted attacks are shown in (a) and (b), and the corresponding metrics of targeted attacks are shown in (c) and (d).

To investigate the influence of the perturbation size $w \times h$ on the attack performance, we train the ULAN on seven different size settings: 22×22, 33×33, 44×44, 55×55, 66×66, 77×77, and 88×88. Then, we evaluate the attack performance on the testing dataset, and the results are shown in Figure 11. As expected, for both non-targeted and targeted attacks, a larger perturbation size improves the attack effectiveness, while the attack stealthiness is getting worse. Meanwhile, we find that when the perturbation size

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 November 2022 doi:10.20944/preprints202211.0243.v1

exceeds 55×55, the SSIM metric of each DNN model shown in Figure 11(b) and 11(d) is continuous decreasing, while the corresponding Acc metric shown in Figure 11(a) and 11(c) tends to a stable value. We speculate the reason is that perturbation offset will inevitably occur as the perturbation size increases, resulting in only partial perturbations can be fed to the victim model such that the attack effectiveness is no longer improved. Therefore, the advised perturbation size in this paper is between 44×44 and 55×55.

Furthermore, the ULAN has superior attack performance even in the case of perturbation offset, which is quite different from baseline methods. Specifically, according to Table 5 and Table 6, a large number of global UAPs generated by baseline methods fail to attack the victim model in the case of perturbation offset. Yet, when the perturbation size reaches 88×88, more than 80% of the adversarial examples generated by the ULAN still work well. This is because the perturbation size is too large to prevent perturbation offset during the training phase. In other words, the ULAN itself is trained in the case of perturbation offset. Thus, there is no doubt that a well-trained ULAN has already equipped with the ability to fool models effectively in the case of perturbation offset.

4.7.2. Weight Coefficient ω

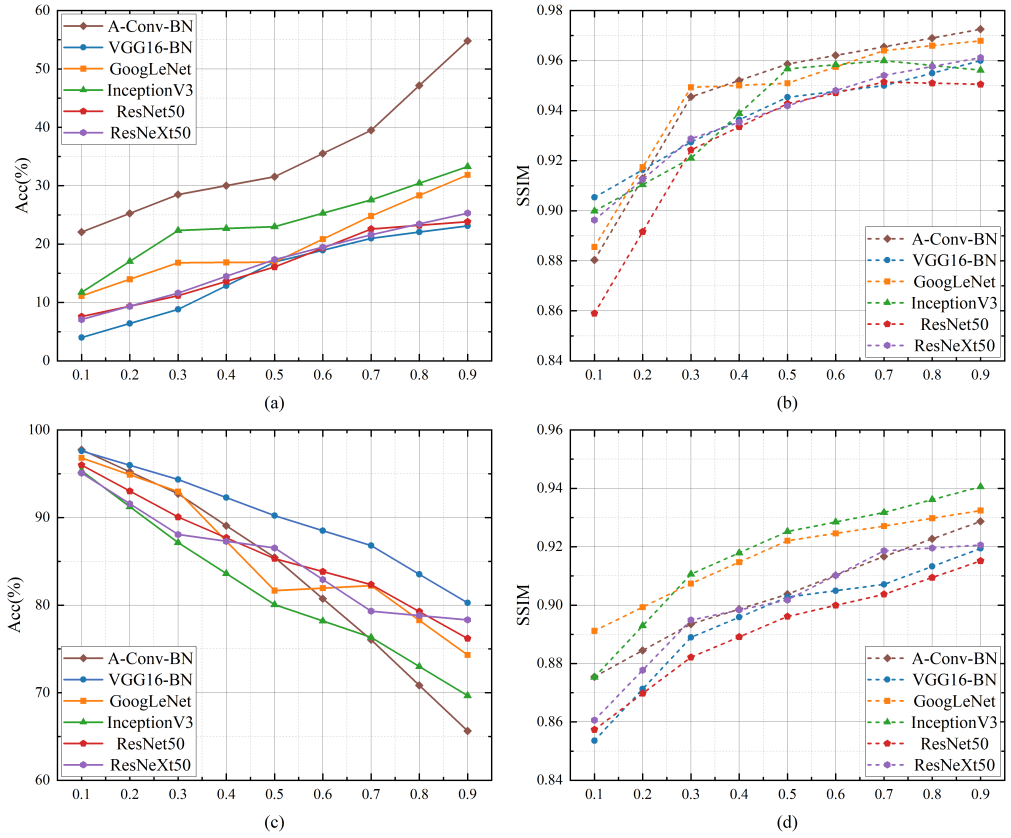

Figure 12. The influence of the weight coefficient ω on the attack performance. The Acc and SSIM metrics of non-targeted attacks are shown in (a) and (b), and the corresponding metrics of targeted attacks are shown in (c) and (d).

The weight coefficient ω is a constant measuring the relative importance of attack effectiveness and stealthiness, which has a great impact on the attack performance. We now train the ULAN on nine different weight coefficients: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9, and report attack results on the testing dataset in Figure 12. As we can see, for both non-targeted and targeted attacks, the attack stealthiness is improved as ω increasing, while the attack effectiveness is getting worse. Meanwhile, Figure 12(a) and 12(c) suggest that the Acc metric of each DNN model cannot converge to a stable value, and the corresponding

 doi:10.20944/preprints202211.0243.v1

SSIM metric shown in Figure 12(b) and 12(d) is also constantly changing. Thus, for superior attack performance, attackers are supposed to choose an appropriate weight as needed in the training phase of the ULAN.

4.7.3. Type of L_p-norm

So far, we have adopted the L_2-norm to measure the image distortion caused by adversarial attacks. However, in addition to the L_2-norm, there are many distance metrics, such as the L_∞-norm and the L_1-norm, etc. In this section, we evaluate the attack performance of the ULAN trained on different distance metrics: the L_2-norm and the L_∞-norm. Note that the values of image distortion calculated by the two metrics differ by several orders of magnitude, so we set the weight ω of L_2-norm to 0.5 and 10 for the L_∞-norm. The results of non-targeted and targeted attacks are shown in Table 7 and Table 8, respectively. We can find that the ULAN trained on the L_2-norm has better performance on both the attack effectiveness and stealthiness. Therefore, to obtain a more threatening attack network, the advised distance metric in this paper is the L_2-norm.

Table 7. The non-targeted attacks that adopt different type of L_p-norm as the distance metric, and we report attack results on the testing dataset.

Victim	Acc		SSIM	
	L_2-norm	L_∞-norm	L_2-norm	L_∞-norm
A-Conv-BN	31.53%	28.85%	0.96	0.92
VGG16-BN	16.94%	21.19%	0.95	0.88
GoogLeNet	16.90%	17.60%	0.95	0.91
InceptionV3	23.00%	24.65%	0.96	0.91
ResNet50	16.08%	14.10%	0.94	0.88
ResNeXt50	17.35%	18.43%	0.94	0.90

Table 8. The targeted attacks that adopt different type of L_p-norm as the distance metric, and we report attack results on the testing dataset.

Victim	Acc		SSIM	
	L_2-norm	L_∞-norm	L_2-norm	L_∞-norm
A-Conv-BN	85.45%	84.33%	0.90	0.85
VGG16-BN	90.21%	87.25%	0.90	0.83
GoogLeNet	81.65%	81.39%	0.92	0.85
InceptionV3	80.06%	78.72%	0.93	0.86
ResNet50	85.31%	83.03%	0.90	0.82
ResNeXt50	86.53%	82.07%	0.90	0.85

5. Conclusions

In this paper, a semi-whitebox attack network called universal local adversarial network is proposed to generate UAPs for the target regions of SAR images, with the benefit of focusing perturbations on the target regions in SAR images that have high relevance to the recognition results. A focused perturbation on the high-relevance target region significantly improves the efficiency of adversarial attacks. Also, it ensures that the well-designed perturbations can be fully fed to the victim model along with the targets such that perturbation offset is fundamentally prevented. To satisfy the feasibility requirement of adversarial attacks, once the ULAN is trained, it can real-time generate adversarial examples for the DNN-based SAR-ATR model without requiring access to the model itself anymore, and thus possesses high potential in practical applications. Experimental results demonstrate that the proposed method prevents perturbation offset effectively and achieves comparable attack performance to the conventional global UAPs by perturbing only a quarter or less of the SAR image area. Moreover, our experiments also indicate that the ULAN is insensitive

to the amount of training data, which makes it still work well under small sample conditions. Potential future work could consider replacing the victim model with a distillation model to construct a black-box attack network. It is also of great interest to enhance the transferability of adversarial examples between different DNN models.

Author Contributions: Conceptualization, M.D. and D.B.; methodology, M.D.; software, M.D.; validation, D.B., X.X. and Z.W.; formal analysis, D.B. and M.D.; investigation, M.D.; resources, D.B.; data curation, M.D.; writing—original draft preparation, M.D.; writing—review and editing, M.D. and D.B.; visualization, M.D.; supervision, D.B.; project administration, D.B.; funding acquisition, D.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant 62071479.

Institutional Review Board Statement: The study does not involve humans or animals.

Informed Consent Statement: The study does not involve humans.

Data Availability Statement: The experiment in this paper uses a public data set, so no data is reported in this work.

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflicts of Interest: The authors declare that they have no conflict of interest to report regarding the present study.

References

1. Zhang, F.; Yao, X.; Tang, H.; Yin, Q.; Hu, Y.; Lei, B. Multiple mode SAR raw data simulation and parallel acceleration for Gaofen-3 mission. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2018**, *11*, 2115–2126. [Crossref]
2. Brown, W.M. Synthetic aperture radar. *IEEE Transactions on Aerospace and Electronic Systems* **1967**, pp. 217–229. [Crossref]
3. Moreira, A.; Prats-Iraola, P.; Younis, M.; Krieger, G.; Hajnsek, I.; Papathanassiou, K.P. A tutorial on synthetic aperture radar. *IEEE Geoscience and remote sensing magazine* **2013**, *1*, 6–43. [Crossref]
4. Zhang, Z.; Wang, H.; Xu, F.; Jin, Y.Q. Complex-valued convolutional neural network and its application in polarimetric SAR image classification. *IEEE Transactions on Geoscience and Remote Sensing* **2017**, *55*, 7177–7188. [Crossref]
5. Chen, S.; Wang, H.; Xu, F.; Jin, Y.Q. Target classification using the deep convolutional networks for SAR images. *IEEE transactions on geoscience and remote sensing* **2016**, *54*, 4806–4817. [Crossref]
6. Ding, J.; Chen, B.; Liu, H.; Huang, M. Convolutional neural network with data augmentation for SAR target recognition. *IEEE Geoscience and remote sensing letters* **2016**, *13*, 364–368. [Crossref]
7. Du, C.; Chen, B.; Xu, B.; Guo, D.; Liu, H. Factorized discriminative conditional variational auto-encoder for radar HRRP target recognition. *Signal Processing* **2019**, *158*, 176–189. [Crossref]
8. Vint, D.; Anderson, M.; Yang, Y.; Ilioudis, C.; Di Caterina, G.; Clemente, C. Automatic Target Recognition for Low Resolution Foliage Penetrating SAR Images Using CNNs and GANs. *Remote Sensing* **2021**, *13*, 596. [Crossref]
9. Huang, T.; Zhang, Q.; Liu, J.; Hou, R.; Wang, X.; Li, Y. Adversarial attacks on deep-learning-based SAR image target recognition. *Journal of Network and Computer Applications* **2020**, *162*, 102632. [Crossref]
10. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* **2013**. [Crossref]
11. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* **2014**. [Crossref]
12. Kurakin, A.; Goodfellow, I.J.; Bengio, S. Adversarial examples in the physical world. In *Artificial intelligence safety and security*; Chapman and Hall/CRC, 2018; pp. 99–112.
13. Moosavi-Dezfooli, S.M.; Fawzi, A.; Frossard, P. Deepfool: a simple and accurate method to fool deep neural networks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2574–2582. [Crossref]
14. Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z.B.; Swami, A. The limitations of deep learning in adversarial settings. In Proceedings of the 2016 IEEE European symposium on security and privacy (EuroS&P). IEEE, 2016, pp. 372–387. [Crossref]
15. Su, J.; Vargas, D.V.; Sakurai, K. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation* **2019**, *23*, 828–841. [Crossref]
16. Chen, P.Y.; Zhang, H.; Sharma, Y.; Yi, J.; Hsieh, C.J. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In Proceedings of the Proceedings of the 10th ACM workshop on artificial intelligence and security, 2017, pp. 15–26. [Crossref]

17. Chen, J.; Jordan, M.I.; Wainwright, M.J. Hopskipjumpattack: A query-efficient decision-based attack. In Proceedings of the 2020 ieee symposium on security and privacy (sp). IEEE, 2020, pp. 1277–1294. [Crossref]
18. Xie, C.; Zhang, Z.; Zhou, Y.; Bai, S.; Wang, J.; Ren, Z.; Yuille, A.L. Improving transferability of adversarial examples with input diversity. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2730–2739. [Crossref]
19. Moosavi-Dezfooli, S.M.; Fawzi, A.; Fawzi, O.; Frossard, P. Universal adversarial perturbations. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1765–1773. [Crossref]
20. Hayes, J.; Danezis, G. Learning universal adversarial perturbations with generative models. In Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW). IEEE, 2018, pp. 43–49. [Crossref]
21. Mopuri, K.R.; Garg, U.; Babu, R.V. Fast feature fool: A data independent approach to universal adversarial perturbations. *arXiv preprint arXiv:1707.05572* **2017**. [Crossref]
22. Mopuri, K.R.; Uppala, P.K.; Babu, R.V. Ask, acquire, and attack: Data-free uap generation using class impressions. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 19–34. [Crossref]
23. Xu, Y.; Du, B.; Zhang, L. Assessing the threat of adversarial examples on deep neural networks for remote sensing scene classification: Attacks and defenses. *IEEE Transactions on Geoscience and Remote Sensing* **2020**, *59*, 1604–1617. [Crossref]
24. Xu, Y.; Ghamisi, P. Universal Adversarial Examples in Remote Sensing: Methodology and Benchmark. *IEEE Transactions on Geoscience and Remote Sensing* **2022**, *60*, 1–15. [Crossref]
25. Thys, S.; Van Ranst, W.; Goedemé, T. Fooling automated surveillance cameras: adversarial patches to attack person detection. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2019, pp. 0–0. [Crossref]
26. Li, H.; Huang, H.; Chen, L.; Peng, J.; Huang, H.; Cui, Z.; Mei, X.; Wu, G. Adversarial examples for CNN-based SAR image classification: An experience study. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2020**, *14*, 1333–1347. [Crossref]
27. Du, C.; Huo, C.; Zhang, L.; Chen, B.; Yuan, Y. Fast C&W: A Fast Adversarial Attack Algorithm to Fool SAR Target Recognition with Deep Convolutional Neural Networks. *IEEE Geoscience and Remote Sensing Letters* **2021**, *19*, 1–5. [Crossref]
28. Wang, L.; Wang, X.; Ma, S.; Zhang, Y. Universal adversarial perturbation of SAR images for deep learning based target classification. In Proceedings of the 2021 IEEE 4th International Conference on Electronics Technology (ICET). IEEE, 2021, pp. 1272–1276. [Crossref]
29. Xia, W.; Liu, Z.; Li, Y. SAR-PeGA: A Generation Method of Adversarial Examples for SAR Image Target Recognition Network. *IEEE Transactions on Aerospace and Electronic Systems* **2022**. [Crossref]
30. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241. [Crossref]
31. Chen, S.; He, Z.; Sun, C.; Yang, J.; Huang, X. Universal adversarial attack on attention and the resulting dataset damagenet. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2020**. [Crossref]
32. Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.R.; Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one* **2015**, *10*, e0130140. [Crossref]
33. Xiao, C.; Li, B.; Zhu, J.Y.; He, W.; Liu, M.; Song, D. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610* **2018**. [Crossref]
34. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM* **2017**, *60*, 84–90. [Crossref]
35. Howard, A.; Zhmoginov, A.; Chen, L.C.; Sandler, M.; Zhu, M. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation **2018**. [Crossref]
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778. [Crossref]
37. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**. [Crossref]
38. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the European conference on computer vision. Springer, 2014, pp. 818–833.
39. Zhou, J.; Troyanskaya, O.G. Predicting effects of noncoding variants with deep learning–based sequence model. *Nature methods* **2015**, *12*, 931–934. [Crossref]
40. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034* **2013**. [Crossref]
41. Teague, M.R. Image analysis via the general theory of moments. *Josa* **1980**, *70*, 920–930.
42. Otsu, N. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics* **1979**, *9*, 62–66. [Crossref]
43. Keydel, E.R.; Lee, S.W.; Moore, J.T. MSTAR extended operating conditions: A tutorial. *Algorithms for Synthetic Aperture Radar Imagery III* **1996**, *2757*, 228–242. [Crossref]

44. Junfan, Z.; Hao, S.; Lin, L.; Kefeng, J.; Gangyao, K. Sparse Adversarial Attack of SAR Image. *Journal of Signal Processing* **2021**, *37*, 11.
45. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9. [Crossref]
46. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826. [Crossref]
47. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1492–1500. [Crossref]
48. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**. [Crossref]
49. Poursaeed, O.; Katsman, I.; Gao, B.; Belongie, S. Generative adversarial perturbations. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4422–4431. [Crossref]
50. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **2004**, *13*, 600–612. [Crossref]