Natural Weather-Style Black-Box Adversarial Attacks Against Optical Aerial Detectors

Guijian Tang, Wen Yao, Tingsong Jiang, Weien Zhou, Yang Yang, and Donghua Wang

***Abstract*— Most existing adversarial attack methods against detectors involve adding adversarial perturbations to benign images to synthesize adversarial examples. However, directly applying these methods, originally designed for natural image detectors, to optical aerial image detectors can lead to perturbations that appear unnatural and suspicious to human eyes, owing to intrinsic dissimilarities between these two types of images. Inspired by the fact that the captured optical aerial images are heavily affected by weather conditions, this article proposes a novel method for conducting adversarial attacks against optical aerial detectors by leveraging natural weather-style perturbations. Compared to existing methods, our scheme produces more natural and stealthy adversarial examples. To enhance the practicality of the proposed method in real-world scenarios, we implement the attacks in black-box settings where only the model's predictions are accessible. Specifically, we formulate the generation of adversarial weather perturbations in black-box as an optimization problem and effectively solve it using the differential evolution (DE) algorithm. Through extensive experiments, we verify the effectiveness of our method and investigate the transferability of generated adversarial examples across different models. In light of the significant generalization and effectiveness of our method, we generate and release the first dataset with adversarial weather-style perturbations based on the DOTA dataset, which we abbreviate as DOTA-W. This dataset serves as a valuable resource for evaluating and improving the robustness of optical aerial detectors. The code and dataset have been released at https://github.com/tang-agui/attADs-AWP.**

***Index Terms*— Black-box adversarial attacks, natural weather-style perturbations, optical aerial imagery detectors.**

I. Introduction

Deep neural networks (DNNs) have exhibited exceptional performance in solving visual problems in recent years [1], [2], [3], [4]. However, the inherent vulnerability of DNNs to adversarial examples has come to light. Adversarial examples are intentionally crafted inputs intended to cause well-trained DNNs to produce incorrect outputs without being perceptible to humans. The existence of adversarial examples was first disclosed by Szegedy et al. [5] in the context of natural images against classifiers. Subsequently, various adversarial attacks have been proposed, ranging from classifiers [6], [7], [8], [9] to object detectors [10], [11], [12]. Regrettably, the vulnerability to adversarial examples also manifests in the domain of aerial detectors.

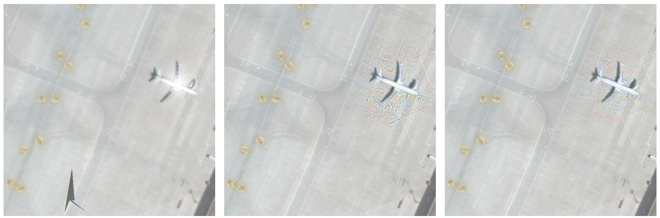


Fig. 1. Comparison results with the state-of-the-art methods on the *Plane* category. Although the target is successfully hided from the detector in all cases, the perturbations generated by our method are more natural and unsuspicious. Best viewed with zooming. (a) Ours. (b) RAP [13]. (c) DAG [14].

Most early works crafted adversarial examples by injecting carefully elaborated perturbations into benign images. In order to improve the imperceptibility and stealthiness of the generated adversarial examples, researchers have employed various strategies. On the one hand, attackers often impose constraints on the magnitude of the perturbations, such as the commonly used $L_p, (p = 0, 1, 2, \infty)$ constraints [13], [14], [15]. On the other hand, efforts have been made to develop natural-looking adversarial perturbations [10], [16], [17]. However, directly applying these methods to evade optical detectors presents new challenges. Unlike natural images captured from the ground, optical aerial images generally display more monochromatic color variations, rendering them visually less salient. As a result, in order to achieve a satisfactory attack, the optimized adversarial perturbations added to the images often become conspicuous to human observers. A comparison of examples highlighting this issue is depicted in Fig. 1.

In the real world, images captured by cameras are significantly influenced by weather factors, such as snow, fog, shadows, rain, etc. Different weather conditions can result in substantial variations in the quality of processed images. Consequently, researchers have conducted numerous studies to evaluate the performance of proposed methods under different weather conditions, aiming to provide a more comprehensive assessment. For example, Eykholt et al. [10]

Manuscript received 4 July 2023; revised 27 August 2023; accepted 8 September 2023. Date of publication 13 September 2023; date of current version 2 October 2023. This work was supported by the National Natural Science Foundation of China under Grant 52005505 and Grant 62001502. *(Corresponding author: Wen Yao.)*

Guijian Tang is with the College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China, also with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing 100071, China, and also with the Intelligent Game and Decision Laboratory, Beijing 100071, China (e-mail: tangbanllniu@163.com).

Wen Yao, Tingsong Jiang, Weien Zhou, and Yang Yang are with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing 100071, China, and also with the Intelligent Game and Decision Laboratory, Beijing 100071, China (e-mail: wendy0782@126.com).

Donghua Wang is with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing 100071, China, also with the Intelligent Game and Decision Laboratory, Beijing 100071, China, and also with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China.

Digital Object Identifier 10.1109/TGRS.2023.3315053

1558-0644 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

conducted experiments in both indoor and outdoor environments to illustrate the effectiveness of their proposed method. Zhao et al. [12] evaluated their methods separately under sunny and cloudy days. Zhang et al. [18] elaborated a dataset to evaluate the robustness of detection models in the physical world. To cover, they also chose different weather conditions to synthesize the dataset. The field of autonomous driving has also witnessed related work in recent years. For instance, [19] and [20] have incorporated diverse weather conditions as critical factors for evaluating and benchmarking the robustness of 3-D object detection in autonomous driving. These observations raise two fundamental questions: Can these naturally occurring weather factors be leveraged as adversarial perturbations? Can we deceive detectors by optimizing these factors?

Based on the aforementioned analysis, this article explores the use of natural weather-style perturbations for adversarial attacks. Specifically, we generate adversarial examples by incorporating snow, fog, shadow, and sun flares into the benign images. While there have been several works conducting adversarial attacks by modeling rain factors, they are mostly focused on attacking natural images captured from the ground. For instance, Zhai et al. [21] have proposed an innovative adversarial rain attack to simulate diverse rain conditions and explore the potential threats to classifiers and detectors. However, it's important to acknowledge that optical aerial images differ significantly from natural images in terms of spectrum, perspective, and orientation, and thus need to be considered independently. Additionally, although rainfall is another prevalent weather phenomenon, it is not directly observable in the optical aerial imagery. Instead, we observe them as puddles on the ground, visually resembling sun flares. Therefore, rainfall is not considered in this study. Considering that the noise introduced by these weather factors appears natural to human observers, we believe that the generated adversarial examples are more deceptive and stealthy, minimizing the likelihood of raising alarms.

Furthermore, adversarial attacks can be categorized as white-box attacks [7], [11], [22], [23] and black-box attacks [9], [24], [25] based on the level of knowledge possessed by the attackers. In white-box attacks, the adversaries have complete access to internal information of the model, including its weights, training parameters, etc. Conversely, in black-box attacks, attackers can only acquire input–output pairs from the target models. While white-box attacks often yield higher attack success rates (ASRs), access to model-specific information is typically restricted and unavailable in real-world scenarios. Therefore, conducting effective attacks in black-box settings presents both practicality and challenges.

The generation of adversarial examples in black-box settings can be formulated as an optimization problem. The goal of this article is to find optimal weather-style perturbations within L_p constraints to deceive optical aerial detectors and induce incorrect predictions. To tackle this challenge, we employ differential evolution (DE), a powerful population-based evolutionary optimization technique [26], [27], to solve it. DE is a gradient-free and problem-specific approach that solely relies on the predicted outputs of the model, making it a suitable tool for black-box attacks.

In summary, the main contributions of this article are as follows.

1) We propose a novel adversarial attack method based on natural weather-style perturbations. Compared with the existing attack methods, the adversarial examples generated by our proposed method are more natural and stealthy.
2) We model the generation of adversarial weather perturbations in black-box settings as an optimization problem and utilize the DE algorithm to solve it effectively under both constraints of L_∞ and L_2, respectively.
3) We conduct intensive experiments to verify the effectiveness of our method and study the transferability of generated adversarial examples between different models. Additionally, we also analyze the performance of generated adversarial examples under typical defense mechanisms, and our experimental results demonstrate the strong attack robustness of the proposed method.
4) Based on our proposed method, we generate and release the first dataset for optical aerial detection, dubbed DOTA-W. Acting as a potential benchmark dataset for evaluating the improving robustness of optical aerial detectors, DOTA-W was built upon the subset of DOTA-v1.0 [28] validation set.

The rest of the article is organized as follows. The related works are reviewed in Section II. In Section III, we overview the framework of the proposed method and introduce the details of loss functions designed for generating adversarial patches against aerial detectors, followed by experimental assessments in Section IV. Further discussions and conclusions are summarized in Section V.

II. Related Work

In this section, we provide an overview of related work on white-box and black-box attacks.

A. White-Box Attacks

Previous studies have shown that adversarial examples generated by adding imperceptible adversarial perturbations to clean images can cause DNNs to produce incorrect labels. In white-box scenarios, Szegedy et al. [5] first introduced adversarial examples against classifiers, and subsequent attack methods such as FGSM [6], PGD [7], C&W [8], and Deepfool [29] have been proposed. For object detectors, Xie et al. [14] made their first effort and proposed dense adversary generation (DAG), which simultaneously attacks all targets by iteratively optimizing the loss function using back-propagation. Li et al. [13] proposed Robust adversarial perturbation (RAP) to attack deep proposal-based object detectors. Similarly, Huang et al. [30] developed RPAttack to evade both one-stage and two-stage detectors, employing a novel pixel selection and refining scheme to remove inconsequential perturbations gradually. Inspired by [30], Sun et al. [15] proposed an improved technique that identifies the most critical sub-regions and utilizes a novel objective function to avoid the

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

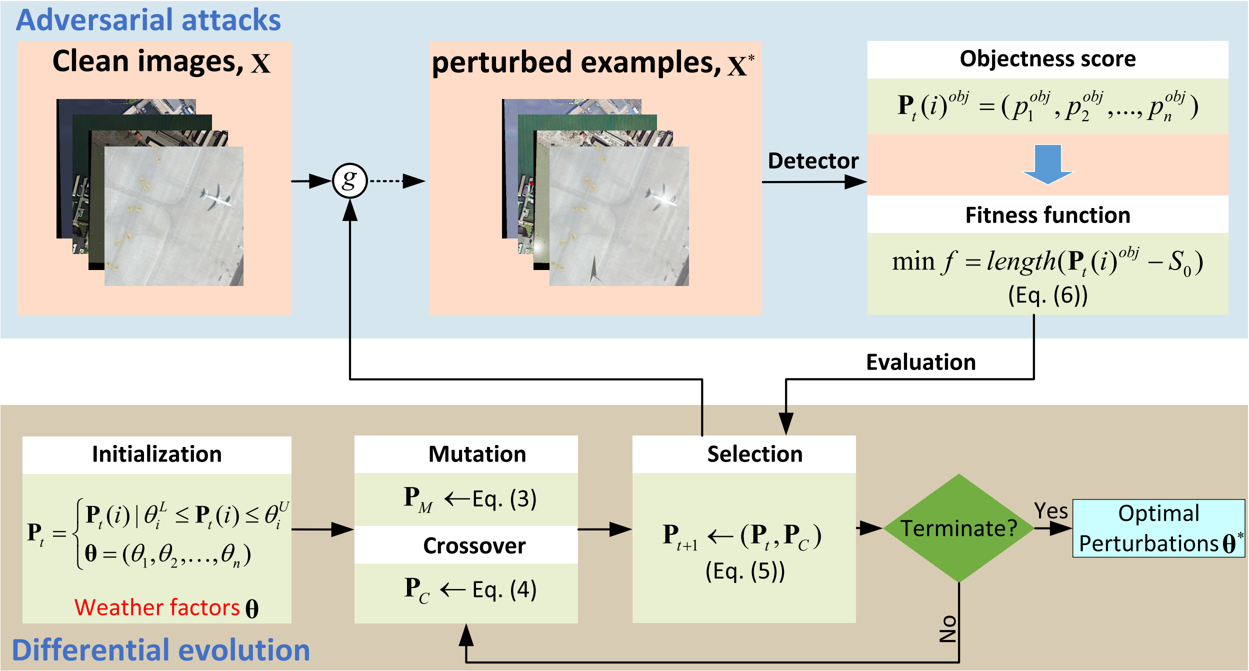

Adversarial attacks
Clean images, $\mathbf{X}$
g
perturbed examples, $\mathbf{X}^*$
Detector
Objectness score
$\mathbf{P}_t(i)^{obj} = (p_1^{obj}, p_2^{obj}, \ldots, p_n^{obj})$
Fitness function
$\min f = length(\mathbf{P}_t(i)^{obj} - S_0)$
(Eq. (6))
Evaluation
Initialization
$\mathbf{P}_t = \begin{cases} \mathbf{P}_t(i) \mid \theta_i^L \le \mathbf{P}_t(i) \le \theta_i^U \\ \boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n) \end{cases}$
Weather factors $\boldsymbol{\theta}$
Mutation
$\mathbf{P}_M \leftarrow$ Eq. (3)
Crossover
$\mathbf{P}_C \leftarrow$ Eq. (4)
Selection
$\mathbf{P}_{t+1} \leftarrow (\mathbf{P}_t, \mathbf{P}_C)$
(Eq. (5))
Terminate?
Yes
No
Optimal Perturbations $\boldsymbol{\theta}^*$
Differential evolution

Fig. 2. Overview of the proposed method. First, the population is initialized randomly according to the preset weather factors $\boldsymbol{\theta}$. Second, the offspring population is generated through mutation, crossover, and selection operations. Third, after obtaining the adversarial weather perturbations, candidate adversarial examples $\mathbf{X}^*$ can be synthesized by adding these perturbations to the clean images $\mathbf{X}$. Next, the candidates are fed into the target aerial detector, and we extract the objectness score from the predictions to construct the fitness function. Finally, we repeat the evolution process until the stop criterion is satisfied. More details refer to Algorithm 1.

problem of gradient inundation when directly applying [30] to attack optical aerial detectors.

The aforementioned works primarily focus on conducting pixel-level adversarial attacks. In recent years, adversarial patches have been widely applied and deployed in the physical world to targeted optical aerial object detectors [10], [31]. Lian et al. [32] investigated the feasibility of using adversarial patches as a means of camouflaging objects in aerial imagery. Lian et al. [33] and [34] introduced contextual background attack (CBA), in which contextual background adversarial patches are employed to protect ground objects from being detected. Tang et al. [35] demonstrated that utilizing intermediate layer outputs of the target model rather than the final output can enhance the attack efficiency against aerial detectors. The approaches mentioned above are all performed in white-box settings, where attackers can iteratively update the perturbations using the model's gradients. However, in real-world scenarios, the knowledge of the target model is typically private and inaccessible, leading to a growing interest in black-box adversarial attacks.

B. Black-Box Attacks

Traditional gradient-based methods cannot be used in black-box settings to generate adversarial examples due to the unavailability of gradients from the target model. Black-box attacks can be further categorized into two main categories: transfer-based and query-based methods [36], [37]. Transfer-based methods argue that the adversarial examples generated against one model will likely deceive another similar model. Researchers utilize the adversarial examples generated by applying existing white-box attack schemes to evade target models without querying them. Representative works in this category include [37], [38], [39], and [40].

Query-based methods can be subdivided into two groups. The first type involves estimating the gradients of the target model through a series of queries. For example, Chen et al. [41] proposed zeroth-order optimization (ZOO) based methods to directly estimate the gradients of the target model, while Ilyas et al. [42] used a natural evolution strategy (NES) to approximate the gradients. Although these works achieve comparable attack performance in most cases, the need for a large number of queries brings a significant computational burden. The second type of query-based method refers to population-based stochastic optimization techniques, where attackers randomly search populations to generate adversarial examples without using the gradients of the model [43], making them suitable for black-box settings. Su et al. [26] considered an extreme scenario where only one pixel can be modified and utilized the DE algorithm to determine the most important pixel that can fool the models. Ghosh et al. [44] proposed the DEceit algorithm, which constructs effective pixel-restricted perturbations using only black-box feedback from the victim model. Additionally, Wei et al. [45] introduced the adversarial attributes attack, successfully optimizing adversarial examples using the DE algorithm. In this work, we focus on utilizing the DE algorithm, a typical population-based optimization technique, to generate adversarial examples.

Our work was partially inspired by [21] and [45]. Zhai et al. [21] designed a rain generator to synthesize rainy images and presented an adversarial rain attack against image classifiers and object detectors, respectively. Wei et al. [45]

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

conducted a black-box attack on typical classifiers by manipulating picture attributes, such as brightness, contrast, and chroma. Compared to theirs, there are two main differences between our work and theirs. First, we specifically focus on conducting experiments on optical aerial images, while their work focuses on natural images captured on the ground. Second, we establish adversarial attacks based on existing techniques without developing any new techniques, showing the flexibility of our method.

III. Methodology

In this section, we first provide a formulation of adversarial attacks against detectors. Subsequently, we report the natural weather-style adversarial perturbations, followed by the description of the DE algorithm employed in this article.

A. Problem Formulation

Considering a benign image $\mathbf{x_0}$, the attacker dedicated to optimizing a malicious perturbation $\mathcal{E}$ that is added to the image to generate an adversarial example $\mathbf{x}'$, which can deceive a victim model into producing wrong predictions. Technically, the generation of adversarial perturbations $\mathcal{E}$ in this article can be formulated as

$$\begin{aligned} &\min\ \|\mathcal{E}\|_p \\ &\text{s.t. } \mathcal{O}(\mathbf{x}') \neq \mathcal{O}(\mathbf{x}) \end{aligned} \tag{1}$$

where $\mathbf{x}' = \mathbf{x} + \mathcal{E}$, $\mathcal{O}$ denotes the victim model, where the attackers have no access to its internal information but can obtain its predictions in this article. For object detection, the model predicts a set of bounding boxes $\mathcal{O}(\mathbf{x}) = \{o_1, o_2, \ldots, o_n\}$ where $o_i = (b_i^x, b_i^y, b_i^w, b_i^h, p_i^{\text{obj}}, \mathbf{P}_i^{\text{class}})$, including the predicted location $(b_i^x, b_i^y, b_i^w, b_i^h)$ where (b_i^x, b_i^y) represent the coordinates of the box and (b_i^w, b_i^h) is the width and height of the box, an objectness score of p_i^{obj} being a real object, and a group of class probability vectors $\mathbf{P}_i^{\text{class}} = (p_1, p_2, \ldots, p_n)$ associated with the bounding box. Equation (1) aims to find the minimum visual distortion, constrained by L_p norm, to mislead the detector, that is, $\mathcal{O}(\mathbf{x}') \neq \mathcal{O}(\mathbf{x_0})$.

B. Adversarial Weather Perturbations

Equation (1) is a general paradigm for conducting adversarial attacks. Since the detector will predict the location, class probabilities, and objectness score of all instances in an image, (1) can be expressed in different forms depending on the distinct purpose of attackers. Following the discussion in [35], the goal of this article is to minimize the objectness score of instances, aiming to realize an adversarial attack effect where the detector cannot detect existing objects in the scene. Therefore, (1) can be reformulated as

$$\mathcal{E}^* = \arg\min_{\mathcal{E}}\ \mathbf{P}^{\text{obj}}(g(\mathbf{x}, \mathcal{E})), \quad \text{s.t. } \|\mathcal{E}\|_p \leq \epsilon \tag{2}$$

where $\mathbf{P}^{\text{obj}} = (p_1^{\text{obj}}, p_2^{\text{obj}}, \ldots, p_n^{\text{obj}})$ denotes the objectness score vector predicted for all instances in the image, $g(\mathbf{x}, \mathcal{E})$ represents the perturbations operator. For cases of directly adding perturbations to the image, $g(\mathbf{x}, \mathcal{E}) = \mathbf{x} + \mathcal{E}$, while in our scenarios of synthesizing adversarial weather-style perturbations, $g(\mathbf{x}, \mathcal{E})$ is an implicit function, details would be outlined in Section IV-A4. Here, $\mathcal{E} = (\theta_1, \theta_2, \ldots, \theta_m)$ are parameters to control the intensity of adversarial weather perturbations, where θ_i represents the ith variable, and m denotes the number of used weathers factors.

From (2), it is evident that the magnitude of perturbations is constrained by the L_p bound $\|\cdot\|_p$. We consider both $p = 2$ and $p = \infty$ to verify the effectiveness of our method, respectively. Specifically, when $p = \infty$, $\|\mathcal{E}\|_\infty$ equals $|\theta_i| \leq \epsilon$, allowing independent evolution of each factor within the range of $[-\epsilon, \epsilon]$, and as for $p = 2$, $\|\mathcal{E}\|_2 = (\theta_1^2 + \theta_2^2 +, \ldots, +\theta_m^2))^{1/2} \leq \epsilon$, implying that optimization of one factor must consider the influence of other factors. We delve into the distinctions between these two constraint strategies and provide a potential explanation in Section IV-B.

C. Differential Evolution

This section first gives an overview of the DE algorithm, and then we introduce the fitness function designed for solving the problem raised in this article.

1) Overview: DE is a widely used evolutionary algorithm that employs random population selection iteratively to search for optimal solutions. Since DE does not rely on gradient information from the model, it is commonly applied to solve various black-box optimization problems. The DE algorithm typically consists of population initialization, mutation, crossover, and selection operations.

In our study on adversarial weather-style perturbations, the population can be expressed as: $\mathbf{P}_t = \{\mathbf{P}_t(i) | \theta_i^L \leq \mathbf{P}_t(i) \leq \theta_i^U, 1 \leq i \leq N, 1 \leq t \leq T\}$, where N denotes the population size and T is the maximum allowable evolutionary generations. $\mathbf{P}_t(i)$ is the ith individual in the tth step. Individuals in every iteration are randomly sampled from the distribution within the ranges of (θ_i^L, θ_i^U). In our attack scenarios, as our optimization variables consist of multiple numeric types (float and integer), the boundaries of different variables are inconsistent. More details can be found in Section IV-A4. Subsequently, a mutation operator is employed to generate candidate solutions by

$$\mathbf{P}_M(i) = \mathbf{P}_t(r_1) + F(\mathbf{P}_t(r_2) - \mathbf{P}_t(r_3)), \quad r_1 \neq r_2 \neq r_3 \neq i \tag{3}$$

where F is the scale parameter set by default to be 0.5 [46], and $r1$, $r2$, $r3$ are randomly chosen from N without replacement. Following mutation, we go through a crossover operation to generate new offspring:

$$\mathbf{P}_C(i) = \begin{cases} \mathbf{P}_M(i), & \text{rand} < C_r \\ \mathbf{P}_t(i), & \text{otherwise} \end{cases} \tag{4}$$

where C_r denotes the crossover probability. Finally, in the selection stage, the DE retains the better individuals by competing with the parent population $\mathbf{P}_t$ with the corresponding

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

offspring population $\mathbf{P}_C$

$$\mathbf{P}_{t+1}(i) = \begin{cases} \mathbf{P}_t(i), & f(\mathbf{P}_t(i)) < f(\mathbf{P}_C(i)) \\ \mathbf{P}_C(i), & \text{otherwise} \end{cases} \tag{5}$$

where f represents the fitness function used to evaluate the quality of per individual.

Although the DE algorithm is a very effective method for solving black-box problems and has been successfully used to generate adversarial examples, the existing techniques are all targeted at classifiers. Because of the large gap between the output of classifiers and detectors, the existing fitness function designed for classifiers [9], [26], [44] cannot be directly utilized for detectors. In Section III-C2, we will report a fitness function elaborated for detectors, which can effectively solve the optimization problem proposed in this article.

2) Fitness Function: The fitness function plays a crucial role in the selection of individuals within the DE algorithm and is closely tied to the optimization problem at hand. As discussed in Sections III-A and III-B, the primary objective of this article is to optimize an adversarial weather-style perturbation solution that manipulates benign images in a way that objects cannot be detected by detectors. To accomplish this objective, the fitness function can be outlined as

$$\min f = \text{length}\big(\mathbf{P}_t(i)^{\text{obj}} - S_0\big) \tag{6}$$

where $\mathbf{P}_t(i)^{\text{obj}} = (p_1^{\text{obj}}, p_2^{\text{obj}}, \ldots, p_n^{\text{obj}})$ signifies the predicted objectness score vector of the ith individual in the tth iteration step, and S_0 represents a predefined threshold. Through the minimization of (6), the objectness score of the targets can be effectively decreased below a specified threshold, ultimately achieving a successful vanishing attack.

Based on the above analysis, this paper combines the differential evolution algorithm to conduct black-box adversarial attacks against aerial detectors based on weather factors. The overview of the proposed method is illustrated in Fig. 2, withdetailed implementation providedin Algorithm 1. The algorithm takes inputs including the clean image $\mathbf{x}$, perturbation dimension m (total number of preset weather factors), population size N, target detection algorithm D, confidence threshold S_0, and the early stopping monitor *flag* = 0. Specifically, line 1 randomly initializes the population $\mathbf{P}_0$ based on the N and m. Following this, lines 3–6 perform mutation and crossover operations to generate new individuals of $\mathbf{P}_M$ and $\mathbf{P}_C$, along with the corresponding objectness score vector extracted from the detector's output. Subsequently, individual selection is conducted according to (6), as outlined in lines 7–16. During this selection process, the number of targets $\mathbf{L}$ (as indicated in lines 12 and 15) that are still detected by the detector is also recorded. If any element of $\mathbf{L}$ equals 0, it signifies that all targets in the image have been successfully attacked, signaling that the optimization should stop. In this case, the *flag* is switched to 1, and we return the corresponding index i.

IV. Experiments

In this section, we first report the datasets, victim models, evaluation metrics, and experiment implementation. Then, we validate the effectiveness of the proposed method on variant detectors and categories, and conduct an ablation to evaluate the impact of a single weather perturbation on the attack performance. Next, we analyze the transferability of generated adversarial examples across different models. Finally, we assess the robustness of the proposed method against various defense mechanisms.

Algorithm 1 Pseudocode of Natural Weather-Style Black-Box Attacks Using DE

Input: Detector D, Benign image $\mathbf{x}$, perturbation dimensions m, population size N, maximum evolutionary generations T, fitness function f, detection score $\mathbf{P}^{obj} \in \mathbf{R}^{N\times n}$, early stopping monitor *flag* $= 0$.

Output: The $index$ of the best individual.

```
1:  $\mathbf{P}_0 = Init(N, m)$, initialization
2:  for $t < T$ do
3:      $\mathbf{P}_M = Mutation(\mathbf{P}_t, F)$, (3)
4:      $\mathbf{P}_C = Crossover(\mathbf{P}_t, \mathbf{P}_M, C_r)$, (4)
5:      $\mathbf{P}_t^{obj} = D(\mathbf{x}, \mathbf{P}_t)$, predicted objectness score of $\mathbf{P}_t$
6:      $\mathbf{P}_C^{obj} = D(\mathbf{x}, \mathbf{P}_C)$, predicted objectness score of $\mathbf{P}_C$
7:      for $i < N$ do
8:          $\mathbf{S}_t(i) = \mathbf{P}_t(i)^{obj} - S_0$
9:          $\mathbf{S}_C(i) = \mathbf{P}_C(i)^{obj} - S_0$
10:         if $length(\mathbf{S}_t(i)) < length(\mathbf{S}_C(i))$ then
11:             $\mathbf{P}_{t+1}(i) = \mathbf{P}_t(i)$
12:             $\mathbf{L}(i) = length(\mathbf{S}_t(i))$
13:         else
14:             $\mathbf{P}_{t+1}(i) = \mathbf{P}_C(i)$
15:             $\mathbf{L}(i) = length(\mathbf{S}_C(i))$
16:         end if
17:         if $\mathbf{L}(i) == 0$ then
18:             $flag = 1$, ← a successful attack
19:             $index = i$
20:             return $index, flag$ ← return the index of the best individual
21:         end if
22:     end for
23:     $t = t + 1$
24: end for
```

TABLE I
Distribution of Weather Factors and Parameter Settings

Factor	Variables	Range	Numeric type
Snow	snow_coeff	$[0, 0.3]$	float
Fog	fog_coeff	$[0, 0.3]$	float
Shadow	dimension	$[3, 8]$	int
	$[x_1, y_1, x_2, y_2]$	[0,1]	float
Sun flare	$[x_0, y_0]$	$[0, 1]$	float

A. Experimental Setups

1) Datasets: Our experimental evaluation was conducted using the DOTA-v1.0 dataset [28], which is a comprehensive dataset consisting of 2806 images, and 15 common categories captured from diverse sensors and platforms. The images in DOTA are of the size in different ranges, to ensure consistent evaluation, we divided all images into sub-images of 608 × 608 [28]. Consequently, there are 39 905 images

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

TABLE II
QUANTITATIVE EVALUATIONS AGAINST DIFFERENT MODELS FOR THE THREE GIVEN CATEGORIES

Categories	Models	YOLOv3		YOLOv4		Faster R-CNN		RetinaNet	
		ASR	AP	ASR	AP	ASR	AP	ASR	AP
Plane	L_∞	**81.51%**	**35.04%**	**73.44%**	**52.18%**	**67.03%**	**46.4%**	**55.56%**	**62.48%**
	L_2	75.68%	43.27%	63.49%	56.99%	51.28%	54.19%	39.15%	72.79%
	Random	35.27%	69.17%	13.69%	84.46%	19.78%	75.28%	12.17%	87.08%
Ship	L_∞	**65.91%**	**47.16%**	**62.15%**	**56.09%**	**64.61%**	**42.5%**	**67.39%**	**49.19%**
	L_2	62.11%	50.21%	59.81%	60.12%	59.83%	45.73%	63.47%	52.25%
	Random	38.71%	63.61%	32.47%	78.6%	34.74%	64.86%	29.99%	69.9%
Large-vehicle	L_∞	**79.91%**	**38.91%**	**71.56%**	**61.23%**	**77.22%**	**44.49%**	**83.96%**	58.15%
	L_2	75.18%	42.71%	63.08%	66.10%	72.10%	49.35%	83.46%	**55.42%**
	Random	40.61%	65.08%	28.36%	81.19%	34.82%	72.95%	59.05%	66.89%

in the training set and 13 603 in the validation set. In our experiments, we select the *Plane*, *Ship*, and *Large-vehicle* categories as our target classes since they are highly important in both civilian and military domains. For each category, we randomly sampled 100 images from the validation for each category to perform black-box attacks.

2) Victim Models: Four mainstream detectors are employed in our experiments, including YOLOv3 [47], YOLOv4 [48], Faster R-CNN [49], and RetinaNet [50]. The YOLOv3 and YOLOv4 are trained using the dataset collected in Section IV-A1. The mean average precision (mAP) measured at the intersection over union (IoU) threshold of 0.50 are 58.5 and 64.34, respectively, which are on par with the models trained on MS-COCO. For the Faster R-CNN[1] [51] and RetinaNet[2] [52], we cloned them from GitHub to conduct attacks. We only get the model's outputs without any modification to the model itself.

3) Metrics: The average precision (AP) is our first metric to evaluate the effectiveness of the proposed method, which can be used to evaluate each object category separately. In the context of attack scenarios, a lower AP indicates a better attack. In addition, we adopt the attack success rate [10] as a supplement, which is defined as the fraction of instances that are not correctly predicted by detectors after the attack. Intuitively, a more powerful attack will yield a higher ASR.

4) Implementation Details: We use the Automold[3] package, which is built on opencv,[4] to implement the $g(\mathbf{x}, \mathcal{E})$ of (2). To validate the effectiveness of our proposed method, we select four commonly encountered weather factors: *snow*, *fog*, *shadow*, and *sun flare*. The specific details of these weather factors are provided in Table I. The $[x_1, y_1, x_2, y_2]$ of *shadow* represent the rectangular constraint of the shadow area, and $[x_0, y_0]$ of *sun flare* indicate the center coordinates of the sun flare. For the case of L_∞ norm in (2), we independently perform the evolutionary selection process for each weather factor. For the case of L_2 norm setting, we employ a two-step scheme because we need to consider the numerical constraints between different factors. In the first step, we sample the population independently for each factor. Subsequently, we conducted a reject-accept process where only the samples satisfying the L_2 bound were retained for the next optimization. In this article, all experiments are conducted on a NVIDIA RTX 3090 24GB GPU.

B. Quantitative Results and Analysis

This section first verifies the effectiveness of the proposed method on different categories and victim models. The quantitative results are shown in Tables II, and some generated adversarial examples are listed in Fig. 3. In the table, *Random* refers to that we randomly sample individuals from each variable range without any optimization, serving as a comparative result. We can conclude from the results.

1) Our method achieves remarkable attack performance across all cases, demonstrating its effectiveness and generality.
2) Attacks under L_∞ constraints generally outperform those under L_2 constraints when setting the same ϵ. Revisiting the definitions of L_∞ and L_2 constraints in Section III-B, we can observe that, compared to the scenarios where individuals can be independently searched under the L_∞ constraint, there are mutual constraints among individuals under the L_2 constraint. Consequently, the L_∞ constraint provides a larger search space, resulting in more efficient attacks.
3) The attack results against different detectors exhibit inconsistent properties across these three categories. For example, the ASR gap between attacking against YOLOv3 and RetinaNt can reach 25.95% when targeting the *Plane* category, whereas there is no significant difference in the other two categories.

In order to further illustrate the effectiveness of our proposed method, we choose two classic methods, namely RAP [13] and DAG [14], for comparison. For simplicity, all experiments are only conducted against YOLOv3. The results are shown in Table III, and some examples are illustrated in Fig. 1. From the perspective of ASR, our method outperforms both RAP and DAG on both *Plane* and *Large-vehicle* categories. Although its attack effectiveness is lower than DAG on the *Ship* category, it still surpasses RAP. Particularly, our method achieves a significant improvement of 50.69% over DAG on the *Plane* category. From the perspective of visualization, although all perturbations are concentrated around the targets, those generated by RAP and DAG are more pronounced compared to the weather-based perturbations we

[1] https://github.com/dingjiansw101/AerialDetection
[2] https://github.com/csuhan/s2anet
[3] https://github.com/UjjwalSaxena/Automold–Road-Augmentation-Library
[4] https://github.com/opencv/opencv

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

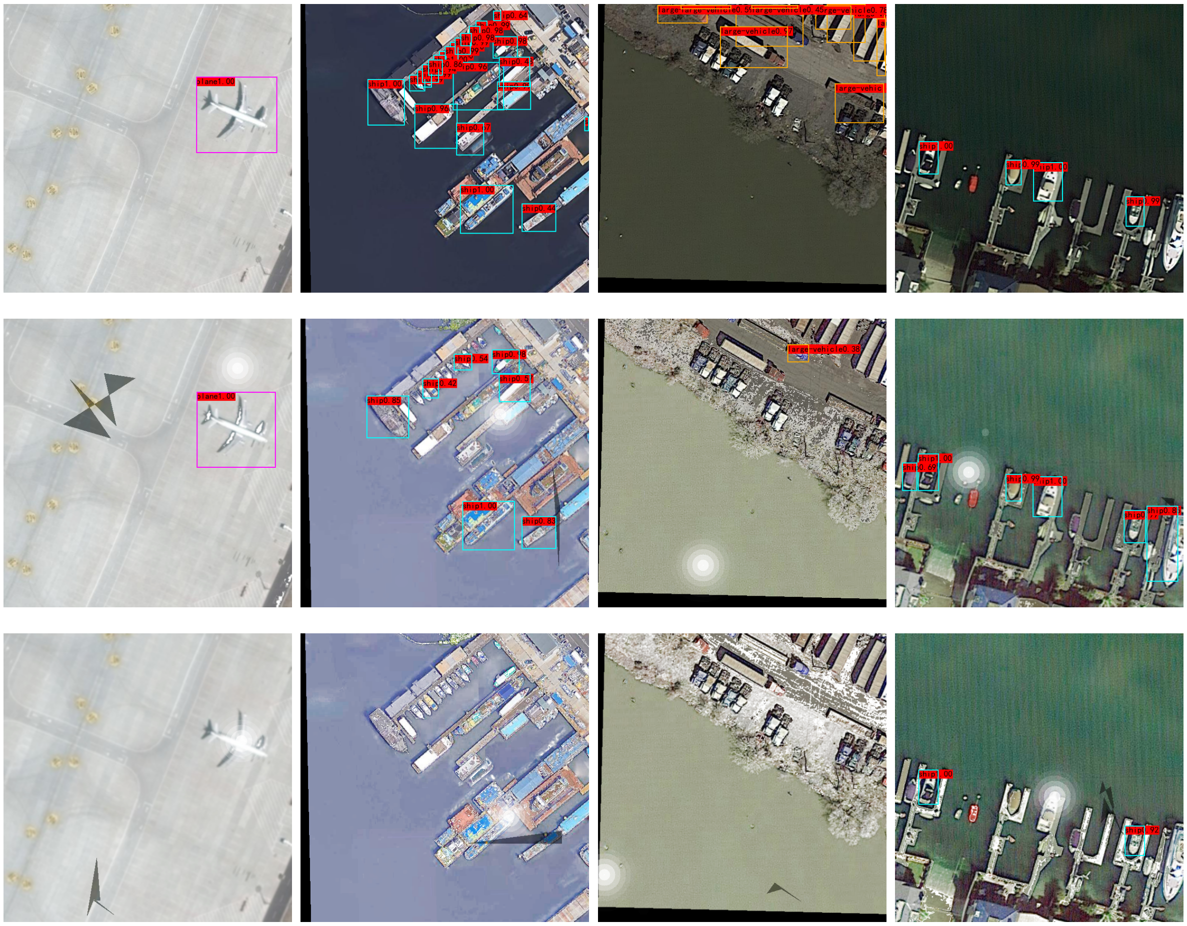


Fig. 3. (Top) Visualization of detection results for clean, (middle) randomly added weather noise and (bottom) corresponding adversarial examples. While the added weather-style adversarial perturbations corrupt the images to some extent, they remain inconspicuous to humans due to their natural properties.

TABLE III

COMPARISON WITH THE STATE-OF-THE-ART ATTACK METHOD

	Plane		Ship		Large-vehicle	
	ASR	AP	ASR	AP	ASR	AP
Clean	0.0%	100.0%	0.0%	100.0%	0.0%	100.0%
RAP [13]	76.02%	**22.15%**	60.37%	43.85%	54.60%	48.67%
DAG [14]	30.82%	61.19%	**77.62%**	**26.47%**	66.69%	**32.97%**
Ours(L_∞)	**81.51%**	35.04%	65.71%	47.16%	**79.81%**	38.91%

TABLE IV

EVALUATION RESULTS VERSUS DIFFERENT WEATHER FACTORS AGAINST *PLANE* CATEGORY

Plane	YOLOv3		YOLOv4		Faster R-CNN		RetinaNet	
	ASR	AP	ASR	AP	ASR	AP	ASR	AP
Snow	72.6%	40.28%	60.17%	57.09%	55.68%	57.22%	33.86%	77.31%
Fog	72.6%	38.62%	1.66%	99.93%	16.48%	97.45%	6.35%	98.22%
Sun flare	12.67%	93.14%	12.03%	95.48%	19.41%	96.41%	8.47%	96.53%
Shadow	9.25%	95.66%	5.39%	98.12%	20.15%	96.82%	7.94%	98.93%
Hybrid	**81.51%**	**35.04%**	**73.44%**	**52.18%**	**67.03%**	**46.4%**	**55.56%**	**62.48%**

employ. Additionally, both RAP and DAG methods exhibit significant clustering of perturbations, resulting in a less effective stealth effect. Furthermore, it is important to note that RAP and DAG are executed in white-box settings, utilizing model's gradients to update adversarial perturbations. In contrast, our method operates in black-box settings, yet achieves superior attack performance, highlighting the effectiveness and advancement of our method.

C. Ablation Study

After jointly optimizing all weather factors to generate adversarial perturbations in Section IV-B, this section focuses on studying the impact of optimizing a single weather factor. The quantitative results are presented in Tables IV–VI. The tables show that while optimizing a single weather factor can still generate adversarial perturbations, their effectiveness varies significantly. Notably, the *snow* factor appears to be

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

TABLE V
Evaluation Results Versus Different Weather Factors Against *Ship* Category

Ship	YOLOv3		YOLOv4		Faster R-CNN		RetinaNet	
	ASR	AP	ASR	AP	ASR	AP	ASR	AP
Snow	54.41%	59.18%	42.79%	72.55%	45.13%	59.88%	46.02%	63.58%
Fog	17.76%	85.96%	20.00%	91.91%	21.97%	88.96%	21.59%	83.66%
Sun flare	14.27%	89.91%	13.55%	93.05%	18.29%	89.89%	14.07%	90.44%
Shadow	9.34%	94.15%	9.78%	95.29%	10.75%	94.83%	10.8%	92.02%
Hybrid	**65.91%**	**47.16%**	**62.15%**	**56.09%**	**64.61%**	**42.5%**	**67.39%**	**49.19%**

TABLE VI
Evaluation Results Versus Different Weather Factors Against *Large-Vehicle* Category

Large-vehicle	YOLOv3		YOLOv4		Faster R-CNN		RetinaNet	
	ASR	AP	ASR	AP	ASR	AP	ASR	AP
Snow	64.41%	50.56%	49.25%	72.21%	67.65%	56.17%	70.67%	66.36%
Fog	21.13%	85.16%	20.27%	88.61%	16.86%	93.41%	59.15%	76.67%
Sun flare	10.65%	94.09%	8.48%	95.6%	12.87%	94.90%	13.09%	94.13%
Shadow	3.14%	98.12%	2.12%	98.71%	6.74%	98.48%	14.96%	94.33%
Hybrid	**79.81%**	**38.91%**	**71.56%**	**61.23%**	**77.22%**	**44.49%**	**83.96%**	**58.15%**

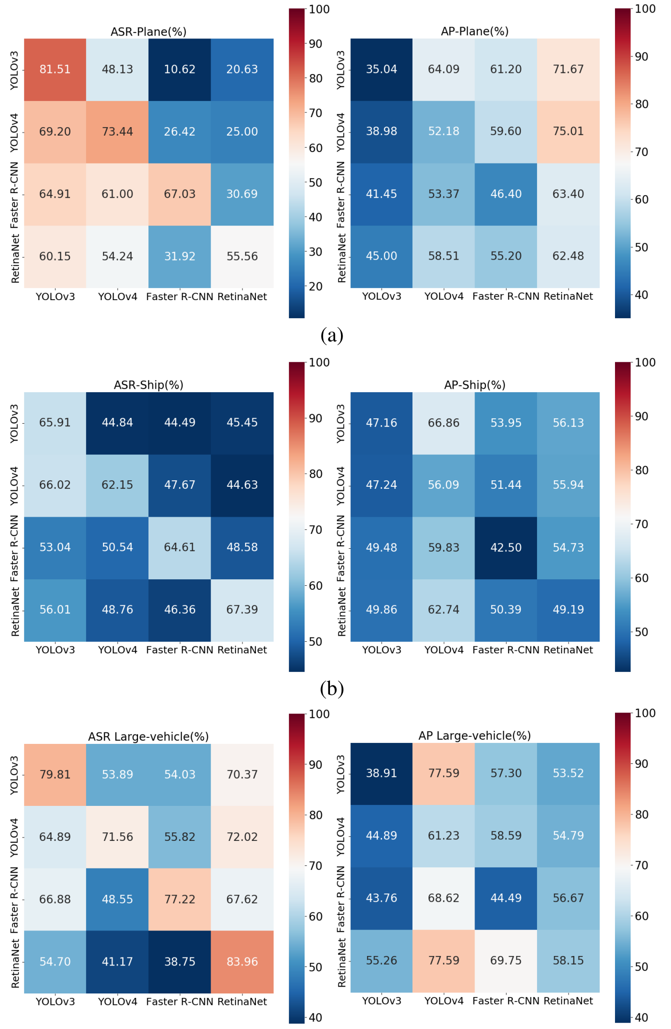
the most influential, while the *shadow* factor has the least impact. This could be attributed to the fact that when optimizing the *snow* factor, it perturbs the whole image, whereas the generated shadows tend to be concentrated in localized regions, thereby limiting its attack effectiveness. To fully exploit the advantages of each weather factor and achieve the best adversarial attack effect, it is recommended to optimize all weather factors simultaneously when applying the method outlined in this chapter.

D. Transfer Attacks

This section conducts transfer attacks across various models to comprehensively evaluate the robustness of adversarial examples crafted by our methods. Specifically, we employ the adversarial examples crafted against the source models to attack the target models directly. The assessment outcomes are illustrated in Fig. 4. In each subgraph, the models on the left of each row represent the source model used to generate adversarial samples, while the target models represented as transfer attacks are identified below each column.

1) Fig. 4(a) reveals that for the *Plane* category, the adversarial examples generated against YOLOv3 and YOLOv4 have limited attack transferability on Faster R-CNN and RetinaNet. Conversely, the adversarial examples crafted against Faster R-CNN and RetinaNet demonstrate strong transferability to YOLOv3 and YOLOv4. This discrepancy may be attributed to differences in the model architectures. Generally, Faster R-CNN and RetinaNet possess more complex architectures, which make the generated adversarial examples more effective.

2) Fig. 4(b) and (c) indicate that the *Ship* and *Large-vehicle* exhibit distinct characteristics compared to the *Plane*. Notably, there is no noticeable difference in the transferability effects among models, as adversarial examples generated based on source models can show superior performance on target models. This observation can be attributed to the fact that from an overhead perspective, ships and large vehicles often exhibit simple geometric shapes (rectangles), making the targets themselves more vulnerable to adversarial perturbations.



(a)
(b)
(c)

Fig. 4. Attack transferability across different models against variant categories. (a) *Plane*. (b) *Ship*. (c) *Large-vehicle*.

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

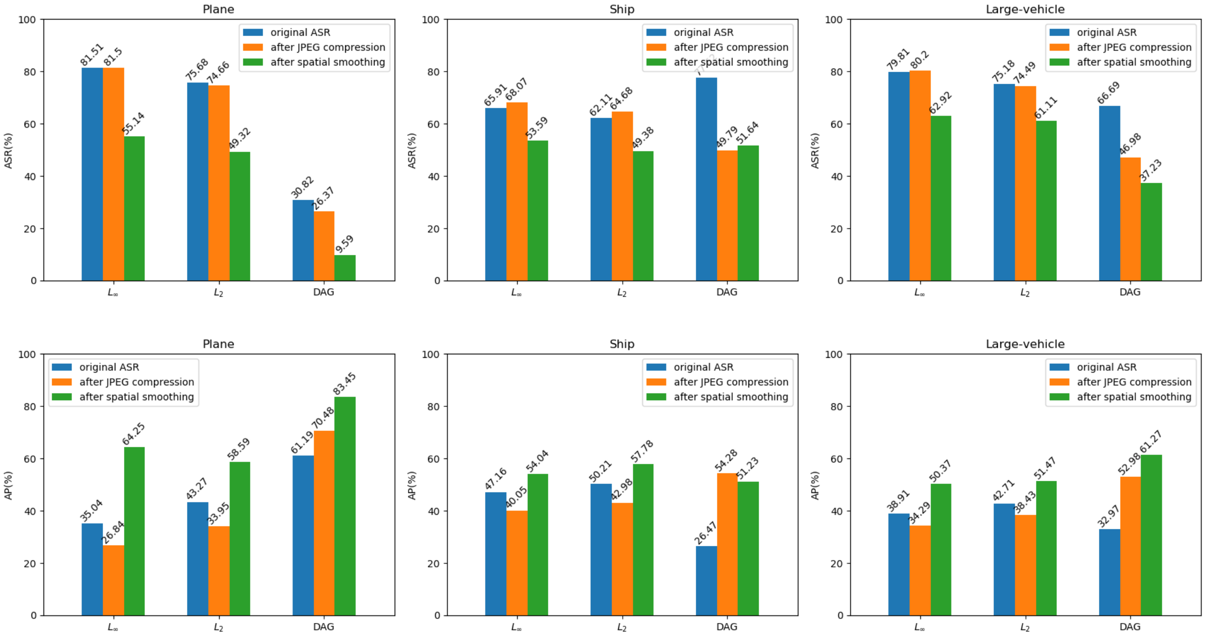


Fig. 5. Attack performance before and after defenses. In these cases, the gap before and after attacks is the key metric for evaluating the robustness of methods. The smaller the drop, the more robust the method.

3) In general, transfer attacks constitute another form of black-box attacks (as illustrated in Section II-B). In existing works targeting optical aerial detectors, transfer attacks generally yield inferior performance compared to white-box attacks [35], [53] (the value of off-diagonal elements in tables will be much smaller than that of diagonal elements). However, the proposed method in this article is essentially a query-based black-box attack technique. Since we do not heavily rely on the internal information of the model, such as gradients, the generated adversarial examples typically show better universality and generalization.

E. Attack Performance Under Defense

In addition to examining the transferability of adversarial examples across models, assessing whether the generated adversarial samples remain effective under defense methods is another important indicator for evaluating their adversarial robustness. This section employs two common defense methods, namely JPEG compression [54] and spatial smoothing [55], to conduct experiments. The results are shown in Fig. 5, where we specifically discuss the performance of adversarial examples generated under both L_∞ and L_2 constraints, as well as the adversarial examples crafted by DAG. For simplicity, all experiments are conducted against YOLOv3. From the results, we can conclude as follows.

1) JPEG compression has no defensive effect on the adversarial examples generated by our method. In fact, it even improves the attack efficiency for the *Ship* category. This can be attributed to the fact that our method goes beyond the simple addition of adversarial noise, rendering JPEG compression ineffective in compressing the adversarial perturbations. Conversely, JPEG compression demonstrates a notable effect on DAG.
2) Unlike JPEG compression, although spatial smoothing demonstrates defensive effects against our method and DAG, our approach exhibits greater robustness compared to DAG in this context. Additionally, the effectiveness of spatial smoothing varies across different categories. For instance, the ASR reduction for the *Plane* category can reach up to 26.37%, whereas for *Ship* and *Large-vehicle*, the maximum drops are only 12.32% and 16.89%, respectively.

F. DOTA-W

Since our method is modeled on black-box settings, the generated adversarial examples do not rely heavily on the target models' private information. Furthermore, the above experiments have verified the effectiveness and generalization of our method, enabling us to synthesize a universal dataset to evade optical aerial detectors. A universal adversarial dataset is of great significance for evaluating and improving the robustness of detectors. Because the original images come from the DOTA validation set, and the adversarial images are perturbed by weather factors, we hence name the dataset DOTA-W. To create DOTA-W, we randomly sample 1000 images from the validation set, considering 13 606 images available (as described in Section IV-A1). We would like to note that our code has been made publicly available, and we encourage researchers to generate additional adversarial samples based on our code.

Our target model for crafting adversarial examples in DOTA-W is YOLOv3, and the parameters used align with the cases involving the L_∞ constraint discussed in Section IV-B. These parameters provide a worst case attack scenario, hence more sufficient to evaluate the robustness of

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

models. Researchers can refer to the GitHub repository at https://github.com/tang-agui/attADs-AWP for more adversarial examples generated using our method.

V. Conclusion

This article proposed a novel adversarial attack method utilizing weather factors, resulting in more natural and stealthy adversarial examples. Given the absence of internal model information in real-world scenarios, this article formulated the optimization problem of generating adversarial examples in black-box settings. To address this challenge, we adopted the DE algorithm to solve the optimization problem effectively. We demonstrated the effectiveness of our proposed method across various detectors and categories under both L_∞ and L_2 constraints. Furthermore, we have shown the universality and generalization of our approach through extensive transfer attacks across different models and evaluated the robustness of the generated adversarial examples against two commonly used defense methods. However, this article has certain limitations. First, while using weather factors to generate adversarial perturbations yields excellent stealth effects, the perturbations are applied globally, preventing us from controlling their specific locations. As a consequence, the generated perturbations might not be physically realizable. Second, we introduced a dataset for robustness evaluation in Section IV-F, but we have not yet conducted corresponding model training based on this dataset. Therefore, in future work, we intend to further enhance our approach from these two branches.

References

[1] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5607514.

[2] C. Li, G. Cheng, G. Wang, P. Zhou, and J. Han, "Instance-aware distillation for efficient object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5602011.

[3] C. Zhang, W. Jiang, Y. Zhang, W. Wang, Q. Zhao, and C. Wang, "Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4408820.

[4] X. X. Zhu et al., "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

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

[6] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–11.

[7] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–28.

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

[9] C. Li, H. Wang, J. Zhang, W. Yao, and T. Jiang, "An approximated gradient sign method using differential evolution for black-box adversarial attack," *IEEE Trans. Evol. Comput.*, vol. 26, no. 5, pp. 976–990, Oct. 2022.

[10] K. Eykholt et al., "Physical adversarial examples for object detectors," in *Proc. USENIX Workshop Offensive Technol. (WOOT)*, 2018, pp. 1–10.

[11] Y. Wang, K. Wang, Z. Zhu, and F.-Y. Wang, "Adversarial attacks on faster R-CNN object detector," *Neurocomputing*, vol. 382, pp. 87–95, Mar. 2020.

[12] Y. Zhao, H. Zhu, R. Liang, Q. Shen, S. Zhang, and K. Chen, "Seeing isn't believing: Towards more robust adversarial attack against real world object detectors," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2019.

[13] Y. Li, D. Tian, M.-C. Chang, X. Bian, and S. Lyu, "Robust adversarial perturbation on deep proposal-based models," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2018, pp. 1–11.

[14] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, "Adversarial examples for semantic segmentation and object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1378–1387.

[15] X. Sun, G. Cheng, L. Pei, H. Li, and J. Han, "Threatening patch attacks on object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5609210.

[16] Y.-C.-T. Hu, J.-C. Chen, B.-H. Kung, K.-L. Hua, and D. S. Tan, "Naturalistic physical adversarial patch for object detectors," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7828–7837.

[17] Z. Hu, W. Chu, X. Zhu, H. Zhang, B. Zhang, and X. Hu, "Physically realizable natural-looking clothing textures evade person detectors via 3D modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 16975–16984.

[18] T. Zhang, Y. Xiao, X. Zhang, H. Li, and L. Wang, "Benchmarking the physical-world adversarial robustness of vehicle detection," 2023, *arXiv:2304.05098*.

[19] Y. Dong et al., "Benchmarking robustness of 3D object detection to common corruptions in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 1022–1032.

[20] Z. Zhu et al., "Understanding the robustness of 3D object detection with Bird's-Eye-View representations in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 21600–21610.

[21] L. Zhai et al., "Adversarial rain attack and defensive deraining for DNN perception," 2020, *arXiv:2009.09205*.

[22] S. Mei, J. Lian, X. Wang, Y. Su, M. Ma, and L.-P. Chau, "A comprehensive study on the robustness of image classification and object detection in remote sensing: Surveying and benchmarking," 2023, *arXiv:2306.12111*.

[23] J. Lian, S. Mei, S. Zhang, and M. Ma, "Benchmarking adversarial patch against aerial detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5634616.

[24] H. Li, H. Li, H. Zhang, and W. Yuan, "Black-box attack against handwritten signature verification with region-restricted adversarial perturbations," *Pattern Recognit.*, vol. 111, Mar. 2021, Art. no. 107689.

[25] X. Sun, G. Cheng, L. Pei, and J. Han, "Query-efficient decision-based attack via sampling distribution reshaping," *Pattern Recognit.*, vol. 129, Sep. 2022, Art. no. 108728.

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

[27] L. Han, H. Wang, and S. Wang, "A surrogate-assisted evolutionary algorithm for space component thermal layout optimization," *Space, Sci. Technol.*, vol. 2022, Jan. 2022.

[28] G.-S. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

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

[30] H. Huang, Y. Wang, Z. Chen, Z. Tang, W. Zhang, and K.-K. Ma, "RPATTACK: Refined patch attack on general object detectors," 2021, *arXiv:2103.12469*.

[31] S.-T. Chen, C. Cornelius, J. Martin, and D. H. Chau, "Robust physical adversarial attack on faster R-CNN object detector," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases (ECML PKDD)*, 2018, pp. 52–68.

[32] R. den Hollander et al., "Adversarial patch camouflage against aerial detection," *Proc. SPIE*, vol. 11543, pp. 77–86, Sep. 2020.

[33] J. Lian, X. Wang, Y. Su, M. Ma, and S. Mei, "CBA: Contextual background attack against optical aerial detection in the physical world," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5606616.

[34] J. Lian, X. Wang, Y. Su, M. Ma, and S. Mei, "Contextual adversarial attack against aerial detection in the physical world," 2023, *arXiv:2302.13487*.

[35] G. Tang, T. Jiang, W. Zhou, C. Li, W. Yao, and Y. Zhao, "Adversarial patch attacks against aerial imagery object detectors," *Neurocomputing*, vol. 537, pp. 128–140, Jun. 2023.

[36] C. Li, W. Yao, H. Wang, and T. Jiang, "Adaptive momentum variance for attention-guided sparse adversarial attacks," *Pattern Recognit.*, vol. 133, Jan. 2023, Art. no. 108979.

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.

[37] H. Huang, Z. Chen, H. Chen, Y. Wang, and K. Zhang, “T-SEA: Transfer-based self-ensemble attack on object detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 20514–20523.

[38] Y. Dong et al., “Boosting adversarial attacks with momentum,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9185–9193.

[39] S. Chen, F. He, X. Huang, and K. Zhang, “Relevance attack on detectors,” *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108491.

[40] Y. Zhang et al., “Boosting transferability of physical attack against detectors by redistributing separable attention,” *Pattern Recognit.*, vol. 138, Jun. 2023, Art. no. 109435.

[41] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, “ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models,” in *Proc. 10th ACM Workshop Artif. Intell. Secur.*, 2017, pp. 1–13.

[42] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, “Black-box adversarial attacks with limited queries and information,” in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2137–2146.

[43] C. Wu, W. Luo, N. Zhou, P. Xu, and T. Zhu, “Genetic algorithm with multiple fitness functions for generating adversarial examples,” in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2021, pp. 1792–1799.

[44] A. Ghosh, S. S. Mullick, S. Datta, S. Das, A. K. Das, and R. Mallipeddi, “A black-box adversarial attack strategy with adjustable sparsity and generalizability for deep image classifiers,” *Pattern Recognit.*, vol. 122, Feb. 2022, Art. no. 108279.

[45] X. Wei, Y. Guo, and B. Li, “Black-box adversarial attacks by manipulating image attributes,” *Inf. Sci.*, vol. 550, pp. 285–296, Jun. 2021.

[46] R. Storn and K. Price, “Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces,” *J. global Optim.*, vol. 11, no. 4, p. 341, 1997.

[47] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, *arXiv:1804.02767*.

[48] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOV4: YOLOV4: Optimal speed and accuracy of object detection,” 2020, *arXiv:2004.10934*.

[49] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[50] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[51] J. Ding et al., “Object detection in aerial images: A large-scale benchmark and challenges,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7778–7796, Nov. 2022.

[52] J. Han, J. Ding, J. Li, and G.-S. Xia, “Align deep features for oriented object detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5602511.

[53] M. Lu, Q. Li, L. Chen, and H. Li, “Scale-adaptive adversarial patch attack for remote sensing image aircraft detection,” *Remote Sens.*, vol. 13, no. 20, p. 4078, Oct. 2021.

[54] G. K. Dziugaite, Z. Ghahramani, and D. M. Roy, “A study of the effect of JPG compression on adversarial images,” 2016, *arXiv:1608.00853*.

[55] W. Xu, D. Evans, and Y. Qi, “Feature squeezing: Detecting adversarial examples in deep neural networks,” 2017, *arXiv:1704.01155*.

Guijian Tang received the M.Sc. degree in aerospace science and technology from the National University of Defense Technology, Changsha, China, in 2017. He is currently pursuing the Ph.D. degree in aeronautical and astronautical science and technology with the College of Aerospace science and Engineering, National University of Defense Technology.

His research interests include adversarial machine learning, evolutionary optimization, and image processing.

Wen Yao received the M.Sc. and Ph.D. degrees in aerospace engineering from the National University of Defense Technology, Changsha, China, in 2007 and 2011, respectively. She is currently a Professor with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing, China. Her research interests include spacecraft systems engineering, multidisciplinary design optimization, uncertainty-based optimization, and data-driven surrogate modeling and evolutionary optimization.

Tingsong Jiang received the B.Sc. and Ph.D. degrees from the School of Electronics Engineering and Computer Science, Peking University, Beijing, China, in 2010 and 2017, respectively. He is currently an Assistant Professor with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing. His research interests include adversarial machine learning, AI safety, and knowledge graph.

Weien Zhou received the B.Sc. degree in mathematics from Nanjing University, Nanjing, China, in 2012, and the Ph.D. degree in computational mathematics from the National University of Defense Technology, Changsha, China, in 2017. He is currently an Assistant Professor with the Unmanned Systems Research Center, National Innovation Institute of Defense Technology. His research interests include adversarial machine learning, uncertainty quantification, and large-scale optimization.

Yang Yang received the M.S. degree in electronic engineering from Xidian University, Xi’an, China, in 2011, and the Ph.D. degree in computer science from Peking University, Beijing, China, in 2017. He is currently a Research Professor with the Chinese Academy of Military Science, Beijing. His research interests include machine learning and pattern recognition.

Donghua Wang received the M.Sc. degree from Ningbo University, Ningbo, China, in 2021. He is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China. He is also with the Defense Innovation Institute, Chinese Academy of Military Science, Beijing, China. His research interests include adversarial machine learning, AI safety, and image processing.

Authorized licensed use limited to: Peking University. Downloaded on October 03,2024 at 14:50:42 UTC from IEEE Xplore. Restrictions apply.