remote sensing

MDPI

Article

Rust-Style Patch: A Physical and Naturalistic Camouflage Attacks on Object Detector for Remote Sensing Images

Binyue Deng [1,†], Denghui Zhang [1,†], Fashan Dong [1], Junjian Zhang [1], Muhammad Shafiq [1] and Zhaoquan Gu [2,3,*]

[1] Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510006, China
[2] School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China
[3] Department of New Networks, Peng Cheng Laboratory, Shenzhen 518055, China
* Correspondence: guzhaoquan@hit.edu.cn
† These authors contributed equally to this work.

Abstract: Deep neural networks (DNNs) can improve the image analysis and interpretation of remote sensing technology by extracting valuable information from images, and has extensive applications such as military affairs, agriculture, environment, transportation, and urban division. The DNNs for object detection can identify and analyze objects in remote sensing images through fruitful features of images, which improves the efficiency of image processing and enables the recognition of large-scale remote sensing images. However, many studies have shown that deep neural networks are vulnerable to adversarial attack. After adding small perturbations, the generated adversarial examples will cause deep neural network to output undesired results, which will threaten the normal recognition and detection of remote sensing systems. According to the application scenarios, attacks can be divided into the digital domain and the physical domain, the digital domain attack is directly modified on the original image, which is mainly used to simulate the attack effect, while the physical domain attack adds perturbation to the actual objects and captures them with device, which is closer to the real situation. Attacks in the physical domain are more threatening, however, existing attack methods generally generate the patch with bright style and a large attack range, which is easy to be observed by human vision. Our goal is to generate a natural patch with a small perturbation area, which can help some remote sensing images used in the military to avoid detection by object detectors and im-perceptible to human eyes. To address the above issues, we generate a rust-style adversarial patch generation framework based on style transfer. The framework takes a heat map-based interpretability method to obtain key areas of target recognition and generate irregular-shaped natural-looking patches to reduce the disturbance area and alleviates suspicion from humans. To make the generated adversarial examples have a higher attack success rate in the physical domain, we further improve the robustness of the adversarial patch through data augmentation methods such as rotation, scaling, and brightness, and finally, make it impossible for the object detector to detect the camouflage patch. We have attacked the YOLOV3 detection network on multiple datasets. The experimental results show that our model has achieved a success rate of 95.7% in the digital domain. We also conduct physical attacks in indoor and outdoor environments and achieve an attack success rate of 70.6% and 65.3%, respectively. The structural similarity index metric shows that the adversarial patches generated are more natural than existing methods.

Keywords: adversarial attack; object detection; style transfer; deep learning

Citation: Deng, B.; Zhang, D.; Dong, F.; Zhang, J.; Shafiq, M.; Gu, Z. Rust-Style Patch: A Physical and Naturalistic Camouflage Attacks on Object Detector for Remote Sensing Images. *Remote Sens.* **2023**, *15*, 885. https://doi.org/10.3390/rs15040885

Academic Editors: Pia Addabbo, Silvia Liberata Ullo and Parameshachari Bidare Divakarachari

Received: 15 December 2022
Revised: 23 January 2023
Accepted: 29 January 2023
Published: 5 February 2023

1. Introduction

With the rapid development of artificial intelligence (AI), deep learning is widely used in the field of computer vision, such as image classification, object detection [1,2], semantic segmentation [3], etc. DNN has achieved its outstanding advantages in remote sensing technologies [4]. When acquiring a large number of remote sensing images from satellite

platforms for providing large geographic information for various applications, we can take DNN to extract valuable information, such as environmental monitoring, urban planning, or in the military. Deep learning-based object detectors are equipped on drone surveillance footage to detect military assets on the ground. However, many researchers [5–8] have found that if some imperceptible perturbations are added to the input samples, it is likely that the neural network will misidentify the adversarial examples as other classifications. The neural network is vulnerable, it can easily be misled by adversarial examples. If an attacker deliberately attacks the neural network in practical applications, it will pose serious security risks. To improve the performance of models, remote sensing detection networks can perform adversarial training and adversarial learning [9]. The adversarial learning model [6] has better generalization capabilities, enabling the detection system to capture information in complex remote sensing scenarios and better learn the data distribution of adversarial examples, thereby improving the detection robustness and accuracy of the network.

Adversarial attacks on neural networks [10,11] can be divided into attacks in the digital domain and attacks in the physical domain. Digital domain attacks directly perturb the input data and then use the modified image as adversarial examples, the generated perturbations are added to the digital image without generating the physical medium so that the computer can directly recognize the perturbations and generate wrong classification results. The physical domain attack applies the attack method in the real world. The attack in the physical domain is to print out the examples after they are generated in the digital domain and place them in the physical world to be captured by systems such as cameras, which will cause the model to make wrong judgments. Compared to the fixed and confirmed background in the digital domain, various environmental conditions in the physical domain are random and unpredictable. Although adversarial attacks in the digital domain have achieved high success rates [8,10,12–16], it is difficult to achieve the same attack performance in the physical domain with small perturbations that successfully attack in the digital domain, due to factors such as viewing angle and camera disturbance, these tiny disturbances are difficult to be captured by the camera successfully, which limits the connection between adversarial examples and the real world. So, it is proposed in [11] that we can use the adversarial patch to attack the physical domain. The adversarial patch does not limit the size of the perturbation and can be effectively reproduced in the real world. In subsequent work [17–20], researchers also confirmed the strong robustness of adversarial patches.

Our main goal is to generate a natural and imperceptible patch with a strong attack effect, then add the patch to the original images to generate adversarial example, so that the object detector cannot detect the targets. Since remote sensing images are widely used in high-security requirements fields such as national defense, it will cause greater harm if the vehicles are attacked in this case. On the one hand, as for military assets such as vehicles or aircraft, we not only want them not to be detected by the object detector algorithm but also want the patch used for the attack to look unobtrusive to the human eye. On the other hand, we can use the generated adversarial examples to conduct adversarial training on our object detector. The trained network will have the generalization ability for this type of adversarial example, thereby improving the recognition accuracy of the detector and the ability to defend such adversarial examples.

At present, the work of attacking neural networks is mainly concentrated on image classification, and the classification model only needs to detect the category of a single object. However, such as remote sensing systems, object detection algorithms are often adapted to detect aircraft and ships in remote sensing images instead of image classification algorithms in practical applications. Compared to classifiers, it is more difficult to attack object detection: (1) The classifier only needs to classify a single target in the picture so attacking the classifier only needs to change the category label of an object, while the object detection needs to locate and classify multiple target objects with different positions and sizes in the image. In order to realize such a function, the neural network for object

detection is also more complicated, so when attacking the object detector, the position and category information in the label must be considered at the same time. (2) Since the object detector detects all objects in a scene, the context's semantic information will also impact the final detection result, and imaging circumstances will also produce different contextual semantic information, which brings difficulty to the attack.

The attack on the object detector can be divided into the creation attack, the misclassification attack, and the disappearance attack according to attack targets. The creation attack enables the target detector to detect the target that does not exist in the image through the adversarial patch, the misclassification attack will recognize the target as a wrong label, and the disappearance attack is to make the detector unable to detect the attack target. We can hide some targets that we do not want to be discovered by disappearing attack, such as the military, traditional methods often use camouflage nets to hide military assets, but for some large items such as airplanes, cars, etc., it is difficult to use traditional camouflage to conceal, so disappearance attack is always an efficient way for us to find a patch perturbation method so that we can hide our large assets in remote sensing images without being detected. In [21], it is found that an adversarial patch block can be generated and pasted on the target to achieve the purpose of not being detected by the object detector, which proves that the adversarial attack can be applied to the remote sensing system. However, in the real scene, the background of remote sensing images is diverse and complex, and there are some challenging physical conditions such as distance and angle which make attacks more difficult. Although some prior studies [22–24] demonstrate the fragility of object detection in the real world, there are still some limitations. In terms of naturalness, most of the adversarial patches generated are irregular and unreal perturbations. However, the size of adversarial patches occupies most of the area of the attack target, which causes the generated patches to not conform to the actual situation and is suspicious to humans.

To address the mentioned problems, this paper proposes the Rust-Style Patch (RSP) attack to implement the disappearance attack on target objects for object detection in the physical domain. The proposed method simulates the effect of rust based on style transfer so that the adversarial example can preserve the content of the image itself while incorporating the texture and color features of the style image. In addition, we use gradient-based attention techniques (Grad-CAM [25], Grad-CAM++ [26], etc.) to obtain a patch mask to narrow the scope of the attack and generate an irregular shape patch that is closer to the real rust effect, to further enhance the imperceptibility and naturalness of the patch. Such an attention algorithm can help us obtain a saliency map of the detection image. We set a threshold on the saliency map to control the size of the patch and obtain the important attacking region which has the highest impact on the final detection result, then we choose this region as our attacking area so that we can get the patch that can both reduce the attack range and stabilize the attack effect.

Our main contributions in this paper are summarized as follows:

1. We propose a method to generate natural, rust-like adversarial patches generation method based on style transfer, which makes the adversarial patches as natural as possible while making the object detector fail for target objects detection.
2. We utilize attention techniques to obtain the most aggressive regions, effectively reducing the size of the adversarial patch, and balancing the attack effect and naturalness by a preset size threshold.
3. Experiments on the coco dataset in both digital and physical domains show the effectiveness and attack performance of the method. Experiments on the NWPU VHR-10 dataset also prove the effectiveness of our method on remote sensing images.

The rest of the paper is organized as follows: the existing related work about adversarial examples will be introduced in Section 2, and an explanation of our method is provided in Section 3. Section 4 will show the attack effect of the method in this paper in the digital domains and physical domain and there are also some ablation experiments to discuss the results. Conclusions and future study directions can be found in Section 5.

2. Related Work and Background

2.1. Naturalistic Adversarial Patch

Adversarial patch attack was first introduced in [11]. The attack method in [11] generates a universal patch by maximizing the probability expectation of the target category, then this patch can be used to attack any scene and cause a classifier to output any target class because it works under a wide variety of transformations. Subsequently, the RP2 method is proposed in [22] to use the mask to map the disturbance in a graffiti-like shape and paste it onto the traffic sign in the form of black and white blocks, making the classifier output the wrong result. The DARTS method [27] generates adversarial logos such that the classifier recognizes out-of-distribution data as the targeted traffic sign category. Some researchers use a laser [28] or optical projection [29] to project the patch onto the target object to attack in the form of light.

Recently, adversarial patch attacks have also been studied for the more challenging scenario of object detection. DPatch [18] proposed a patch-wise object detection attack that obtains the adversarial patch by simultaneously computing the positional regression loss and the classification loss. It attaches an adversarial patch on the upper-left corner of an image and can make the object detector unable to recognize all the objects on the image. This method simultaneously attacks the YOLO [30] network and the Faster-RCNN [31] network, but the method is only implemented in the digital domain. A square card-style adversarial patch generation method is proposed in adversarial-yolo [32]. When a person puts an adversarial piece of paper in front of the body, it can make the person invisible in front of the YOLOv2 [33] detector (hide a person from YOLOv2). Ref. [34] extended this approach to attack YOLOv3 [35] and Faster-RCNN networks and demonstrated the transferability of attacking different object detectors trained on different datasets. An improvement of adversarial-yolo is proposed in [36], the method input the optimized latent variables into the pre-trained GAN network in each iteration, and finally generated an adversarial pattern similar to the real thing (such as Dog), and experiments on multiple object detectors demonstrate the robustness of the attack. In [21], the author extended the method in adversarial-yolo to military remote sensing object detection and realized the attack of hidden targets by adding perturbation patches on different types of aircraft fuselages. There are two kinds of adversarial patches proposed in [37] to fool the aerial imagery object detector, patches were installed on or near target objects to significantly reduce the efficacy of an object detector applied on overhead images. For static targets, three rectangular strips of patches similar to the stop line are generated and installed around the objects, and for dynamic targets, a patch block is installed on the car roof, so that the target objects in the remote sensing image are not detected.

Regarding traffic signs, ShapeShifter method [24] proposed the first targeted attack on Faster-RCNN by generating adversarial and perturbation posters to replace the background images of traffic signs. The method extended the attack on YOLOv2 based on the RP2 attack method proposed in [22], and realized the disappearance attack and creation attack in the physical domain, they used Faster-RCNN for detection to prove the transferability of adversarial examples. NaturalAE method is proposed in [38], using an adaptive mask to limit the perturbation attack range so that the adversarial examples are relatively natural, and this method also introduced the real-world perturbation score (RPS) to make the noise similar to the real noise of the physical world. However, as shown in Figure 1a,b, the adversarial perturbations generated by these methods almost occupy the entire background of traffic signs and are easy to perceive. In contrast, our method can effectively reduce the attack area of the patch (as shown in Figure 1c).

2.2. Explainable Adversarial Attack

To better understand the behavior and decision-making of neural networks, researchers have conducted a lot of research on visual attention techniques. The linear graph interpretation method is a classic and commonly used technique to explain DNN. For example, the saliency map obtained based on the class activation mapping [39], which

can locate the specific position of the target in the input image under the supervision of image-level labels, but the saliency map generated by this method needs to be modified the structure of the network, so the Grad-CAM algorithm [25] was proposed to identify the continuous spatial region of the input and obtain a visual activation map for a certain category without modifying the original model.

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

Figure 1. (**a**,**b**) show the suspicious appearance of camouflages generated by previous work, (**c**) shows the rust-style adversarial example generated by our method.

With the property of visual attention techniques, these algorithms are also commonly used in adversarial attacks [40]. In PS-GAN [41], the author used Grad-CAM to find the important area of image classification as the central point of the patch. In [42], the Grad-CAM constraint is introduced in the loss function to improve the robustness against patches, but these methods can only be applied to image classification. Grad-CAM is also used in DAS [43] to provide saliency map information. The author uses activation map weights to distract the attention of target detection, but this method needs to use most of the salient regions to achieve the purpose of a black-box attack, so adversarial perturbations also occupy most of the target.

2.3. Background

Among the currently existing object detectors, YOLO is the most commonly used one. YOLO [30] is a one-shot state-of-the-art object detector. Compared to the two-stage target detector (i.e., Faster-RCNN [31]), it will not use the RPN network to generate the region of interest in advance, so its detection speed is almost three times that of the two-stage object detection network. Since YOLO has achieved complete functions in the third version, in this work, we choose the third version of YOLO [35] to attack. The structure of YOLO is shown in Figure 2. YOLOV3 uses the darknet-53 network as the backbone network, it contains three detection heads of different sizes to complete multi-scale prediction. The detection network divides the input image into S × S grids, and each grid will output B predicted anchor boxes bias information, class possibility, and target possibility. Then output the most reasonable prediction results through the non-max-suppression algorithm, and each prediction result includes the position coordinates and confidence of the bounding box.

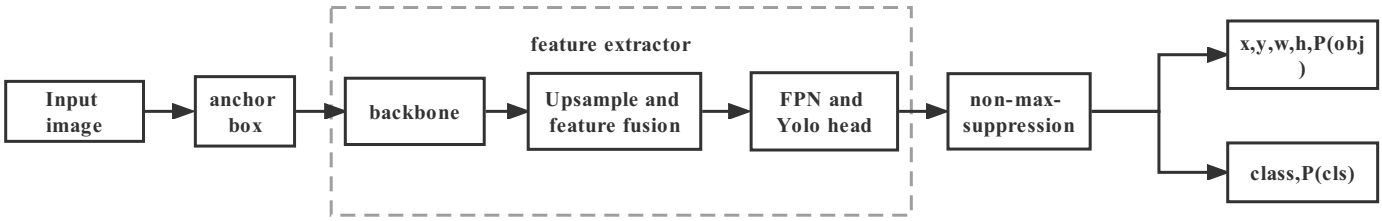


Figure 2. The detection process of YOLO.

3. Methods

In this part, we first define the adversarial attack problem and an overview of the method (Section 3.1), then we introduce the attack region generation method (Section 3.2), the generation of natural adversarial patches (Section 3.3), physical domain adaptation (Section 3.4) and overall optimization processing (Section 3.5).

3.1. Overview

The overview of the RSP attack is illustrated in Figure 3. We give an input image x at the beginning of the attack and generate the mask of its main attack area through the saliency map, then map the patch δ to the attack area. The generated examples will output several targets O_i after passing through the detection network f. Our goal is to find an adversarial patch δ that causes the object detector invalid and unable to recognize the object. We use the following Equation (1) to indicate our optimization goal of finding an adversarial patch.

$$\text{argmin}_{\delta}\mathbb{E}_{t\sim T}[L(f(A(\delta, x, t)))] \tag{1}$$

where t represents the distribution of the patch transformation (which we will further discuss in Section 3.4), A is a function that maps the patch δ after transformation t to the input image x to generate an adversarial example x_{adv}, L is the loss function designed for adversarial example.

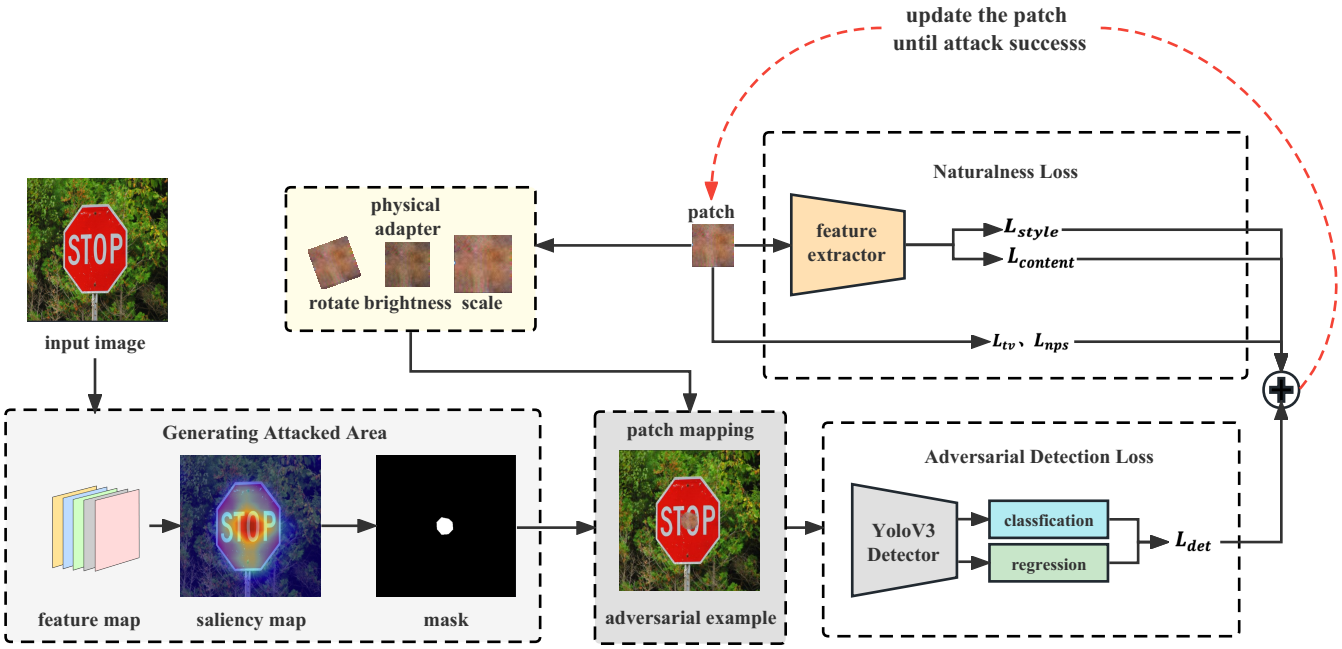


Figure 3. The framework of our method.

3.2. Generating Attacked Area

Previous attacks were aimed at the entire background of traffic signs. Since the attack performance largely depends on the visual sensitivity of the attacked network, we consider using visual attention technology to obtain areas that have a greater impact on the prediction results and use them as attack areas, ensuring the attack robustness of the network while reducing the attack scope. Visual attention techniques are used to improve the interpretation and understanding of model decisions. In this work, we use the Grad-CAM algorithm. The algorithm is aimed at a specific category c, and it will extract the score y^c predicted by the model and perform backpropagation to obtain the gradient information A' of the feature map A of the last layer of the convolutional neural network. We use Equation (2) to get the weight α_k^c of each channel of the feature map; finally, we can

obtain the saliency map G_c by Equation (3). The value of each point on the attention map represents the contribution of the corresponding pixel in the original image to the category detection results.

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \tag{2}$$

$$G_c = \text{RELU}\left(\sum_k \alpha_k^c A^k \right) \tag{3}$$

where c is the target class to attack, k is the k^{th} channel in feature layer A, Z represents the size of feature layer. A_{ij}^k represents the data at the position of (i, j) in the channel k of feature layer A, α_k^c is the weight of the k^{th} channel of feature layer A for category c, G_c is the attention map for category c and $G_c \in [0, 255]$.

Finally, we binary process the heat map by setting a threshold η, when the value of the attention map is greater than the threshold, it means that it has a great impact on the detection results and it will be used as a mask to generate the patch area. The selection of the threshold will affect the size of the disturbance area. Too much disturbance will make the patch unnatural, which is easily observed. However, a patch that is too small is likely to fail a successful attack which may affect the attack performance. Therefore, we need to select an appropriate value η through multiple comparison experiments to get the final mask we need which can balance the patch naturalness and attack performance.

The mask helps project the computed perturbation patch onto the physical area of the object surface in our patch applier function A. In addition to providing spatial locality, masks help produce perturbations that are visible but not obvious to human observers. The mask is a matrix M with the same dimension as the input image to the detector. M has a pixel value of 0 in the area where no perturbation is added, and 1 in the area where the perturbation is added during optimization, which means $M \in \{0, 1\}^n$. The adversarial example obtained after mapping is expressed as follows:

$$x_{adv} = (1 - M) \odot x + M \odot \delta \tag{4}$$

where $\odot$ represents element-wise multiplication.

3.3. Generating Naturalistic Adversarial Patches

3.3.1. Naturalness Loss

To generate imperceptible adversarial patches, we consider using style transfer [44] to make adversarial patches approximate the effect of rust spots. As shown in Figure 4, input a target image G, a style image S, and a content image C, style transfer will minimize the content loss $L_{content}$ between the target image G and the content image C and the style loss L_{style} between the target image G and the style image S through iterations, it makes the texture of the style image can be transferred to the target image G while the target image can maintain the structural information of the content picture. Among them, the content and style information in the image can be separated from the feature representation learned by the convolutional neural network. We use cross-entropy as a loss function.

In this paper, the target image G is the patch δ we want to optimize, and the content image is selected from the part mapped by the adversarial patch clipped from the original image, to ensure that the original image can still retain a certain content outline after being covered by the patch. In a neural network, the deep network extracts high-dimensional semantic content information, so the content loss is compared with the features of deep layers. The content loss definition equation is as follows:

$$L_{\text{content}}(C, \delta) = \frac{1}{2} \sum_{i,j} \left(C_{ij}^l - \delta_{ij}^l \right)^2 \tag{5}$$

where C_{ij}^{l} represents the feature value of the content image C at position j on the i^{th} feature map in the l^{th} network layer.

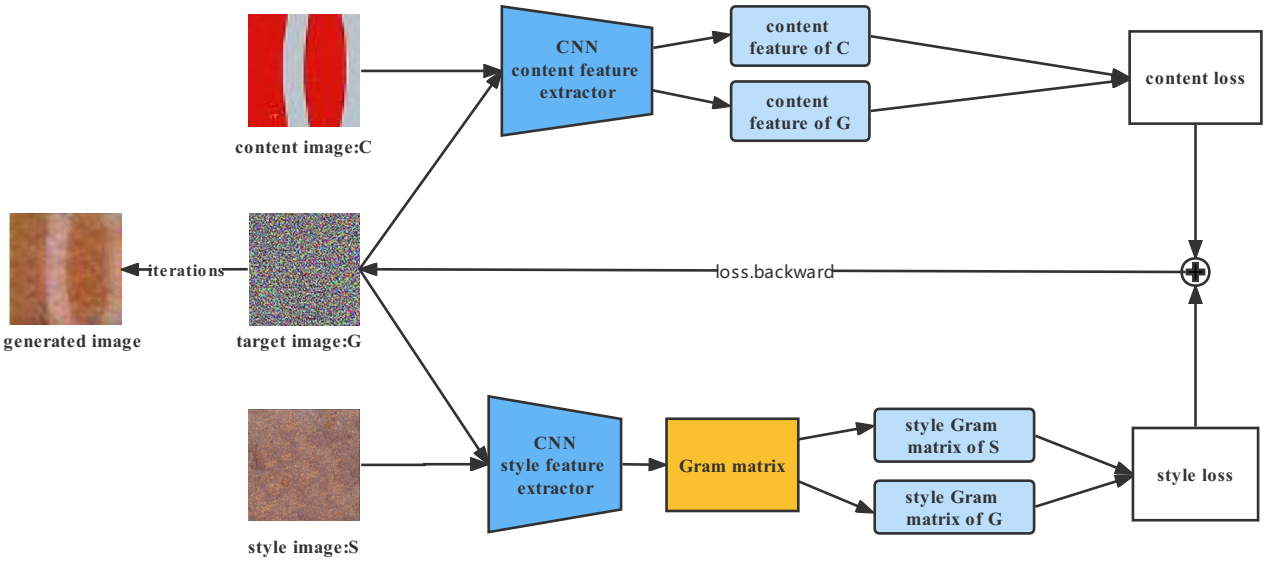


Figure 4. The overview of style transfer.

we choose a picture with rust texture as the style image. The low-level features in the neural network pay more attention to the pixel information of the image input, and extract low-dimensional features such as color, so the style loss is often compared to the shallow layer (usually multiple feature layers) network features. In style transfer, the style features of the image are obtained from the Gram matrix. The Gram matrix contains the correlation of different features of the convolution kernel. The Gram matrix is obtained by the inner product between different convolution kernel features in the feature map F. The equation is as follows:

$$G_{ij}^{l}(x) = \sum_{k} F_{ik}^{l}(x) \times F_{jk}^{l}(x) \tag{6}$$

where $G_{ij}^{l}(x)$ represents the style feature of the image x in the l^{th} network layer, F_{ik}^{l} is the feature value of the k^{th} channel of the i^{th} feature map in the l^{th} layer, i, j are sequences of vectorized features, then we can get the equation of style loss:

$$L_{\text{style}}(S, \delta) = \sum_{l=0}^{L} \left[\frac{G_{ij}^{l}(\delta) - G_{ij}^{l}(S)}{2N_l M_l} \right]^2 \tag{7}$$

where N_l is the number of feature map in the l^{th} network layer, M_l is the size of feature map in the l^{th} layer. The final style loss is the weighted sum of the style losses of each convolutional layer.

3.3.2. Adversarial Detection Loss

As for each object O_i in the input image, YOLO will output their class probability P_{cls}^{i} and objectness probability P_{obj}^{i}. The objectness probability represents the possibility of the i^{th} object. When the objectness probability is reduced below the threshold, the detector cannot recognize the object. The class probability represents the probability that the i^{th} object is of category c. By reducing the class possibility, the target object can be classified as the wrong class, so when we want the detector to fail to recognize the object, we need to drop its confidence below the threshold.

For the disappearance attack, we achieve the attack goal by minimizing both the class probability and objectness probability of the attack target at the same time. Since the perturbation generated during the iteration process may allow the detector to identify additional non-existent targets, we add the possibility of additional objects in the detection loss. Our adversarial detection loss is defined as follows:

$$L_{O_D} = \frac{1}{N}\sum_{i=1}^{N} P_{obj}^{i} \times P_{cls}^{i} \tag{8}$$

$$L_{fab} = \frac{1}{M}\sum_{j=1}^{M} P_{obj}^{j} \times P_{cls}^{j} \tag{9}$$

$$L_{det} = L_{O_D} + L_{fab} \tag{10}$$

where L_{O_D} represents the detection loss of target attack objects, N represents the number of target attack objects, L_{fab} represents the detection loss of additional generated objects in the iterative process, M is the number of generated non-existing objects.

3.4. Adaptation for Physical-World Conditions

Our goal is that the generated adversarial examples can also achieve attack effects in the real world. Since there are many factors in the physical world, such as the combination of viewing angle, camera distance, and other natural transformations, we adopt the EOT (Expectation Over Transformation) framework [17] to simulate the situation in the physical domain. During the patch generation process, we randomly simulate the changes in illumination, position, and angle on the patch and the attack target through affine transformation, and simulate different kinds of backgrounds, making the patch better adapt to the real situation.

Since we need to print the generated adversarial examples through a device (such as a printer), the color space of the pictures produced by these devices is a subset of the RGB color space in the digital domain. So for attacks in the physical domain, we also introduced the non-printability score L_{nps} [45] to reduce the discrepancy between digital field-generated patches and actual printed ones. We can get NPS score from Equation (11).

$$L_{nps} = \sum_{\delta_{\text{patch}} \epsilon \delta} \min_{c_{\text{patch}} \epsilon c} \left| \delta_{patch} - c_{patch} \right| \tag{11}$$

where δ_{patch} represents the pixel value in the patch δ, c_{patch} is the corresponding color value in a set of printable colors c.

3.5. Overall Optimization Process

Finally, we optimize adversarial perturbations by optimizing the losses $L_{content}$, L_{style}, L_{tv}, L_{det} and L_{nps}. The total loss equation is as Equation (12), we use the Adam optimizer [46] to optimize the patch δ such that the total loss is minimized. Our overall process is described in Algorithm 1.

$$L_{total} = L_{nps} + L_{content} + \alpha L_{style} + \beta L_{det} \tag{12}$$

We use the hyper-parameters α and β to control the weight of style effect and detection loss to keep the balance between attack performance and realism in the optimization process, where α and β are determined by experience and experiment.

Algorithm 1: Generate rust-style adversarial patch

Input: a clean image x; balance parameters α, β; mask threshold η; a style image S; iteration parameters $maxEpoch$;
Output: naturalistic adversarial patch δ and adversarial example x_adv

```
epoch ← 0;
obtain attention map G_c of x by Grad-CAM;
obtain mask M and location loc of patch;
clip x with loc as content image C and initial Patch δ;
while epoch < maxEpoch do
    Clip x_adv in 0–255;
    rotate, scale, adjust brightness for δ;
    x_adv = x ⊙ (1 − M) + I ⊙ δ;
    caculate L_det with Equation (10);
    caculate L_content, L_style with Equations (5) and (7);
    caculate L_nps with Equation (11);
    L_total = L_nps + L_content + αL_style + βL_det;
    optimize Patch δ to min L_total;
    epoch+=1 ;
end
```

4. Experimentation and Results Discussion

In this part, we first introduce the experimental settings, then conduct digital and physical experiments on object detectors to prove the effectiveness of our method, and compare it with other methods. Finally, we demonstrate the effectiveness of the functionalities of each part of our method.

4.1. Experiment Settings

Datasets and Target models: We attack the YOLOv3 model which is based on the Darknet-53 network for feature extraction. In order to simplify the process and get our adversarial examples more quickly, we choose the tiny version of YOLOv3 for experiments and evaluations. The model is pre-trained on the coco dataset and NWPU VHR-10 dataset (416 $\times$ 416 pixels). The coco dataset contains 80 general object types such as various animals, people, and vehicles. Although our method can potentially be used to attack other categories, we choose to focus on the 'stop sign' category in the coco dataset to attack in our experiment, since it is more relevant and important to the real scene of the application. The NWPU VHR-10 dataset is a public dataset for research on geospatial object detection which contains 10 categories such as airplane, ship, vehicle and so on. The dataset contains a total of 800 high-resolution remote sensing images cropped from the Google Earth and Vaihingen datasets.

Evaluation metrics: To evaluate the attack effect of our method, we mainly evaluate it from two aspects: attack performance and imperceptibility. For the attack performance, we use the attack success rate (ASR) for evaluation, and for the naturalness, we use the patch size ratio and SSIM (Structure Similarity Index Measure) as our main evaluation metrics. In some cases, we need a comprehensive evaluation metric that considers both attack ability and naturalness, so we define a metric *score* to combine all the evaluation metrics of the two aspects to calculate the final score. The equation of the score is calculated as follows:

$$\text{score} = \frac{\text{SSIM} * \text{ASR}}{\text{patch size}} \tag{13}$$

Attack success rate refers to the ratio of the number of examples that successfully realize our attack effect to the total number of experiments, the patch ratio is the ratio of the area occupied by the adversarial patch to the area of the entire traffic sign. SSIM is used to measure the degree of distortion of adversarial examples relative to clean examples, it consists of three parts: brightness comparison, contrast comparison, and structure compari-

son. Unlike traditional evaluation metrics such as PSNR, SSIM is a perceptual model that is more in line with the intuitive experience of human vision. The value of SSIM is between [0, 1], which means if the value is closer to 1, the example is closer to the original one. We use the two indicators of SSIM and patch size ratio to measure the size, imperceptibility, and perturbation on the clean examples of our generated adversarial patches.

Implementation details: Through multiple rounds of experiments, we obtain various parameters that adapt to our attack. In the experiment, we select 220 for the mask's threshold η of the attack area, and the learning rate of the Adam optimizer is set to 0.01. In the adversarial patch generation process we set the weight α of the style loss to 1000, and the weight β of the detection loss to 10,000. The maximum number of training epochs is 1000. All codes are implemented in PyTorch. Our training and testing are performed on the NVIDIA GeForce GTX2080Ti GPU cluster.

4.2. Digital Attack

We selected a total of 1000 images containing stop signs from the coco dataset for digital domain experiments. We generate the mask of the attack area and the corresponding patch for the attack example. It is considered a success if an adversarial patch that can have an attack effect is obtained within the maximum number of epochs. Our evaluation results are shown in Table 1, it can be seen that our method has a strong attack capability in the digital domain, and the success rate of the disappearance attack on the YOLOV3-tiny network can be as high as 95.7%. The attack is more practical in the physical domain, so the digital domain attack part is only used to demonstrate its attack performance, and we will compare the effect of the method with prior work in Section 4.4.

Table 1. The performance of our method in the digital domain.

Dataset	ASR (%)	SSIM
Disappearance Attack	95.7	0.913
NWPU VHR-10	78.2	0.926

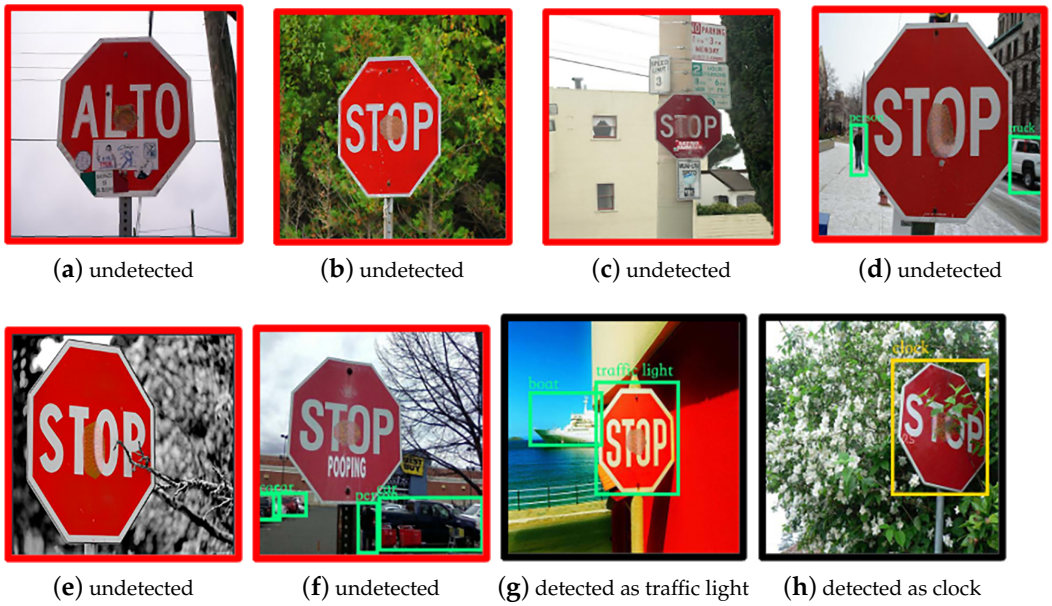
Figure 5 shows the adversarial examples generated by the disappearance attack in the digital domain and the corresponding attack effect. We found that in the disappearance attack, it may be misrecognized as the other categories when the loss iteration is difficult to converge, resulting in the failure of the attack. The converged misclassifications, such as Figure 5g,h, are often fire hydrants, traffic lights, and clocks. We conjecture that the shape feature and the color feature can have a large impact on the detection results and are vulnerable to attack mistakes, which we put into future work to investigate.

We also conduct the experiment on NWPU VHR-10 dataset in the digital domain. As shown in Table 1, our attack method can achieve a success rate of 78.2% on the remote sensing dataset. Figure 6 shows the detection results of clean examples and their corresponding adversarial examples, it can be seen that most of the objects in the adversarial examples after the adversarial attack will not be detected by the object detector, which proves that our disappearance attack method is also effective on remote sensing images. However, due to the low accuracy of the YOLOV3 object detector itself in the detection of few-shot objects in remote sensing images, which causes the generated saliency map is not accurate enough, the final attack success rate is lower than that of the coco dataset. In future work, we will focus on improving its accuracy and success rate.

4.3. Physical Attack

For the attack in the physical domain, we first simulate changes in various physical domains for a stop sign, put the stop sign into different background situations for training, then generate an adversarial patch with the disappearing effects and print out the stop sign with the patch. We use the mobile phone (iPhone 13) to collect 150 photos in both indoor and outdoor environments and put the photos into the detector for detection to evaluate our

attack effect in the physical domain. Figure 7 shows some adversarial examples taken from different angles in indoor and outdoor environments and successfully attacked. According to the experimental results, it can be seen in Table 2 that the success rate in the physical domain is lower than that in the digital domain due to various environmental factors. In an indoor environment, the success rate can reach 70.6%, but only 65.3% in an outdoor environment. Attacks in the physical domain still have certain limitations at present, and it is difficult to get a high success rate at a relatively long distance.



(**a**) undetected (**b**) undetected (**c**) undetected (**d**) undetected

(**e**) undetected (**f**) undetected (**g**) detected as traffic light (**h**) detected as clock

Figure 5. Some examples of adversarial camouflages in the coco dataset generated in the digital domain. Figures with red outlines are examples of successful attacks (i.e., (**a**–**f**) are undetected by object detector), figures with black outlines are examples of unsuccessful attacks but misclassification (i.e., (**g**) is misclassified as a traffic light and (**h**) is detected as clock).

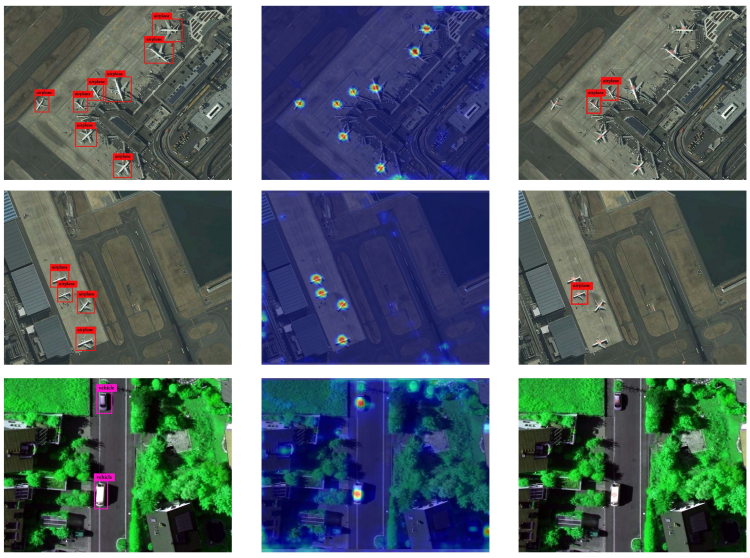
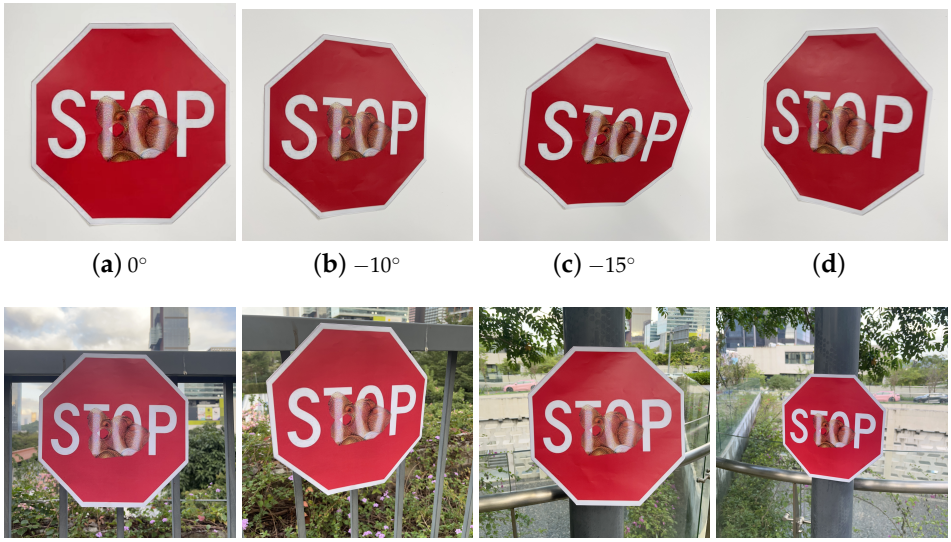


Figure 6. Some adversarial perturbation examples of remote sensing images.The first column in the figure represents the detection results of the clean examples, the second column represents the saliency map used to generate the mask, and the third column represents the detection results of the adversarial examples after the attack.

Table 2. The performance of our method in the physical domain and comparison with other methods.

Method	ASR (%)		Patch Size Ratio (%)	SSIM	Score
	Indoor	Outdoor			
our	70.6	65.3	13.5	0.861	25.0
[47]	72.7	56.7	24.0	0.558	8.7
NaturalAE	62.0	72.0	100.0	0.642	2.5
Clean image	0.0	0.0	0.0	1.000	-



(**a**) 0° (**b**) −10° (**c**) −15° (**d**)

(**e**) 0° (**f**) −10° (**g**) 0° (**h**) 20°

Figure 7. Some examples of successful adversarial attacks in the physical domain. The contents in parentheses indicate the shooting angle. (**a**–**d**) are adversarial examples under indoor physical scenes, (**e**–**h**) are adversarial examples under outdoor physical scenes.

We also compare our method with those proposed in NaturalAE [38,47]. It can be seen in Table 2 that the NaturalAE method adds perturbation to the entire traffic sign object, and the final patch needs to cover the entire target, so its patch size ratio is 100% (as shown in Figure 8c), while the patch area in the method proposed in [47] is not so large, but its patch style is a messy and brightly colored pattern, so it has a low SSIM (as shown in Figure 8d). Compared with the two methods, the real naturalness metric SSIM of the patch we generated is much higher than the other two methods, and the attack area is also smaller (as shown in Figure 8b). From the perspective of attack success rate, our attack performance can effectively improve its naturalness while maintaining a success rate similar to the previous method, and in some cases even slightly higher than these two methods. Above all, our method obtains the highest score combining the three indicators and experiments can prove that our method is effective and can be balanced in naturalness and attack performance.

4.4. Ablation Experimental

4.4.1. The Balance between Attack Performance and Naturalness

It can be seen from Table 3 that there is a contradiction between the naturalness and the success rate of attack. When the proportion of our patch size is reduced (the attack area becomes smaller), the generated adversarial camouflage has higher naturalness, but it will definitely reduce the attack success rate, which means that the attack effect will become worse. So naturalness is an incidental goal, but it should not be the only goal. We keep the balance between naturalness and attack effect by controlling the patch to the right size in the actual application.

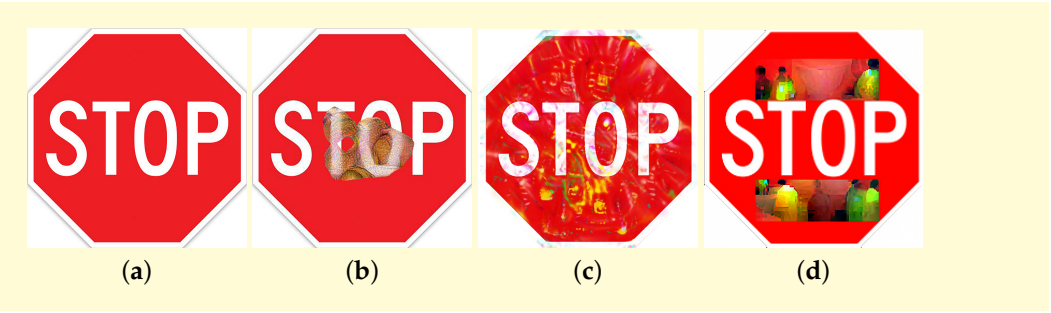


Figure 8. (**a**) is the figure of a clean image without attacked, (**b**) is the figure of our method, (**c**) is the figure of NaturalAE and (**d**) is the figure of [47].

Table 3. The values of corresponding SSIM and attack success rate with the change of patch size ratio.

Patch Size Ratio (%)	6.0	10.2	13.5	20.4	33.6	40.0
SSIM	0.985	0.917	0.913	0.897	0.854	0.812
ASR (%)	7.7	61.5	95.7	96.2	96.6	97.1

4.4.2. The Influence of the Location of Patch and the Threshold η

To prove that the area obtained by using the interpretability vision technique can get the highest success rate and to get the attack size that can balance the attack effect and naturalness, we select different thresholds η to obtain the different sizes of the attack area (the larger η will get the smaller the size) and choose different attack positions for different attack sizes in the experiment. We use *score* to compare their combined attack capabilities here.

As shown in Figure 9, we take the value of η from 190 to 250, then choose eight directions (up, down, left, right, upper left, lower left, upper right, and lower right) relative to the position of the patch in our method for comparison. It can be seen that the score obtained by our method is much higher than that of several other positions, which means that we can guarantee a high attack success rate while obtaining small-sized and natural adversarial patches, meanwhile, it is proved in DAS [43] that the generation of attention maps is similar between different models, which shows that adopting this method as an attack area is general and transferable. From the results of the experiment, we also found that when the η is 220, relatively high scores can be obtained in all positions, but when the η is 240, the attack cannot be realized because the generated patch area is too small, so that the final score is low.

4.4.3. The Performance of Camouflage Losses

Figure 10 illustrates the effectiveness of camouflage losses (including content loss $L_{content}$ and style loss L_{style}) in this method under disappearance attack. We generate the final adversarial example with the detection loss weight of 1×10^5 and a threshold of 220. It can be found that when we generate without style loss, the color of the patch generated in Figure 10b tends to be bright and bright, which is easily noticeable. When only style loss is used without a content loss (Figure 10c), the patch will cover the original outline on the stop sign. If the attack area is larger, it will affect the recognition of the sign by human vision and affect realism.

4.4.4. The Effectiveness of Physical Adaption

We conduct experiments both in the digital domain attack and the physical domain attack after adding the physical domain simulation to prove the effectiveness of the physical adaptation. As shown in Figure 11 above, the left picture shows the adversarial patch simulated in the digital domain, and the right one shows the adversarial example generated after physical adaptation. We printed them out and carried out attack experiments in the real world. The results are shown in Table 4. We found that the adversarial examples

generated directly in the digital domain are not effective in the real world, the attack success rate is only 22.0%. After physical adaptation, the success rate has increased by 48.6%, while maintaining its naturalness without being greatly affected, and the SSIM value has only decreased by 0.031.

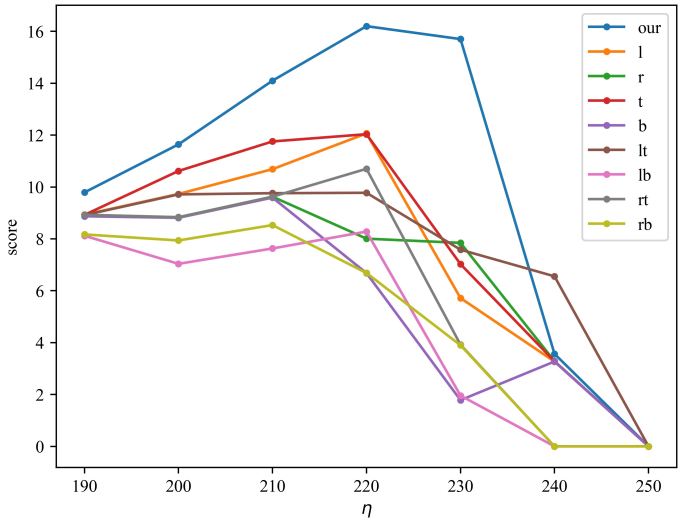


Figure 9. The scores of different locations with different thresholds η. where the y-axis represents the score and the x-axis indicate different thresholds η, which can control the size of the patch. The broken lines of different colors represent the scores when the patch is placed in different positions of the target, and the blue one is the score broken lines obtained by our proposed method under different thresholds.

(**a**) Original (**b**) $L_{content}$ (**c**) L_{style} (**d**) $L_{content}$+L_{style}

Figure 10. Results of the different camouflage losses: (**a**) is the original image, (**b**) is the example generated only with $L_{content}$, (**c**) is the example generated only with L_{style} and (**d**) is camouflaged adversarial examples with both two kinds of loss functions.

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

Figure 11. (**a**) is the adversarial example generated without physical adaption and (**b**) is the one with physical adaption.

Table 4. Performance with and without patches for physical adaption.

Physiccal Adaption	Success Rate (%)	SSIM
No	22.0	0.889
Yes	70.6	0.858

5. Conclusions

Deep neural networks are vulnerable and easily attacked by adversarial perturbations. We need to find effective adversarial examples to improve the robustness and accuracy of the remote sensing detection system. In this paper, we use visual attention techniques to effectively reduce the attack area, and generate a rust-like adversarial patch through style transfer, which can make the object detector unable to detect the target, compared with the previous methods that directly iterate the entire target for adversarial attacks, we solve the problems that most of the adversarial patches are unnatural, and the patch occupies a large position on the target. We use different metrics to evaluate the attack performance and naturalness of our method, and we demonstrate the effectiveness of each function in the attack process through ablation experiments.

In future work, we hope to improve the accuracy of small target detection, thereby improving the detection ability of remote sensing images and the success rate of adversarial attacks. Then we plan to introduce a method for attacking three-dimensional objects on this basis so that the attack could be more general. On the other hand, although attacks in the physical domain have been achieved in our work, it is still difficult for us to obtain a high attack success rate due to the diversity of external environment changes. Therefore, how to generate the perturbation closer to the real world is still a key point in subsequent research. At the same time, we found that the process is difficult to converge after a certain number of iterations if we use style transfer to generate a natural patch in the experiment, so we will find a more suitable and short-term method to get natural adversarial patches.

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

Funding: This work is supported in part by the Major Key Project of PCL (PCL2022A03), Guangdong Key R&D Program of China (2019B010136003), Guangdong Higher Education Innovation Group (2020KCXTD007), Guangzhou Higher Education Innovation Group (202032854), National Natural Science Foundation of China (62250410365), and Guangzhou Science and Technology Program of China (202201010606).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Qian, R.; Lai, X.; Li, X. 3D Object Detection for Autonomous Driving: A Survey. *Pattern Recognit.* **2022**, *130*, 108796. [CrossRef]
2. Fang, W.; Shen, L.; Chen, Y. Survey on Image Object Detection Algorithms Based on Deep Learning. In *Artificial Intelligence and Security*; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2021; pp. 468–480.
3. Minaee, S.; Boykov, Y.; Porikli, F.; Plaza, A.; Kehtarnavaz, N.; Terzopoulos, D. Image Segmentation Using Deep Learning: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 3523–3542. [CrossRef] [PubMed]
4. Yasir, M.; Jianhua, W.; Mingming, X.; Hui, S.; Zhe, Z.; Shanwei, L.; Colak, A.T.I.; Hossain, M.S. Ship Detection Based on Deep Learning Using SAR Imagery: A Systematic Literature Review. *Soft Comput.* **2022**, *27*, 63–84. [CrossRef]
5. Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; Fergus, R. Intriguing Properties of Neural Networks. In Proceedings of the International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
6. Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; Vladu, A. Towards Deep Learning Models Resistant to Adversarial Attacks. In Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.

7. Gu, Z.; Li, H.; Khan, S.; Deng, L.; Du, X.; Guizani, M.; Tian, Z. IEPSBP: A Cost-Efficient Image Encryption Algorithm Based on Parallel Chaotic System for Green IoT. *IEEE Trans. Green Commun. Netw.* **2021**, *6*, 89–106. [CrossRef]
8. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. *arXiv* **2015**, arXiv:1412.6572.
9. Van Etten, A. The Weaknesses of Adversarial Camouflage in Overhead Imagery. *arXiv* **2022**, arXiv:2207.02963.
10. Kurakin, A.; Goodfellow, I.; Bengio, S. Adversarial Examples in the Physical World. *arXiv* **2017**, arXiv:1607.02533.
11. Brown, T.B.; Mané, D.; Roy, A.; Abadi, M.; Gilmer, J. Adversarial Patch. *arXiv* **2018**, arXiv:1712.09665.
12. Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; Li, J. Boosting Adversarial Attacks with Momentum. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 9185–9193.
13. Carlini, N.; Wagner, D. Towards Evaluating the Robustness of Neural Networks. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 39–57.
14. Zhu, B.; Gu, Z.; Qian, Y.; Lau, F.; Tian, Z. Leveraging transferability and improved beam search in textual adversarial attacks. *Neurocomputing* **2022**, *500*, 135–142. [CrossRef]
15. Moosavi-Dezfooli, S.M.; Fawzi, A.; Frossard, P. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2574–2582.
16. Moosavi-Dezfooli, S.M.; Fawzi, A.; Fawzi, O.; Frossard, P. Universal Adversarial Perturbations. *arXiv* **2017**, arXiv:1610.08401.
17. Athalye, A.; Engstrom, L.; Ilyas, A.; Kwok, K. Synthesizing Robust Adversarial Examples. In Proceedings of the 35th International Conference on Machine Learning (PMLR), Stockholm, Sweden, 10–15 July 2018; pp. 284–293.
18. Xin Liu.; Yang, H.; Liu, Z.; Song, L.; Li, H.; Chen, Y. DPatch: An Adversarial Patch Attack on Object Detectors. *arXiv* **2019**, arXiv:1806.02299.
19. Chow, K.H.; Liu, L.; Loper, M.; Bae, J.; Gursoy, M.E.; Truex, S.; Wei, W.; Wu, Y. Adversarial Objectness Gradient Attacks in Real-time Object Detection Systems. In Proceedings of the 2020 Second IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA), Atlanta, GA, USA, 28–31 October 2020; pp. 263–272.
20. Le, T.T.H.; Kang, H.; Kim, H. Robust Adversarial Attack Against Explainable Deep Classification Models Based on Adversarial Images with Different Patch Sizes and Perturbation Ratios. *IEEE Access* **2021**, *9*, 133049–133061. [CrossRef]
21. Adhikari, A.; Hollander, R. D.; Tolios, I.; Bekkum, M. V. ; Raaijmakers, S. Adversarial Patch Camouflage against Aerial Detection. *arXiv* **2020**, arXiv:2008.13671.
22. Eykholt, K.; Evtimov, I.; Fernandes, E.; Li, B.; Rahmati, A.; Xiao, C.; Prakash, A.; Kohno, T.; Song, D. Robust Physical-World Attacks on Deep Learning Visual Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 1625–1634.
23. Zhang, H.; Zhou, W.; Li, H. Contextual Adversarial Attacks For Object Detection. In Proceedings of the 2020 IEEE International Conference on Multimedia and Expo (ICME), London, UK, 6–10 July 2020; pp. 1–6.
24. Chen, S.T.; Cornelius, C.; Martin, J.; Chau, D.H. ShapeShifter: Robust Physical Adversarial Attack on Faster R-CNN Object Detector. In *Machine Learning and Knowledge Discovery in Databases*; Springer International Publishing: Cham, Switzerland, 2019; Volume 11051, pp. 52–68.
25. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.
26. Chattopadhay, A.; Sarkar, A.; Howlader, P.; Balasubramanian, V.N. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 839–847.
27. Sitawarin, C.; Bhagoji, A.N.; Mosenia, A.; Chiang, M.; Mittal, P. DARTS: Deceiving Autonomous Cars with Toxic Signs. *arXiv* **2018**, arXiv:1802.06430.
28. Duan, R.; Mao, X.; Qin, A.K.; Chen, Y.; Ye, S.; He, Y.; Yang, Y. Adversarial Laser Beam: Effective Physical-World Attack to DNNs in a Blink. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 16062–16071.
29. Gnanasambandam, A.; Sherman, A.M.; Chan, S.H. Optical Adversarial Attack. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, Montreal, BC, Canada, 11–17 October 2021; pp. 92–101.
30. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
31. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2015; Volume 28.
32. Thys, S.; Van Ranst, W.; Goedeme, T. Fooling Automated Surveillance Cameras: Adversarial Patches to Attack Person Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Long Beach, CA, USA, 16–20 June 2019.
33. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
34. Wang, Y.; Lv, H.; Kuang, X.; Zhao, G.; Tan, Y.A.; Zhang, Q.; Hu, J. Towards a Physical-World Adversarial Patch for Blinding Object Detection Models. *Inf. Sci.* **2021**, *556*, 459–471. [CrossRef]

35. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1802.06430.
36. Hu, Y.C.T.; Kung, B.H.; Tan, D.S.; Chen, J.C.; Hua, K.L.; Cheng, W.H. Naturalistic Physical Adversarial Patch for Object Detectors. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 10–17 October 2021; pp. 7848–7857.
37. Du, A.; Chen, B.; Chin, T.J.; Law, Y.W.; Sasdelli, M.; Rajasegaran, R.; Campbell, D. Physical Adversarial Attacks on an Aerial Imagery Object Detector. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 4–8 January 2022; pp. 3798–3808.
38. Xue, M.; Yuan, C.; He, C.; Wang, J.; Liu, W. NaturalAE: Natural and Robust Physical Adversarial Examples for Object Detectors. *J. Inf. Secur. Appl.* **2021**, *57*, 102694. [CrossRef]
39. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning Deep Features for Discriminative Localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
40. Gu, Z.; Hu, W.; Zhang, C.; Lu, H.; Yin, L.; Wang, L. Gradient Shielding: Towards Understanding Vulnerability of Deep Neural Networks. *IEEE Trans. Netw. Sci. Eng.* **2021**, *8*, 921–932. [CrossRef]
41. Liu, A.; Liu, X.; Fan, J.; Ma, Y.; Zhang, A.; Xie, H.; Tao, D. Perceptual-Sensitive GAN for Generating Adversarial Patches. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 1028–1035. [CrossRef]
42. Subramanya, A.; Pillai, V.; Pirsiavash, H. Fooling Network Interpretation in Image Classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, South Korea, 27 October–2 November 2019; pp. 2020–2029.
43. Wang, J.; Liu, A.; Yin, Z.; Liu, S.; Tang, S.; Liu, X. Dual Attention Suppression Attack: Generate Adversarial Camouflage in Physical World. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 8565–8574.
44. Gatys, L.A.; Ecker, A.S.; Bethge, M. A Neural Algorithm of Artistic Style. *arXiv* **2015**, arXiv:1508.06576.
45. Sharif, M.; Bhagavatula, S.; Bauer, L.; Reiter, M.K. Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition. In *CCS '16: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1528–1540.
46. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
47. Song, D.; Eykholt, K.; Evtimov, I.; Fernandes, E.; Li, B.; Rahmati, A.; Tramer, F.; Prakash, A.; Kohno, T. Physical Adversarial Examples for Object Detectors. In Proceedings of the 12th USENIX Workshop on Offensive Technologies, Baltimore, MD, USA, 13–14 August 2018.