Defense against Adversarial Cloud Attack on Remote Sensing Salient Object Detection

Huiming Sun[1], Lan Fu[2], Jinlong Li[1], Qing Guo[3],
Zibo Meng[2], Tianyun Zhang[1], Yuewei Lin[4]*, Hongkai Yu[1]*

[1]Cleveland State University [2]OPPO US Research Center
[3]IHPC and CFAR, Agency for Science, Technology and Research (A*STAR)
[4]Brookhaven National Laboratory

Abstract

Detecting the salient objects in a remote sensing image has wide applications. Many existing deep learning methods have been proposed for Salient Object Detection (SOD) in remote sensing images with remarkable results. However, the recent adversarial attack examples, generated by changing a few pixel values on the original image, could result in a collapse for the well-trained deep learning model. Different with existing methods adding perturbation to original images, we propose to jointly tune adversarial exposure and additive perturbation for attack and constrain image close to cloudy image as Adversarial Cloud. Cloud is natural and common in remote sensing images, however, camouflaging cloud based adversarial attack and defense for remote sensing images are not well studied before. Furthermore, we design DefenseNet as a learnable pre-processing to the adversarial cloudy images to preserve the performance of the deep learning based remote sensing SOD model, without tuning the already deployed deep SOD model. By considering both regular and generalized adversarial examples, the proposed DefenseNet can defend the proposed Adversarial Cloud in white-box setting and other attack methods in black-box setting. Experimental results on a synthesized benchmark from the public remote sensing dataset (EORSSD) show the promising defense against adversarial cloud attacks.

1. Introduction

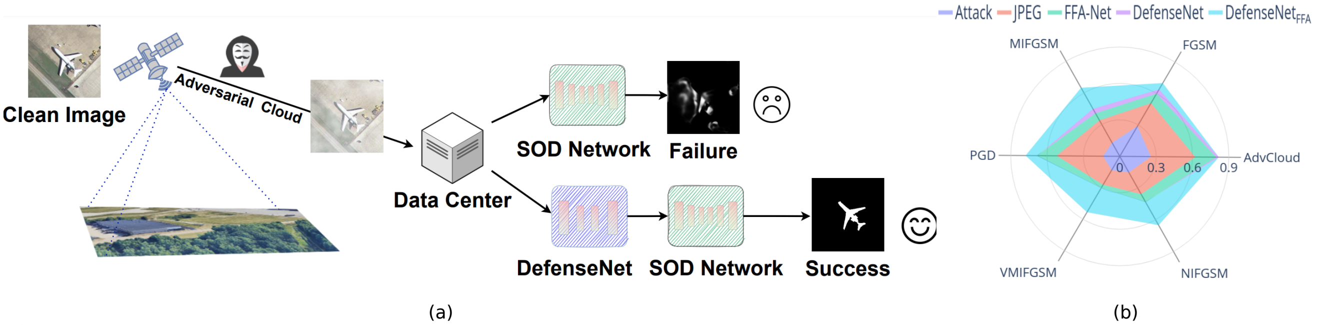
The computer vision and artificial intelligence have many applications in the remote sensing domain, such as hyperspectral image classification [1, 2], cross-view geolocation [3–5], scene classification [6, 7], aerial-view object detection [8, 9], change detection [10, 11], and so on. Remote sensing Salient Object Detection (SOD) aims to extract the salient objects in an aerial-view image, leading to potential benefits to the above applications.

For remote sensing images, some SOD methods have been proposed [12, 13] by using deep learning based architecture, whose efforts are mainly concentrated on typical context learning [13] and multi-scale feature aggregation [12]. However, in some scenarios, these deep learning based remote sensing SOD models might suffer from the attacks by the adversarial examples on deep neural networks. Recent research [14] shows that the adversarial noises can be added to fool the deep learning based SOD models, leading to the low SOD performance. For example, by adding a small portion of adversarial noises on the original remote sensing image between the image acquisition and data processing, *e.g.*, during the communication, the salient objects in the remote sensing image might be hided to some extent by the deep SOD model. This kind of malicious attack exposes a potential security threat to the remote sensing.

Many researches have been proposed for the adversarial examples based attack and defense in deep learning [15–18]. Meanwhile, some attack and defense researches have been proposed for remote sensing tasks, such as the remote sensing scene classification [19]. Different with existing methods adding the perturbation on the original image, we propose to generate Adversarial Cloud as attack to the deep learning based remote sensing SOD model. Cloud is widely common in remote sensing images [20]. However, cloud based adversarial attack and defense for remote sensing images has not been well studied. The proposed Adversarial Cloud has realistic appearance close to a normal cloud, which might be difficult to be perceived but will be malicious in the remote sensing applications.

In this paper, we propose a novel DenfenseNet to defend the proposed Adversarial Cloud attack to preserve the advanced SOD performance. In general, the adversarial attack

*Co-corresponding authors: Yuewei Lin (ywlin@bnl.gov), Hongkai Yu (h.yu19@csuohio.edu).



(a)

(b)

Figure 1. (a) Illustration of the proposed defense against the adversarial cloud attacks for remote sensing salient object detection. (b) Performance (F Measure) of the proposed DefenseNet against different adversarial cloud attacks. Bigger area means better defense.

and defense networks will be trained with an adversarial deep learning by iteratively training the Adversarial Cloud and DenfenseNet. However, the already deployed deep remote sensing SOD model is kept unchanged to simplify the real-world setting. Thus, the proposed DefenseNet is designed as a learnable pre-processing technique to preserve the SOD performance. In specific, the adversarial examples will go through the DefenseNet to become clean examples as the input to SOD models. Based on the publicized remote sensing SOD dataset (EORSSD [13]), we build a benchmark by synthesizing the Adversarial Cloud to test the performance of attack and defense for the SOD problem in the remote sensing images. As shown in Fig. 1 (b), our proposed method could defend different adversarial attack methods. Experimental results on the built benchmark show the effectiveness and accuracy of the proposed method. The contributions of this paper are summarized as follows.

- This paper proposes a novel attack method by jointly tuning adversarial exposure and additive perturbation and constraining image close to cloudy image as Adversarial Cloud for the SOD in remote sensing images.
- This paper proposes a novel DefenseNet as learnable pre-processing against the adversarial cloud attack for safety-ensured SOD in remote sensing images, without tuning the already deployed deep learning-based SOD model.
- By considering both regular and generalized adversarial examples, the proposed DefenseNet can defend the proposed Adversarial Cloud in white-box setting and other attack methods in black-box setting.

2. Related Work

2.1. Salient Object Detection for Remote Sensing

Salient object detection (SOD) is to automatically extract the salient objects in an image. Many existing methods are proposed for SOD in natural images, while the SOD in remote sensing images is more challenging due to the unique, complex and diverse environments [12]. SOD in satellite or drone images has wide applications, such as building extraction [21], Region-of-Interest extraction [22], airport detection [23], oil tank detection [24], ship detection [25].

2.2. Adversarial Attack

There are two types of adversarial attacks: *white-box* attacks, where the adversary has full access to the target model, including its parameters, *i.e*., the model is transparent to the adversary, and *black-box* attacks, where the adversary has little knowledge of the target model. As the white-box attacks are usually more destructive than black-box ones in practice, the literature more focuses on the white-box attacks. Among these white-box attacks, Szegedy *et al*. [26] used a box-constrained L-BFGS method to generate effective adversarial attacks for the first time. After that, the fast gradient sign method (FGSM) [15] used the sign of the gradient to generate attacks, with ℓ_∞-norm bound. Several variants of FGSM were proposed, *e.g*., Kurakin *et al*. [27] applied FGSM iteratively and designed basic iterative method (BIM), and Dong *et al*. [28] integrated the momentum into the model. As a multi-step attack method, the projected gradient descent (PGD) was proposed in [29]. Carlini and Wagner [30] proposed the so-called CW attack which is a margin-based attack. More recently, Croce *et al*. introduced a parameter-free attack named AutoAttack [31], which is an ensemble of four diverse attacks, including two proposed variants of PGD attacks and two existing complementary attacks, *i.e*., FAB [32]. DeepFool [33] were proposed to fool deep classification networks.

2.3. Adversarial Defense

With the development of adversarial examples, studies on how to defend against those attacks and improve the robustness of the neural networks emerge. Among them, the most effective and widely used defense model is adversarial training (AT), although the most straightforward way is simply by attaching a detection network to detect and reject adversarial examples [34]. AT based models, which aim to minimize the loss function to the strongest adversarial attacks within a constraint, were first proposed by [15]. After that, a number of defending methods [18, 29, 35–40] based on adversarial training were proposed. For example, [35] and [36] built a triplet loss to enforce a clean image and its corresponding adversarial example has a short distance in feature space.

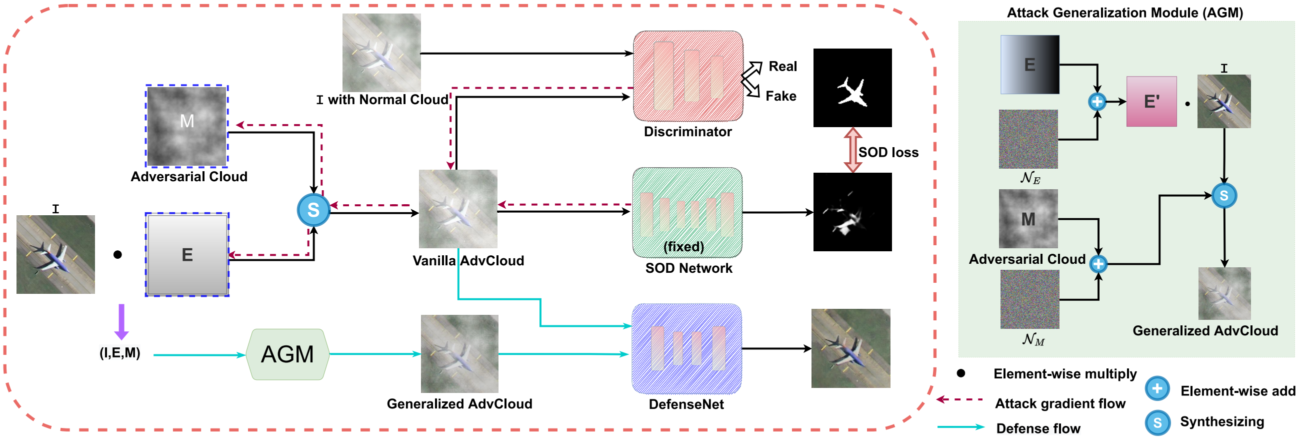


Figure 2. Structure of the proposed Adversarial Cloud (AdvCloud) based attack and the proposed DefenseNet as the defense against the AdvCloud for the remote sensing Salient Object Detection (SOD). Left side: Given a clean image $\mathbf{I}$ multiplied by exposure matrix $\mathbf{E}$ and plus cloud mask $\mathbf{M}$, the synthesized cloudy image $\hat{\mathbf{I}}$ could be obtained, and the DefenseNet is a learnable pre-processing for the SOD network. Right side: $\mathcal{N}_{\mathrm{E}}, \mathcal{N}_{\mathrm{M}}$ are Gaussian noises for the Attack Generalization Module (AGM).

3. METHODOLOGY

3.1. Cloud Synthesizing for Remote Sensing

Given a clean remote sensing RGB image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$, we aim to simulate a cloudy image via $\hat{\mathbf{I}} = \text{Cloud}(\mathbf{I}, \mathbf{E}, \mathbf{M})$, where $\mathbf{E} \in \mathbb{R}^{H \times W \times 1}$ is an exposure matrix to define exposure degree, $\mathbf{M} \in \mathbb{R}^{H \times W \times 1}$ is a cloud mask to simulate clouds, and $\text{Cloud}(\cdot)$ represents the cloudy image synthesis function. Inspired by [20], the cloud mask $\mathbf{M}$ can be synthesized via a summation of multi-scale random noises, and is defined as

$$\mathbf{M} = \sum_{s} \mathbf{R}\left(\mathbf{f}(2^s)\right)/2^s, \tag{1}$$

where $\mathbf{f}$ represents a randomizing function, $\mathbf{R}$ denotes a resize process and s is a scale factor. $\mathbf{f}$ produces random noises with the image size 2^s followed by being resized by $\mathbf{R}$. s is a natural number with range $\in [1, \log_2 N]$, where $N = H \times W$ is the image size. Given a clean image $\mathbf{I}$, exposure matrix $\mathbf{E}$, and cloud mask $\mathbf{M}$, we could synthesize a cloudy image $\hat{\mathbf{I}}$ via

$$\hat{\mathbf{I}} = \text{Cloud}(\mathbf{I}, \mathbf{E}, \mathbf{M}) = \mathbf{I} \odot \mathbf{E} \odot (1 - \mathbf{M}) + \mathbf{M}, \tag{2}$$

where $\odot$ denotes pixel-wise multiplication.

With this cloudy image synthesis, we could study the effects of cloud from the viewpoint of adversarial attack by tuning the exposure matrix $\mathbf{E}$ and cloud mask $\mathbf{M}$ to render the synthesized cloudy images to fool the deep learning based SOD models. Later, we also employ these adversarial examples, obtained by the proposed attack method, to study the defense performance.

3.2. Network Architecture

In this section, we show the whole pipeline of adversarial cloud attack (AdvCloud) and DefenseNet as attack and defense stages to fully explore the cloud effects to a deployed deep SOD model in Fig. 2. In the attack stage, given a clean image $\mathbf{I}$, an exposure matrix $\mathbf{E}$, a cloud mask $\mathbf{M}$, a pre-trained deep remote sensing SOD model $\phi(\cdot)$, and a well-trained discriminator $\mathcal{D}$, we aim to generate adversarial cloudy image examples via the proposed AdvCloud. Then, we analyze how the synthetic adversarial cloudy images hurt the SOD performance. As the other main step of the pipeline, we perform defense process, *i.e.*, DefenseNet, as a pre-processing stage for the adversarial images to generate cloud-removed images as defense for the SOD model. The proposed DefenseNet can avoid retraining the deep SOD model and make the salient object detector process adaptive to cloudy images.

3.3. Adversarial Cloud based Attack

In general, adversarial attack fails a deep model by adding an imperceptible noise-like perturbation to an image under the guidance of the deep model. In this work, we propose a novel adversarial attack method, *i.e.*, AdvCloud, to generate adversarial cloudy remote sensing images that can fool the SOD model to verify the robustness of the SOD model.

By intuition, we can tune $\mathbf{E}$ and $\mathbf{M}$ to generate adversarial cloudy images. Specifically, given $\mathbf{I}$, $\mathbf{E}$, $\mathbf{M}$, and a pre-trained SOD detector $\phi(\cdot)$, we aim to tune the $\mathbf{E}$ and $\mathbf{M}$ under a norm constraint by

$$\begin{aligned} &\arg\max_{\mathbf{E},\mathbf{M}} \mathcal{J}(\phi(\text{Cloud}(\mathbf{I}, \mathbf{E}, \mathbf{M})), y), \\ &\text{subject to } \;\|\mathbf{M} - \mathbf{M}_0\|_{\mathrm{p}} \le \epsilon_{\mathrm{M}}, \|\mathbf{E} - \mathbf{E}_0\|_{\mathrm{p}} \le \epsilon_{\mathrm{E}}, \end{aligned} \tag{3}$$

where $\mathcal{J}(\cdot)$ is the loss function of the SOD model $\phi(\cdot)$ under the supervision of the annotation label y. We set ϵ_{E} and ϵ_{M} as the bound under L_{p} around their initialization (*i.e.*, $\mathbf{E}_0$ and $\mathbf{M}_0$) for the parameters $\mathbf{E}$ and $\mathbf{M}$ to avoid the clean image $\mathbf{I}$ being changed significantly and we set $\mathrm{p} = \infty$.

Similar to existing perturbation based adversarial attacks (*e.g.*, [29]), the object function, *i.e.*, Eq. (3), can be optimized by gradient descent-based methods. In specific: ❶ We initialize $\mathbf{E}_0$ as a mask with all elements as 1 and set $\mathbf{M}_0$ via

Eq. (1). Then, we get the initial synthesized cloudy image by Eq. (2). ❷ We feed the synthesized image to the SOD model $\phi(\cdot)$ and calculate the SOD loss ℓ. ❸ We perform back-propagation to obtain the gradient of $\mathbf{E}$ and $\mathbf{M}$ with respective to the loss function. ❹ We calculate the sign of the gradient to update the variables $\mathbf{E}$ and $\mathbf{M}$ by multiplying the sign of their gradients with the corresponding step sizes for the next iteration, which is formulated to

$$\begin{aligned} \ell &= \mathcal{J}(\phi(\mathrm{Cloud}(\mathbf{I}, \mathbf{E}_i, \mathbf{M}_i)), y), \\ \mathbf{M}_{i+1} &= \mathbf{M}_i + \alpha_{\mathrm{M}} \cdot \mathbf{sign}(\nabla_{\mathbf{M}_i}(\ell)), \\ \mathbf{E}_{i+1} &= \mathbf{E}_i + \alpha_{\mathrm{E}} \cdot \mathbf{sign}(\nabla_{\mathbf{E}_i}(\ell))), \end{aligned} \tag{4}$$

where α_{M} and α_{E} represents the step sizes, and $i \in \{0, 1, \ldots, K-1\}$ is the iteration number. ❺ We generate a new adversarial cloudy image and loop from ❷ to ❹ for K iterations.

Let us redefine $\hat{\mathbf{I}}$ as the generated adversarial cloudy image, to make it with close visualization to the normal cloudy image, we also incorporate a discriminator $\mathcal{D}$ to align the distribution of normal cloudy images and adversarial cloudy images to avoid artifacts which might be introduced by Eq. (3). The inputs of the discriminator are an adversarial cloudy image $\hat{\mathbf{I}}$ and a normal cloudy image $\mathbf{I}_c$, obtained by $\mathbf{I}_c = \mathrm{Cloud}(\mathbf{I}, \mathbf{M}) = \mathbf{I} \odot (1 - \mathbf{M}) + \mathbf{M}$, then the adversarial training loss of the discriminator $\mathcal{D}$ is

$$\begin{aligned} \mathcal{L}_{\mathcal{D}}(\hat{\mathbf{I}}, \mathbf{I}_c) = \ & \mathbb{E}_{\mathbf{I}_c \sim \mathbf{X}_c}[\log(\mathcal{D}(\mathbf{I}_c))] \\ & + \mathbb{E}_{\hat{\mathbf{I}} \sim \hat{\mathbf{X}}}[\log(1 - \mathcal{D}(\hat{\mathbf{I}}))], \end{aligned} \tag{5}$$

where $\mathbf{I}_c$ and $\hat{\mathbf{I}}$ are instances from normal cloudy images set $\mathbf{X}_c$ and adversarial cloudy images set $\hat{\mathbf{X}}$, respectively. The whole attack pipeline, incorporating AdvCloud and discriminator $\mathcal{D}$, is trained on the training set of the remote sensing SOD dataset EORSSD [13]. The above setting has an assumption for a reliable discriminator $\mathcal{D}$ ahead for the following inference stage. Specifically, we alternatively freeze adversarial parameters $\mathbf{E}$, $\mathbf{M}$ and the discriminator $\mathcal{D}$ to optimize the other one to get a reliable discriminator $\mathcal{D}$ in the training set of EORSSD before the following inference.

For the inference stage of the proposed AdvCloud attack, we attack the testing set of EORSSD guided by the pre-trained discriminator $\mathcal{D}$ and the SOD detector $\phi(\cdot)$. Given a clean image $\mathbf{I}$ from the testing set of EORSSD, exposure matrix $\mathbf{E}$ and cloud mask $\mathbf{M}$, a well-trained discriminator $\mathcal{D}$, and a SOD detector $\phi(\cdot)$, we tune $\mathbf{E}$ and $\mathbf{M}$ for K iterations based on back-propagation, while the optimization function Eq. (3) is reformulated to

$$\begin{aligned} & \arg\max_{\mathbf{E}, \mathbf{M}} (\mathcal{J}(\phi(\mathrm{Cloud}(\mathbf{I}, \mathbf{E}, \mathbf{M})), y) - \mathcal{L}_{\mathcal{D}}(\hat{\mathbf{I}}, \mathbf{I}_c)), \\ & \text{subject to } \ \|\mathbf{M} - \mathbf{M}_0\|_{\mathrm{p}} \leq \epsilon_{\mathrm{M}}, \|\mathbf{E} - \mathbf{E}_0\|_{\mathrm{p}} \leq \epsilon_{\mathrm{E}}, \end{aligned} \tag{6}$$

which means the adversarial cloudy image $\hat{\mathbf{I}}$ could fail the SOD detector and have the realistic cloud appearance and pattern close to normal cloudy images. Then, the updating process of variables $\mathbf{E}$ and $\mathbf{M}$, in Eq. (4), is reformulated to

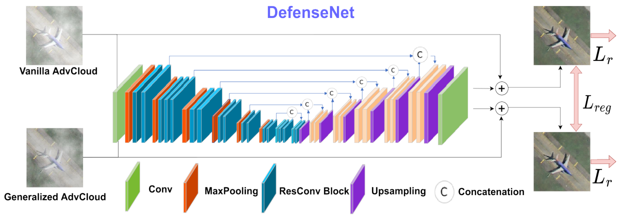


Figure 3. Structure of the proposed DefenseNet.

$$\begin{aligned} \ell &= \mathcal{J}(\phi(\mathrm{Cloud}(\mathbf{I}, \mathbf{E}_i, \mathbf{M}_i)), y), \\ \mathbf{M}_{i+1} &= \mathbf{M}_i + \alpha_{\mathrm{M}} \cdot \mathbf{sign}(\nabla_{\mathbf{M}_i}(\ell - \mathcal{L}_{\mathcal{D}}(\hat{\mathbf{I}}, \mathbf{I}_c))), \\ \mathbf{E}_{i+1} &= \mathbf{E}_i + \alpha_{\mathrm{E}} \cdot \mathbf{sign}(\nabla_{\mathbf{E}_i}(\ell - \mathcal{L}_{\mathcal{D}}(\hat{\mathbf{I}}, \mathbf{I}_c)))). \end{aligned} \tag{7}$$

After obtaining the updated $\mathbf{E}$ and $\mathbf{M}$ for each image from the testing set of EORSSD, we can get the corresponding adversarial cloudy images via Eq. (2).

Novelty of AdvCloud: The conventional attack methods first synthesize the normal cloudy image (original image + normal cloud) and then add perturbations to the synthesized image. Differently, our AdvCloud employs a novel approach to jointly optimize the adversarial cloud mask and exposure matrix during the Cloud Synthesizing for Remote Sensing. Moreover, our AdvCloud integrates constraints that ensure the modified images retain characteristics akin to normal cloudy images.

3.4. Defense against Adversarial Cloud

The proposed AdvCloud attack can easily hurt the SOD performance, while performing defense against adversarial attack is an effective way to alleviate such performance drop. In this section, we propose a DefenseNet as a learnable pre-processing for adversarial cloudy images to acquire cloud-removed images for SOD models to improve the robustness. The proposed DefenseNet contains the two following branches as the inputs.

Vanilla AdvCloud Branch. Given the updated adversarial attack variables $\mathbf{E}$ and $\mathbf{M}$, we can obtain an adversarial cloudy image $\hat{\mathbf{I}}$. Then, it is the first-branch input to the DefenseNet to perform the reconstruction for adversarial cloud removal. This is a simple vanilla defense setting to make DefenseNet see the proposed AdvCloud attack to defend it.

Generalized AdvCloud Branch. To benefit a black-box defense which makes DefenseNet robust to other cloud based adversarial examples generated by different attack methods never seen before, we design an Attack Generalization Module (AGM) to include the generalized AdvCloud images. We use two different levels of Gaussian noise to simulate the changes produced by the gradient-based learned exposure matrix ($\mathbf{E}$) and cloud mask ($\mathbf{M}$) under a specified budget. Specifically, we add Gaussian noise $\mathcal{N}_{\mathrm{E}} = \omega_{\mathrm{E}} \cdot \mathcal{N}(\cdot)$ and $\mathcal{N}_{\mathrm{M}} = \omega_{\mathrm{M}} \cdot \mathcal{N}(\cdot)$ to $\mathbf{E}$ and $\mathbf{M}$ respectively to obtain $\mathbf{E}_g$ and $\mathbf{M}_g$ so as to extend the distribution space of parameters around the gradient direction, where $\mathcal{N}(\cdot)$ is a standard Gaussian random noise generation function in the range of [-1, 1]. Then, we could acquire a generalized adversarial cloudy image $\hat{\mathbf{I}}_g$ with the generalized $\mathbf{E}_g$ and $\mathbf{M}_g$ via Eq. (2), *i.e.*,

$$\hat{\mathbf{I}}_g = \text{Cloud}(\hat{\mathbf{I}}, \mathbf{E}_g, \mathbf{M}_g), \tag{8}$$

as the second-branch input to the DefenseNet.

DefenseNet Loss. We feed adversarial cloudy images $\hat{\mathbf{I}}$ and $\hat{\mathbf{I}}_g$ to the DefenseNet to output the cloud-removed images $\mathbf{I}' = \text{DefenseNet}(\hat{\mathbf{I}}; \theta)$ and $\mathbf{I}'_g = \text{DefenseNet}(\hat{\mathbf{I}}_g; \theta)$ respectively, where θ means the parameter of DefenseNet. In the defense stage, the output cloud-removed images are optimized by the image reconstruction loss function L_r and regularization loss item L_{reg}. The objective loss function is shown below:

$$\mathcal{L} = L_r(\mathbf{I}', \mathbf{I}) + L_r(\mathbf{I}'_g, \mathbf{I}) + wL_{reg}(\mathbf{I}', \mathbf{I}'_g), \tag{9}$$

where $\mathbf{I}$ is the clean image for $\hat{\mathbf{I}}$ and $\hat{\mathbf{I}}_g$, and w is the balance weight which is set to 0.1, and L_r and L_{reg} loss functions are both implemented as L_1 loss.

The whole algorithm flow for the defense against the Adversarial Cloud based attack for remote sensing salient object detection is summarized in Algorithm 1.

Algorithm 1 Defense algorithm against the Adversarial Cloud based attack for remote sensing SOD.

```
Input: Clean images from the training set of EORSSD [13], $\epsilon_{\text{M}} = 0.03$, $\epsilon_{\text{E}} = 0.06$, iteration $K = 10$, $\alpha_{\text{M}} = 0.003$, $\alpha_{\text{E}} = 0.015$, a pre-trained remote sensing SOD detector $\phi(\cdot)$ [13], and a discriminator $\mathcal{D}$ pre-trained on training set.
Output: Adversarial Cloudy Images, parameter $\theta$ for DefenseNet.
1: repeat
2:     Attack Step:
         • Initial cloudy image synthesizing by Eq. (2) with $\mathbf{E_0}$ and $\mathbf{M_0}$.
         • Solve Eq. (6) via Eq. (7) to obtain optimal $\mathbf{E}$ and $\mathbf{M}$ with $K$ iterations for each image to learn the corresponding adversarial cloudy image $\hat{\mathbf{I}}$.
3:     Defense Step:
         • Obtain the generalized adversarial cloudy image $\hat{\mathbf{I}}_g$ via AGM and Eq. (8).
         • Solve Eq. (9) via AdamW optimizer [41] to obtain optimal $\theta$ via $\hat{\mathbf{I}}$, $\hat{\mathbf{I}}_g$ from the optimal $\mathbf{E}$ and $\mathbf{M}$.
4: until convergence or maximum epochs reached.
```

4. Experiments

4.1. Experimental Setting

Benchmark Datasets: To evaluate the salient object detection in remote sensing images, we use the public EORSSD dataset [13] to perform experiments. Using each clean image in EORSSD dataset, we generate its corresponding image with the normal cloud, leading to a new synthetic dataset named EORSSD$_c$. Similarly, adding the cloud-like attack to each clean image of EORSSD dataset, we could generate a new adversarial attacked cloudy image. Figure 4 shows some example images of EORSSD, EORSSD$_c$, and EORSSD with AdvCloud.

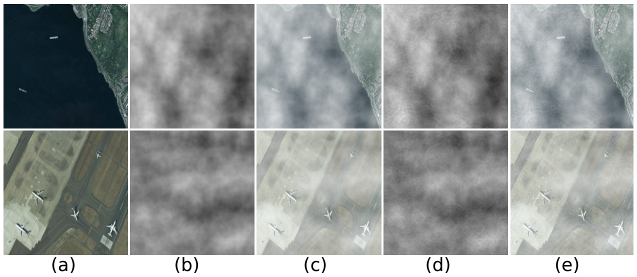


Figure 4. Example images of EORSSD, EORSSD$_c$, EORSSD with AdvCloud. (a) clean image of EORSSD, (b) synthesized normal cloud, (c) clean image with normal cloud leading to EORSSD$_c$, (d) proposed AdvCloud, (e) EORSSD with the proposed AdvCloud.

Evaluation Metrics: We evaluate the remote sensing salient object detection performance using F-measure (F$_\beta$), Mean Absolute Error (MAE) score and S-measure (S$_m$), same as those in [13].

Comparison Methods: For the attack experiment, we compare the proposed AdvCloud method with five additive perturbation based attack methods on the EORSSD$_c$ dataset, *i.e.*, FGSM [15], MIFGSM [28], PGD [29], VMIFGSM [44], and NIFGSM [45]. The maximum perturbation for these comparison methods is set to be 8 pixel values in [0, 255]. These comparison attack methods are applied on the testing images of EORSSD$_c$.

For the defense experiment, we compare our proposed DefenseNet with JPEG Compression [42], FFA-Net [43], and DefenseNet$_{FFA}$ (using FFA-Net as the backbone). The defense methods are all trained on EORSSD with AdvCloud generated by attacking DAFNet [13] which aims to remove the adversarial attack to obtain a clean image.

For evaluating the generalization ability of the proposed attack and defense methods, we additionally employ three SOD detectors, *i.e.*, BasNet [46], U^{2}Net [47], and RRNet [48]. All SOD models are trained on EORSSD dataset until convergence. Since the proposed AdvCloud are generated based on cloud, to ensure fairness in evaluating the effectiveness of different SOD models in attacking and defending against adversarial examples, the performance of four SOD models should treat the EORSSD$_c$ as the starting point for attacking rather than EORSSD.

Implementation Details: The SOD Network to be attacked is the deep learning based remote sensing salient object detection network DAFNet [13] pre-trained on the clean training images of EORSSD dataset. For the proposed

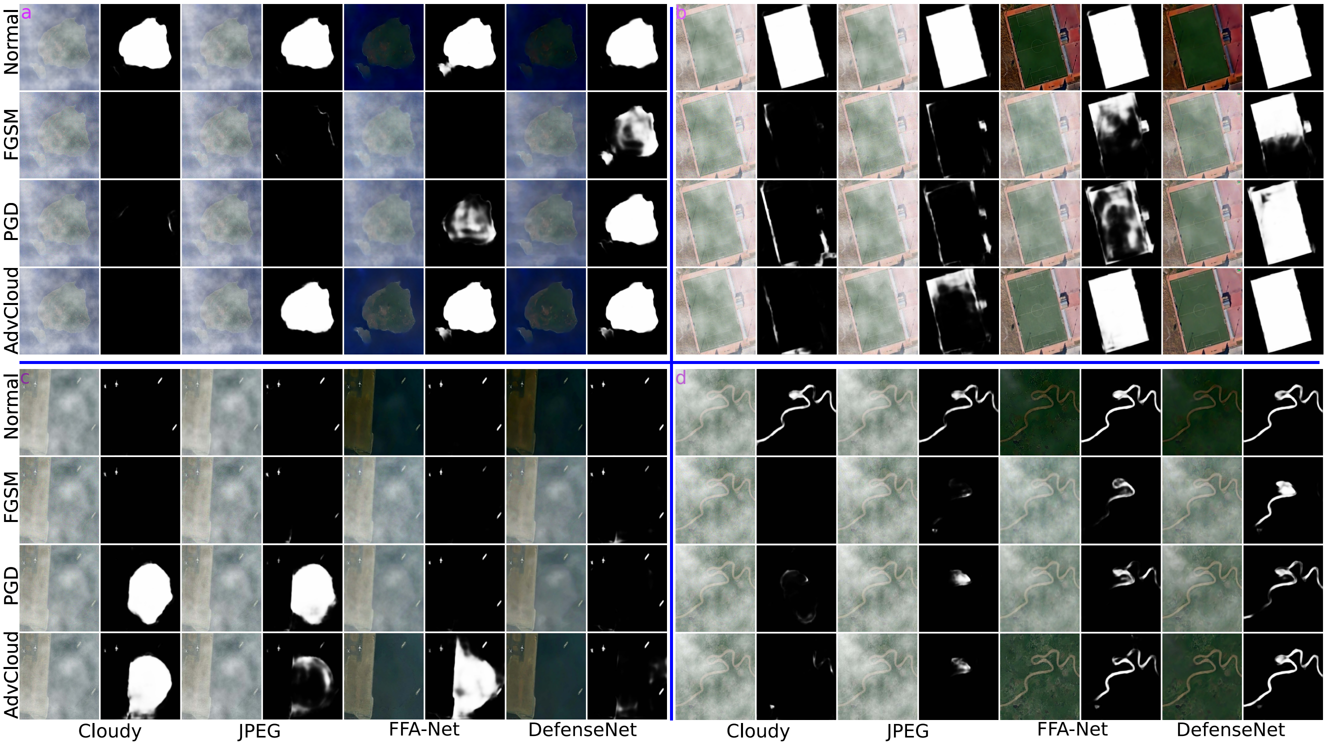


Figure 5. Defense against the remote sensing salient object detection attacks. From top to bottom: normal cloudy image, attacked cloudy images by FGSM [15], PGD [29], and the proposed AdvCloud. From left to right: cloudy images, defense images by JPG Compression [42], FFA-Net [43], proposed DefenseNet, each of which is followed by its corresponding SOD result.

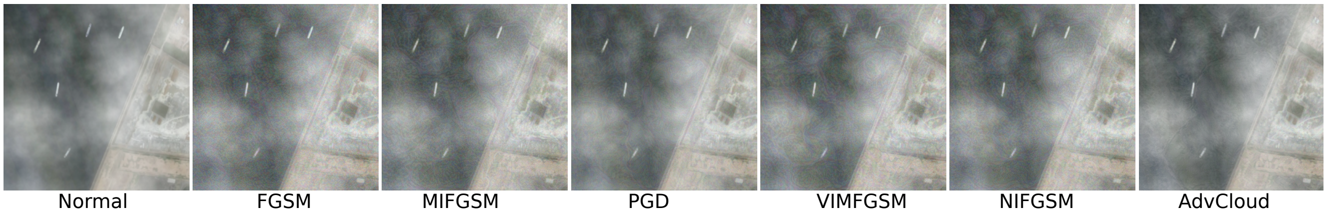


Figure 6. Visualization of normal and attacked cloud examples by different attacks. AdvCloud looks more similar to Normal.

AdvCloud attack, we set $\epsilon_{\mathrm{M}} = 0.03$ (8 pixel values in [0, 255]), $\epsilon_{\mathrm{E}} = 0.06$, and the generalization random noise range of ω_{M}, ω_{E} are 0.05 and 0.1, respectively. The input image is resized to 256×256. We use the AdamW optimization algorithm [41] for the network training with the following hyperparameters: learning rate as 0.0001, batch size as 8, and total epoch as 100. All the experiments were run on a single NVIDIA RTX 3090 GPU card.

4.2. Experimental Results

Attack Result. Table 1 shows the quantitative SOD performance for the baseline attack. When the dataset is clean, *i.e.*, no cloud is added, the target SOD network, DAFNet [13], achieves 0.9049 overall F-measure on EORSSD dataset. After normal clouds are added to the EORSSD dataset, the F-measure decreases to 0.8253. When the proposed AdvCloud is added to the EORSSD dataset, the SOD network is misled by the adversarial examples and the F-measure is 0.2572. This demonstrates that the proposed AdvCloud severely reduces the performance of the SOD network. Furthermore, we compare the proposed AdvCloud with other attack methods, as shown in Table 1. It shows that each attack method could effectively reduce the SOD performance Moreover, the white-box attacks on DAFNet can be effective to other SOD detectors with varying degrees of decline.

Figure 5 shows the qualitative comparisons among different attack methods and their corresponding SOD map. Due to the attack, some objects predicted by the SOD model are ignored (a, b, d) and misidentified (c) in Fig. 5. As we can observe, the proposed attacked image is very similar to the normal cloud in human perception compared to that from other attack methods. We can see visible defect and moire on the attacked images by other attack methods in Fig. 6. Therefore, the proposed AdvCloud is more visually close to

Table 1. Baseline SOD performance before and after attacks. We mark white-box attacks with * and highlight the best performance in red. The gray part shows the black-box attacks with other SOD models.

	Attack Performance	DAFNet [13]			BasNet [46]			U^2Net [47]			RRNet [48]		
		MAE ↑	F_β ↓	S_m ↓	MAE ↑	F_β ↓	S_m ↓	MAE ↑	F_β ↓	S_m ↓	MAE ↑	F_β ↓	S_m ↓
	Clean Image	0.0060	0.9049	0.9058	0.0162	0.8071	0.8871	0.0157	0.7890	0.8516	0.0077	0.9086	0.925
	Normal cloud	0.0126	0.8253	0.8540	0.0295	0.7270	0.8352	0.0359	0.6170	0.7410	0.0100	0.8345	0.8917
ATTACK	FGSM	0.0432 *	0.2880 *	0.5773 *	0.0381	0.5974	0.7488	0.0441	0.5027	0.6743	0.0202	0.6815	0.7937
	MIFGSM	0.0497 *	0.1292 *	0.5247 *	0.0452	0.5176	0.7063	0.0461	0.4666	0.6611	0.0208	0.6344	0.7695
	PGD	0.0680 *	0.1376 *	0.5166 *	0.0401	0.5860	0.7478	0.0426	0.5142	0.6869	0.0169	0.7026	0.8060
	VMIFGSM	0.0497 *	0.1326 *	0.5267 *	0.0463	0.4924	0.6952	0.0463	0.4564	0.6561	0.0245	0.5807	0.7416
	NIFGSM	0.0472 *	0.1519 *	0.5360 *	0.0439	0.5176	0.7108	0.0456	0.4698	0.6623	0.0213	0.6354	0.7735
	AdvCloud w/o Noise	0.0256 *	0.6583 *	0.7556 *	0.0311	0.7080	0.8198	0.0373	0.5930	0.7286	0.0120	0.8018	0.8671
	AdvCloud w/o Exposure Matrix	0.0484 *	0.4265 *	0.6435 *	0.0317	0.7026	0.8145	0.0379	0.5953	0.7265	0.0116	0.8103	0.8765
	AdvCloud	0.0714 *	0.2572 *	0.5609 *	0.0361	0.6396	0.7771	0.0404	0.5504	0.7072	0.0143	0.7484	0.8370

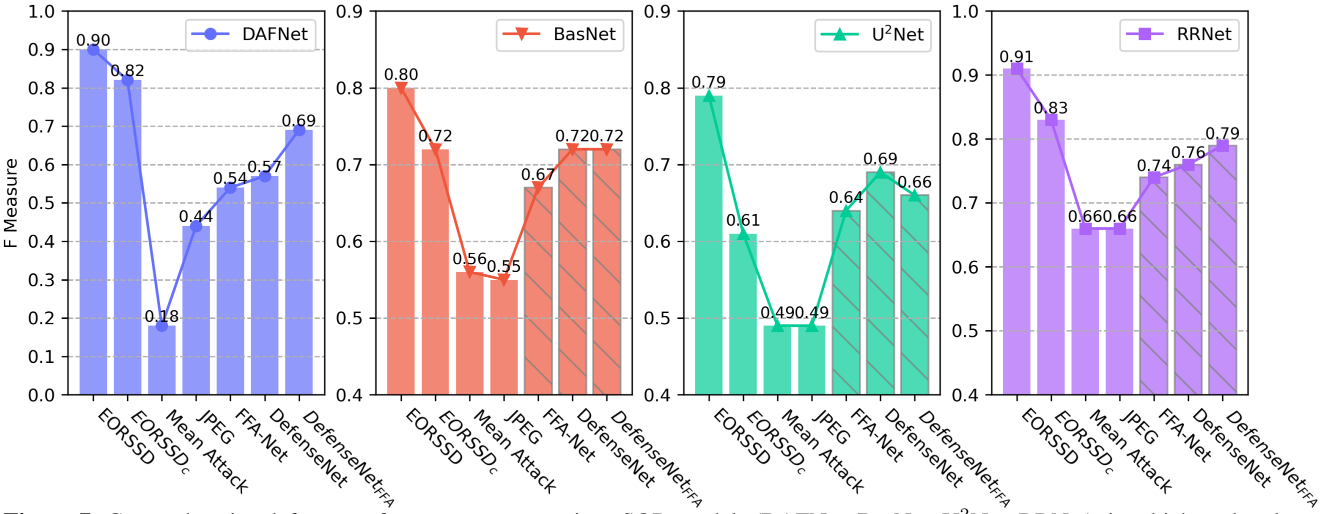


Figure 7. Comprehensive defense performance across various SOD models (DAFNet, BasNet, U^2Net, RRNet), in which each column represents the mean testing performance under different attack and defense scenarios. The EORSSD and EORSSD$_c$ represent each detector's performance under clean image and normal cloudy image (both without attack); the Mean Attack column shows the mean attacked performance under FGSM, MIFGSM, PGD, VMIFGSM, NIFGSM, and AdvCloud attacks trained with DAFNet; and the subsequent columns show the mean defense results by JPEG, FFA-Net, DefenseNet, and DefenseNet$_{FFA}$, respectively. The gray stripes indicate the machine learning based DAFNet-trained black-box defenses on other SOD models.

normal cloud but with very competitive attack performance.

Defense Result. Table 2 shows the detailed defense performance under different attack methods across various SOD models. It shows that the defense methods effectively improve the SOD performance after applying defense methods to adversarial examples generated by the attack strategies in Table 1. The Fig. 7 shows the comprehensive/summarized defense on all attack strategies with different SOD models. We can clearly see that the proposed defense method, *i.e.*, as a pre-processing step, achieves better F_β and S_m gains comparing with FFA-Net. The proposed DefenseNet could not only predominantly defend the proposed AdvCloud attack (*i.e.*, white-box defense) but also effectively defend other attack methods (*i.e.*, black-box defense). As shown in Table 2, the F_β performance gain by the proposed DefenseNet and DefenseNet$_{FFA}$ can be generalized to defend other attack methods. Despite the proposed defense method never sees the adversarial attack images created by other attack methods during training, the proposed defense method trained on AdvCloud still achieves better generalization performance to defend against other attack methods, with the help of the proposed AGM.

Ablation Study for Proposed DefenseNet. The proposed DefenseNet has two input branches, *i.e.*, regular attack image branch and generalized attack image branch. Table 3 shows both the regular attack branch and the generalized attack branch contribute to the final defense SOD performance, where the best defense performance is obtained when combining the two branches. If the branch of generalized attack is removed, it will lead to more significant defense performance drop. The DefenseNet contain AGM module can provide a promising and effective solution for generative defense on different adversarial attacks.

Discussion about Defense on Normal Cloudy Images. The DefenseNet$_{FFA}$'s performance in defense remote sensing SOD was assessed using normal cloudy images of EORSSD$_c$. The results in Table 4 indicate that the proposed defense mechanism is capable of effectively defend-

Table 2. Detailed defense performance across various SOD models. DefenseNet$_{FFA}$ means the proposed DefenseNet using FFA-Net as backbone. All attacks are trained with DAFNet, and the gray part highlights the black-box defense to different attacks on other SOD models.

	Defense Performance	DAFNet [13]			BasNet [46]			U^{2}Net [47]			RRNet [48]		
		MAE ↓	F$_\beta$ ↑	S$_m$ ↑	MAE ↓	F$_\beta$ ↑	S$_m$ ↑	MAE ↓	F$_\beta$ ↑	S$_m$ ↑	MAE ↓	F$_\beta$ ↑	S$_m$ ↑
JPEG	FGSM	0.0332	0.5084	0.6756	0.0389	0.5848	0.7410	0.0427	0.5112	0.6790	0.0204	0.6780	0.7900
	MIFGSM	0.0421	0.3409	0.6117	0.0434	0.5147	0.7082	0.0451	0.4625	0.6599	0.0205	0.6278	0.7681
	PGD	0.0323	0.5205	0.6939	0.0396	0.5851	0.7479	0.0418	0.5136	0.6874	0.0170	0.6874	0.8049
	VMIFGSM	0.0485	0.2710	0.5745	0.0464	0.4868	0.6926	0.0459	0.4518	0.6531	0.0242	0.5818	0.7413
	NIFGSM	0.0422	0.3575	0.6139	0.0433	0.5211	0.7088	0.0447	0.4671	0.6628	0.0215	0.6262	0.7685
	AdvCloud	0.0242	0.6228	0.7486	0.0363	0.6334	0.7756	0.0401	0.5524	0.706	0.0144	0.7353	0.8312
DefenseNet	FGSM	0.0260	0.6468	0.7548	0.0250	0.7182	0.8247	0.0237	0.6776	0.7865	0.0159	0.7668	0.8414
	MIFGSM	0.0569	0.4534	0.6651	0.0265	0.7017	0.8154	0.0237	0.6776	0.7865	0.0160	0.7476	0.8352
	PGD	0.0213	0.7244	0.8039	0.0265	0.7017	0.8154	0.0209	0.7210	0.8106	0.0126	0.7963	0.8656
	VMIFGSM	0.0762	0.3268	0.5917	0.0302	0.6689	0.7954	0.0260	0.6523	0.7704	0.0194	0.6912	0.8040
	NIFGSM	0.0516	0.4698	0.6689	0.0165	0.7508	0.8345	0.0241	0.6762	0.7844	0.0165	0.7508	0.8345
	AdvCloud	0.0128	0.8226	0.8572	0.0193	0.7496	0.8549	0.0173	0.7644	0.8368	0.0111	0.8365	0.8952
FFANet	FGSM	0.0292	0.5993	0.7309	0.0363	0.6260	0.7725	0.0369	0.5846	0.7264	0.0190	0.7015	0.8092
	MIFGSM	0.0535	0.4077	0.6427	0.0331	0.6607	0.7907	0.0322	0.6168	0.7485	0.0174	0.7185	0.8170
	PGD	0.0244	0.6861	0.7799	0.0332	0.6557	0.7873	0.0306	0.6439	0.7653	0.0139	0.7722	0.8529
	VMIFGSM	0.0692	0.3017	0.5838	0.0354	0.6280	0.7711	0.0334	0.5972	0.7367	0.0205	0.6722	0.7909
	NIFGSM	0.0484	0.4318	0.6518	0.0332	0.6557	0.7873	0.0330	0.6112	0.7441	0.0180	0.7210	0.8182
	AdvCloud	0.0145	0.7965	0.8443	0.0180	0.7826	0.8710	0.0165	0.7768	0.8462	0.0102	0.8423	0.8971
Defense$_{FFA}$	FGSM	0.0224	0.6995	0.7821	0.0316	0.6891	0.8095	0.0354	0.6072	0.7363	0.0136	0.7901	0.8561
	MIFGSM	0.0255	0.6488	0.7618	0.0279	0.7145	0.8244	0.0301	0.6479	0.7637	0.0138	0.7788	0.8551
	PGD	0.0149	0.7778	0.8313	0.0260	0.7294	0.8357	0.0288	0.6689	0.7781	0.0122	0.7971	0.8692
	VMIFGSM	0.0393	0.5338	0.6962	0.0291	0.6894	0.8113	0.0313	0.6278	0.7528	0.0159	0.7419	0.8306
	NIFGSM	0.0259	0.6512	0.7586	0.0276	0.7097	0.8227	0.0313	0.6378	0.7580	0.0139	0.7860	0.8573
	AdvCloud	0.0130	0.8178	0.8592	0.0171	0.7924	0.8761	0.0169	0.7834	0.8486	0.0097	0.8586	0.9031

Table 3. Ablation study for the SOD performance of DefenseNet to different attacks on DAFNet. DefenseNet‡: DefenseNet w/o Generalized AdvCloud, DefenseNet†: DefenseNet w/o Vanilla AdvCloud. The white-box defense to the attacks seen in training is shown in blue.

Attack Methods	DefenseNet‡			DefenseNet†			DefenseNet		
	MAE ↓	F$_\beta$ ↑	S$_m$ ↑	MAE ↓	F$_\beta$ ↑	S$_m$ ↑	MAE ↓	F$_\beta$ ↑	S$_m$ ↑
FGSM [15]	0.0373	0.4734	0.6652	0.0279	0.6161	0.7395	0.0260	0.6468	0.7548
MIFGSM [28]	0.0554	0.3144	0.5966	0.0600	0.4010	0.6399	0.0569	0.4534	0.6651
PGD [29]	0.0400	0.5256	0.6986	0.0267	0.6770	0.7783	0.0213	0.7244	0.8039
VMIFGSM [44]	0.0659	0.2271	0.5535	0.0754	0.2844	0.5760	0.0762	0.3268	0.5917
NIFGSM [45]	0.0517	0.3187	0.6004	0.0553	0.4027	0.6386	0.0516	0.4698	0.6689
Proposed AdvCloud	0.0249	0.7033	0.8011	0.0182	0.7477	0.8227	0.0128	0.8226	0.8572
Mean	0.0459	0.4271	0.6526	0.0439	0.5215	0.6992	0.0408	0.5740	0.7236

Table 4. Defense SOD performance of normal cloudy images of EORSSD$_c$ with the SOD detector DAFNet.

Methods	MAE ↓	F$_\beta$ ↑	S$_m$ ↑
Clean Image	0.0060	0.9049	0.9058
Normal Cloud	0.0126	0.8253	0.8540
JEPG Compression [42]	0.0139	0.7913	0.8367
DefenseNet	0.0171	0.7747	0.8315
FFA-Net [43]	0.0144	0.8079	0.8492
DefenseNet$_{FFA}$	**0.0126**	**0.8320**	**0.8620**

ing against anonymous types of attacks, while maintaining strong performance on normal images. This suggests that our defense method is reliable and effective in both attack and non-attack scenarios.

5. Conclusion

In this paper, we proposed a new Adversarial Cloud to attack the deep learning based remote sensing salient object detection model, meanwhile a new DefenseNet as pre-processing defense is proposed to purify the input image without tuning the deployed remote sensing deep SOD model. To study this research problem, we synthesized new benchmarks EORSSD$_c$ with normal clouds and the EORSSD with the adversarial cloud based attacks. The extensive experiments on four SOD models show that the proposed DefenseNet could well pre-process the attacked cloudy images as defense against different adversarial attack methods without changing the deployed remote sensing deep SOD model, while the SOD performance on the remote sensing normal cloudy images without attack is still promising.

6. Acknowledgement

This work was partially supported by NSF 2215388, CSU FRD grant, National Research Foundation, Singapore, and DSO National Laboratories under the AI Singapore Programme (AISG Award No: AISG2-GC-2023-008).

References

[1] Q. Wang, J. Lin, and Y. Yuan, "Salient band selection for hyperspectral image classification via manifold ranking," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 6, pp. 1279–1289, 2016.

[2] M. Zhang, W. Li, and Q. Du, "Diverse region-based cnn for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2623–2634, 2018.

[3] Y. Tian, C. Chen, and M. Shah, "Cross-view image matching for geo-localization in urban environments," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3608–3616, 2017.

[4] S. Zhu, T. Yang, and C. Chen, "Revisiting street-to-aerial view image geo-localization and orientation estimation," in *IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 756–765, 2021.

[5] H. Sun, J. Ma, Q. Guo, Q. Zou, S. Song, Y. Lin, and H. Yu, "Coarse-to-fine task-driven inpainting for geoscience images," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[6] Q. Zou, L. Ni, T. Zhang, and Q. Wang, "Deep learning based feature selection for remote sensing scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 11, pp. 2321–2325, 2015.

[7] S. Song, H. Yu, Z. Miao, Q. Zhang, Y. Lin, and S. Wang, "Domain adaptation for convolutional neural networks-based remote sensing scene classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 8, pp. 1324–1328, 2019.

[8] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[9] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning roi transformer for oriented object detection in aerial images," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2849–2858, 2019.

[10] L. Bruzzone and D. F. Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 452–466, 2002.

[11] B. Du, L. Ru, C. Wu, and L. Zhang, "Unsupervised deep slow feature analysis for change detection in multi-temporal remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 12, pp. 9976–9992, 2019.

[12] C. Li, R. Cong, J. Hou, S. Zhang, Y. Qian, and S. Kwong, "Nested network with two-stream pyramid for salient object detection in optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 11, pp. 9156–9166, 2019.

[13] Q. Zhang, R. Cong, C. Li, M.-M. Cheng, Y. Fang, X. Cao, Y. Zhao, and S. Kwong, "Dense attention fluid network for salient object detection in optical remote sensing images," *IEEE Transactions on Image Processing*, vol. 30, pp. 1305–1317, 2020.

[14] R. Gao, Q. Guo, F. Juefei-Xu, H. Yu, H. Fu, W. Feng, Y. Liu, and S. Wang, "Can you spot the chameleon? adversarially camouflaging images from co-salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[15] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *International Conference on Learning Representations*, 2015.

[16] R. Gao, Q. Guo, F. Juefei-Xu, H. Yu, and W. Feng, "Advhaze: Adversarial haze attack," *arXiv preprint arXiv:2104.13673*, 2021.

[17] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu, "Defense against adversarial attacks using high-level representation guided denoiser," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1778–1787, 2018.

[18] H. Zhang and J. Wang, "Defense against adversarial attacks using feature scattering-based adversarial training," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] Y. Xu, B. Du, and L. Zhang, "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*, vol. 59, no. 2, pp. 1604–1617, 2020.

[20] H. Sun, Y. Lin, Q. Zou, S. Song, J. Fang, and H. Yu, "Convolutional neural networks based remote sensing scene classification under clear and cloudy environments," in *IEEE/CVF International Conference on Computer Vision Workshop*, pp. 713–720, 2021.

[21] E. Li, S. Xu, W. Meng, and X. Zhang, "Building extraction from remotely sensed images by integrating saliency cue," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 3, pp. 906–919, 2016.

[22] L. Ma, B. Du, H. Chen, and N. Q. Soomro, "Region-of-interest detection via superpixel-to-pixel saliency analysis for remote sensing image," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 12, pp. 1752–1756, 2016.

[23] Q. Zhang, L. Zhang, W. Shi, and Y. Liu, "Airport extraction via complementary saliency analysis and saliency-oriented active contour model," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 7, pp. 1085–1089, 2018.

[24] Z. Liu, D. Zhao, Z. Shi, and Z. Jiang, "Unsupervised saliency model with color markov chain for oil tank detection," *Remote Sensing*, vol. 11, no. 9, p. 1089, 2019.

[25] C. Dong, J. Liu, F. Xu, and C. Liu, "Ship detection from optical remote sensing images using multi-scale analysis and fourier hog descriptor," *Remote Sensing*, vol. 11, no. 13, p. 1529, 2019.

[26] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014.

[27] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *International Conference on Learning Representations Workshop*, 2017.

[28] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[29] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018.

[30] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *IEEE Symposium on Security and Privacy*, 2017.

[31] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International Conference on Machine Learning*, 2020.

[32] F. Croce and M. Hein, "Minimally distorted adversarial examples with a fast adaptive boundary attack," in *International Conference on Machine Learning*, 2020.

[33] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

[34] J. Lu, T. Issaranon, and D. Forsyth, "Safetynet: Detecting and rejecting adversarial examples robustly," in *International Conference on Computer Vision*, 2017.

[35] C. Mao, Z. Zhong, J. Yang, C. Vondrick, and B. Ray, "Metric learning for adversarial robustness," in *Advances in Neural Information Processing Systems*, 2019.

[36] Y. Zhong and W. Deng, "Adversarial learning with margin-based triplet embedding regularization," in *International Conference on Computer Vision*, 2019.

[37] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!," in *Advances in Neural Information Processing Systems*, 2019.

[38] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International Conference on Machine Learning*, 2019.

[39] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, "Improving adversarial robustness requires revisiting misclassified examples," in *International Conference on Learning Representations*, 2019.

[40] H. Wang, Y. Deng, S. Yoo, H. Ling, and Y. Lin, "Agkd-bml: Defense against adversarial attack by attention guided knowledge distillation and bi-directional metric learning," in *IEEE/CVF International Conference on Computer Vision*, pp. 7658–7667, 2021.

[41] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[42] N. Das, M. Shanbhogue, S.-T. Chen, F. Hohman, S. Li, L. Chen, M. E. Kounavis, and D. H. Chau, "Shield: Fast, practical defense and vaccination for deep learning using jpeg compression," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 196–204, 2018.

[43] X. Qin, Z. Wang, Y. Bai, X. Xie, and H. Jia, "Ffa-net: Feature fusion attention network for single image dehazing," in *AAAI Conference on Artificial Intelligence*, pp. 11908–11915, 2020.

[44] X. Wang and K. He, "Enhancing the transferability of adversarial attacks through variance tuning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1924–1933, 2021.

[45] J. Lin, C. Song, K. He, L. Wang, and J. E. Hopcroft, "Nesterov accelerated gradient and scale invariance for adversarial attacks," *arXiv preprint arXiv:1908.06281*, 2019.

[46] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7479–7489, 2019.

[47] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U2-net: Going deeper with nested u-structure for salient object detection," *Pattern Recognition*, vol. 106, p. 107404, 2020.

[48] R. Cong, Y. Zhang, L. Fang, J. Li, Y. Zhao, and S. Kwong, "Rrnet: Relational reasoning network with parallel multiscale attention for salient object detection in optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2021.