Journal of Membrane Computing (2024) 6:130–147
https://doi.org/10.1007/s41965-024-00142-3

REVIEW PAPER

Adversarial attacks in computer vision: a survey

Chao Li[1,2] · Handing Wang[1] · Wen Yao[2] · Tingsong Jiang[2]

Received: 12 December 2023 / Accepted: 10 March 2024 / Published online: 10 April 2024

Abstract
Deep learning, as an important topic of artificial intelligence, has been widely applied in various fields, especially in computer vision applications, such as image classification and object detection, which have made remarkable advancements. However, it has been demonstrated that deep neural networks (DNNs) suffer from adversarial vulnerability. For the image classification task, the carefully crafted perturbations are added to the clean images, and the resulting adversarial examples are able to change the prediction results of DNNs. Hence, the presence of adversarial examples presents a significant obstacle to the security of DNNs in practical applications, which has garnered considerable attention from researchers in related fields. Recently, a number of studies have been conducted on adversarial attacks. In this survey, the relevant concepts and background are first introduced. Then, based on computer vision tasks, we systematically review the existing adversarial attack methods and research progress. Finally, several common defense methods are summarized, and some challenges are discussed.

Keywords Deep learning · Computer vision · Adversarial attacks · Adversarial examples

1 Introduction

Deep neural networks (DNNs) have made great progress in various computer vision tasks [1, 2]. However, recent research has demonstrated the susceptibility of DNNs to adversarial examples [3, 4], where some imperceptible perturbations are added to a clean image, the generated perturbed image may change the classification result of the DNNs [5]. More seriously, this phenomenon also exists in the real world [6–8]. For example, some researchers attempt to fool the recognition system of the autonomous vehicle by putting some black and white stickers on specific locations of traffic signs, in which the "STOP" sign is misrecognized as "Limit 45", resulting in a serious traffic accident [9]. The above phenomenon raises concerns regarding the safety of artificial intelligence (AI) applications in the physical world, and the process of making DNNs output wrong prediction results by constructing perturbed images is called adversarial attacks [10]. Recently, adversarial attacks have become a research hotspot in AI field. Since 2018, there has been an exponential increase in the number of papers published each year on adversarial attacks in conferences and journals [11]. Therefore, it is necessary to systematically review the adversarial attacks and understand its developments. Although several surveys on adversarial attacks have been proposed [11, 12], they suffer from the following drawbacks: first, the reviewed literature is stale. With the rapid development in the field of adversarial attacks, the latest advances should be investigated. Second, they only investigated the work on a single visual task, mostly for classification and detection tasks, and their coverage is narrow. Currently, adversarial attacks have been used in a variety of visual tasks. Finally, some surveys only review related work towards attack or defence. In fact, adversarial attack and defence are complementary and both should be introduced simultaneously. Therefore, to address the above issues, based on a variety of computer vision tasks, this work comprehensively investigates the recent advances in adversarial attack and defence. Specifically, the concept and classification of adversarial attacks are first introduced. Then, the existing mainstream adversarial attack methods are summarized for the classification, detection and other computer vision tasks, respectively. In addition, several

✉ Handing Wang
hdwang@xidian.edu.cn

✉ Wen Yao
wendy0782@126.com

[1] School of Artificial Intelligence, Xidian University, Xi'an 710071, Shaanxi, China

[2] Defense Innovation Institute, Chinese Academy of Military Science, Beijing 100071, China

commonly used adversarial defense methods are reviewed. Finally, some challenges and promising research directions are proposed.

1.1 Definition of adversarial attacks

As depicted in Fig. 1, for a original image, a trained DNN predicts the label of the image as "Panda"; however, after adding the adversarial perturbations, the same DNN again predicts the crafted adversarial example, which predicts the label as "Gibbon". It is worth noting that these added perturbations are imperceptible to humans, making the produced adversarial examples visually indistinguishable from the original sample. Hence, adversarial attacks can be viewed as the process of producing adversarial examples that successfully fool DNNs, and the essence is how to design an algorithm to generate adversarial perturbations. Normally, the expression of adversarial attacks is as follows:

$$\begin{aligned} &\arg\max_{\Delta x} P_{\text{adv}}(x_{\text{adv}}), \\ &x_{\text{adv}} = x + \Delta x, \\ &\text{s.t.} \|\Delta x\| \leq \varepsilon, \end{aligned} \tag{1}$$

where P_{adv} is the probability that the adversarial example x_{adv} is predicted by the DNN as adversarial label *adv*, and x represents the clean image. Δx denotes adversarial perturbation, where its magnitude is bounded by the threshold ε, and $\|\cdot\|$ is used to evaluate the intensity of Δx. In general, $\|\cdot\|$ can be l_0, l_1, l_2, or l_∞ norm.

1.2 Classification of adversarial attacks

Based on the attack scenarios, adversarial attacks can generally be categorized into white-box [14] and black-box attacks [15]. The former means that the attacker can obtain and use the internal model information like model weights and training data. Therefore, white-box attacks have relaxed constraints, and it is easier to obtain better attack performance. In contrast, The latter implies that the attacker can not obtain any information about the model during the process of generating adversarial examples, and only model outputs can be exploited. In real-world scenarios, obtaining internal information about targeted systems is impossible. Thus, black-box attacks are more applicable.

Based on the different objectives, adversarial attacks can further be classified into targeted [16] and non-targeted attacks [17]. Non-targeted attacks imply that the adversarial examples only need to make the prediction result of DNNs change, while targeted attacks require successful classification of adversarial examples into a predetermined class. Thus, achieving targeted attacks is considerably more challenging than non-targeted ones.

1.3 Common data sets

For image classification task, the common image data sets in adversarial attacks include MNIST [18], CIFAR-10 [19], CIFAR-100 [19] and ImageNet [20]. Table 1 presents the basic information of each data set. The MNIST data set comprises handwritten digits from 0 to 9, with a training set consisting of 60,000 samples and a testing set containing 10,000 samples. Different from other data sets, the images in the MINIST data set are single-channel grey scale images. Both CIFAR-10 and

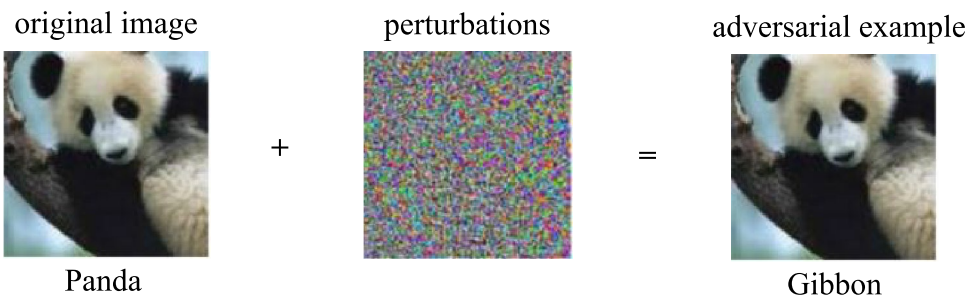


Fig. 1 Demonstration of adversarial attacks

Table 1 Information of the common data sets in classification tasks

Data sets	Label number	Training set	Testing set	Validation set	Size
MNISIT	10	60,000	10,000	–	28×28
CIFAR-10	10	50,000	10,000	–	32×32
CIFAR-100	10	50,000	10,000	–	32×32
ImageNet	1000	1,200,000	100,000	50,000	224×224

CIFAR-100 data sets are similar in terms of their structure, where the training sets consist of 50,000 images while the test sets contain 10,000 images. The only distinction is that CIFAR-10 has ten image labels whereas CIFAR-100 has one hundred labels. In comparison to the aforementioned three data sets, ImageNet stands out due to its vast number of high-quality images and diverse categories.

For object detection task, the commonly used image data sets include COCO [21] and VOC [22], and their basic information is shown in Table 2. The COCO data set is usually used for object detection, semantic segmentation, and other computer vision tasks, containing 80 target categories. COCO is divided into two versions: COCO2014 and COCO2017, where COCO2017 is more commonly used, and the training set comprises 118,287 images, while the validation set consists of 5000 images, and the test set contains 40,670 images. There are eight versions of VOC data set, VOC2005-VOC2012, and since the number of images and categories included in the data set before VOC2007 are less, the popular data sets at present are VOC2007 and VOC2012. For example, VOC2007 contains 20 categories, the training set and validation set include 2501 and 2510 images, respectively, and the test set includes 4952 images.

1.4 Performance indicators

In adversarial attacks, the performance of a method is commonly assessed by considering metrics such as attack success rate (ASR), fooling rate (FR), and Lp norm of perturbations. The details of the calculations are as follows:

$$\mathrm{ASR} = \frac{N_{\mathrm{s}}}{N_{\mathrm{true}}}. \tag{2}$$

The ASR can be calculated using Eq. (2), where N_{true} is the number of examples correctly classified by the model, and N_{s} represents the number of adversarial examples with successful attacks in N_{true}. ASR is the most intuitive indicator to measure the performance of an attack method. Larger ASR values are desirable:

$$\mathrm{FR} = \frac{N_{\mathrm{d}}}{N_{\mathrm{total}}}. \tag{3}$$

The FR can be computed by Eq. (3), where N_{total} is the number of examples input to the model, and N_d represents the number of adversarial examples that make the classification result of the model change. Unlike ASR, FR does not require the model to classify input examples correctly. Larger FR values are satisfactory:

$$L_p = \|\Delta x\|_p = \sqrt[p]{\sum_{i=1}^{n} \Delta x_i^p}, \Delta x = \left(\Delta x_1, \Delta x_1, ..., \Delta x_n\right), \tag{4}$$

The L_p norm of perturbations is described in Eq. (4), where Δx is the perturbation matrix and n is the number of perturbed pixels. In general, L_2 or L_∞ norm is commonly used to evaluate the intensity of the added perturbation. The use of a smaller L_p norm indicates better performance of the method.

2 Image classification-based adversarial attacks

Based on the image classification task, this section describes representative adversarial attack methods, including full pixel attacks, sparse attacks, universal adversarial attacks and style transfer attacks, respectively.

2.1 Full pixel attacks

For clarity, we further divide the full-pixel attack methods into white-box and black-box attacks.

2.1.1 White-box attacks

Fast gradient sign method (FGSM) FGSM [13] is a classical white-box attack method, which pointed out that due to the nonlinearity of DNNs, even a small perturbation added to the input is enough to mislead the classification results of DNNs. FGSM can be expressed as follows:

$$x_{\mathrm{adv}} = x + \varepsilon \times sign\left(\nabla_x J(\theta, x, y)\right), \tag{5}$$

where ε is the perturbation size, and $sign()$ involves the sign function. J denotes the cross-entropy loss function and θ represents the weights of the DNNs. y is the true label of the input image x. FGSM first calculates the gradient of the input image by the loss function, and the sign function is used to obtain the attack direction. Then, the gradient ascent method is employed to add perturbation to the image, resulting in an increase in the loss between the prediction labels and the true labels. Note that FGSM adds the same perturbation size on each pixel, and the method only needs one

Table 2 Information of the common data sets in detection tasks

Data sets	Label number	Training set	Testing set	Validation set	Versions
COCO	80	118,287	40,670	5000	COCO2017
VOC	20	2501	4952	2510	VOC2007

iteration to complete the attack. Thus, FGSM is an efficient white-box attack method.

Basic iterative method (BIM) Due to the nonlinear characteristics of the model, the gradient may change drastically in a narrow range. When FGSM only performs one iteration, the added perturbation amplitude may be too large, so that the crafted adversarial examples cannot perform a successful attack. Therefore, an iterative version of FGSM is proposed [23], as shown in Eq. (6):

$$x_{\text{adv}}^{t+1} = x_{\text{adv}}^{t} + \alpha \times \text{sign}\Big(\nabla_{x_{\text{adv}}^{t}} J\big(\theta, x_{\text{adv}}^{t}, y\big) \Big), \tag{6}$$

where α is the size of the perturbation added in each iteration. Compared to FGSM, BIM searches in a smaller step in each iteration, and experimental results illustrate that BIM is more competitive than FGSM.

Project gradient descent (PGD) PGD [24] is regarded as a variant of BIM. Compared with BIM, PGD firstly increases the number of iterations. Before the iterative attack, PGD initializes a perturbation to add to the original image. Then, the size of the perturbation added at each iteration is user-defined as a hyper-parameter, which is not correlated with the iterations. After each iteration, the added perturbations are constrained to be within the set perturbation threshold by projection. PGD is considered as the strongest white-box method, and it is often used to assess the model robustness.

Deepfool Moosav-Dezfooli et al. [25] proposed Deepfool attack, where the proposed method obtains a smaller perturbation by approximating the distance from the example to the nearest decision boundary, and this distance is also used to quantify the robustness of the model. Specifically, Deepfool pushes the original image towards the decision boundary by adding perturbation iteratively, and once the decision boundary is crossed, the previous perturbations are summed and added to the original examples to form the adversarial examples.

Carlini and Wagner attack (C &W) C &W [26] is an optimization-based white-box attack method, where the optimization objective function is as follows:

$$\min \|\delta\|_p + r \times F(x+\delta), \text{s.t.} x + \delta \in [0,1]^m, \tag{7}$$

where r is a hyperparameter, and m denotes the image channel dimension. δ represents the intensity of perturbation. F is defined as follows:

$$F(x+\delta) = \max(\max\{Z(x+\delta)_i : i \neq t\} - Z(x+\delta), -k), \tag{8}$$

where $Z(x)$ is the probability output of the classifier and the hyperparameter k constrains the confidence that the adversarial examples are misclassified as label t. In addition, the C &W attack introduces a new optimization variable w to deal with the constraints of Eq. (7). As shown in Eq. (9), the optimization error caused by the constraints in Eq. (7) is avoided by function mapping:

$$\delta = \frac{1}{2}(\tanh(w) + 1) - x, \tag{9}$$

The C &W attack demonstrates that the generated adversarial examples are able to achieve a high attack success rate even in the face of defensive distillation methods, but the optimization process is time-consuming, since it involves finding suitable hyperparameters.

2.1.2 Black-box attacks

Unlike white-box attacks, black-box attacks do not have access to the relevant knowledge of the model, and only the output of the model can be exploited. Therefore, it is relatively difficult to perform black-box attacks compared to white-box attacks. In general, black-box attacks can be divided into two types, namely transfer-based and query-based attacks. For both types of attacks, this section first describes the basic design ideas and then reviews some representative methods.

Transfer-based attacks Transfer-based black-box attack methods assume that adversarial samples crafted for one model are also able to attack other models [27], and the general scheme of transfer-based attacks is as follows: first, the adversarial examples are produced through white-box attack methods on a white-box model, and then these adversarial examples are employed to launch attacks on other black-box models. Therefore, how to enhance the transferability of the adversarial samples is crucial. Currently, transfer-based attacks can be roughly categorized into gradient-based and input transformation-based methods, where gradient-based transfer attacks mainly use advanced momentum technology to improve the transferability of adversarial examples, in which the calculation of momentum depends on the gradient of the model to the image. Input transformation-based transfer attacks focus on modifying the spatial features of the input image to improve the transferability of adversarial examples, such as random cropping, padding, scaling and other operations.

In gradient-based methods, FGSM is generally used as the basis to enhance the transferability of the adversarial samples by utilizing some advanced gradient computation methods. For example, Dong et al. [28] proposed Momentum Iterative Fast Gradient Sign Method (MI-FGSM) to enhance the transferability by utilizing the momentum technique. As shown in Eqs. (10) and (11):

$$g_{t+1} = \mu \cdot g_t + \frac{\nabla_{x_{\text{adv}}^{t}} J(\theta, x_{\text{adv}}^{t}, t)}{\left\| \nabla_{x_{\text{adv}}^{t}} J(\theta, x_{\text{adv}}^{t}, t) \right\|_1}, \tag{10}$$

$$x_{\text{adv}}^{t+1} = x_{\text{adv}}^{t} + \frac{\varepsilon}{T} \times \text{sign}\left(\nabla_{x_{\text{adv}}^{t}} J\left(\theta, x_{\text{adv}}^{t}, y\right)\right), \tag{11}$$

where g_t is called the momentum term, and $g_0 = 0$. MI-FGSM can stabilize the update direction and prevent the method from falling into a local optimum by accumulating a momentum term. Experimental results demonstrate that MI-FGSM significantly improves the transferability of the adversarial samples compared to BIM and FGSM. Furthermore, based on MI-FGSM, Wang et al. [29] introduced the neighborhood idea and a momentum variance term, in which the current momentum is fine-tuned by computing the average gradient of data points in the neighborhood of examples for the purpose of helping the algorithm to escape from the local optimum and improving the transferability. Recently, inspired by [29], Li et al. [30] proposed an adaptive mechanism to enhance the global search ability of the algorithm, where the authors introduce a forward momentum bias mechanism and a linear adaptive method to balance the global exploration and local convergence ability. In addition, a fine-tuning perturbation method is proposed to enhance the generalization of the adversarial examples. Experimental results demonstrate that the proposed method achieves competitive performance.

Input transformation-based methods usually perform random cropping, scaling and other operations on the input image to enhance the transferability. Xie et al. [31] proposed the first input transformation-based method, diverse input method. The proposed method randomly crops the image in the intervals [299,330). Then, the cropped image is restored to the size of 330×330×3 using padding, and the gradient of the image is computed by inputting into the network. Dong et al. [32] proposed the translation-invariant method, which computes the average gradient over a set of translated images. Lin et al. [33] proposed scale-invariant method using the scale invariant property. Before calculating the gradient, the input images need to be scaled by a scaling factor $1/2^i$, where i is a hyperparameter.

Several other techniques have been developed in addition to the two types of methods mentioned above. For instance, Liu et al. [34] proposed a model ensemble method, where the authors simultaneously use the gradient of multiple models to guide the updating of the adversarial samples. Hence, the transferability can greatly improve compared to a single white-box model. In [35], an attention mechanism-based loss is employed to replace the cross-entropy loss function to enhance the transferability. In this method, the authors pointed out that although different models have different structures, they have similar attention when recognizing images. Thus, increasing the attention loss of the models can enhance the transferability of the adversarial examples.

Query-based attacks Since only the output information of the model can be obtained, query-based methods usually use the difference of the model output to approximate the update direction of the perturbation. For example, Chen et al. [36] proposed Zeroth Order Optimizer (ZOO), which evaluates the gradient by adding perturbation to 128 pixels at each iteration. Brendel et al. [37] proposed the boundary attack, which first adds a large perturbations to the original example, so that the model misclassifies the sample in the initial stage. Then, the perturbation size is gradually reduced while keeping the adversarial examples aggressive. However, the above methods usually need to consume many queries, since the perturbation needs to be added repeatedly to query the output of the model. To reduce the number of queries, some works proposed different modeling methods to improve the efficiency of the attack. In [38], a square attack is proposed, where the square color block is used as the perturbation, and the coordinate position and perturbation value of the square color block are optimized by random search. Shukla et al. [39] proposed a Bayesian optimization-based black-box attack, in which the query results of the surrogate model replace those of the real model to improve the attack efficiency.

In the query-based methods, another commonly used technology is to utilize evolutionary algorithm (EA) to produce adversarial samples [40]. Compared with the traditional optimization algorithm, EA is not restricted by the characteristics of the problem, and it is able to deal with discontinuous, nonlinear and non-differentiable optimization problems [41]. Query-based adversarial attacks are a typical black-box optimization problem as the gradient information of the objective function cannot be obtained. Therefore, many scholars have turned to using EA to address the problem of black-box adversarial attacks [42]. Alzantot et al. [43] proposed GenAttack using genetic algorithm, where the uniform distribution is used to generate perturbations. Furthermore, the perturbations are added to the original sample to produce adversarial examples. Then, the generated adversarial examples are used as the initial population. Finally, GenAttack iteratively optimizes the perturbation using crossover, mutation, and selection until a successful adversarial example is obtained. In [44], a momentum-based differential evolution algorithm is proposed. Different from GenAttack, the optimization variable is designed as the gradient direction, that is, the optimization variable only has two values of 1 and −1. Hence, the individuals in the population are the sign matrix with the same dimension as the image. After the sign matrix is determined, inspired by the idea of FGSM, the perturbation matrix can be obtained by multiplying a predefined perturbation value. Finally, the perturbation matrix is added to the original example to obtain the adversarial example. If the attack fails, the sign matrix is optimized by employing differential evolution.

The above methods consider the adversarial attacks as a single-objective optimization problem. In [45], a pixel-related multi-objective optimization method is proposed, where the first optimization objective is to minimize the probability that the adversarial example is predicted as the true label, and other optimization objective is to minimize the l_2 norm of the adversarial perturbations. Tian et al. [46] designed the adversarial attacks as a constrained multi-objective optimization problem, in which l_2 and l_0 norm of the adversarial perturbation are employed as two optimization objectives, respectively, and the produced adversarial samples must successfully execute an attack as the constraint condition. To address the above optimization problem, a multi-objective evolutionary algorithm based on dual population is proposed. In addition, some work also uses particle swarm optimization algorithm (PSO) [47], covariance matrix adaptive evolution strategy(CMA-ES) [48] and other EA [49] to produce adversarial samples.

Although EA based the black-box attacks have achieved certain results, the optimization variables have the same dimensions as the images. For instance, in the ImageNet data set, the size of an image is 224×224×3, that is, the dimensions of the optimization variables are 150,528, resulting in a larger search space that significantly affects the optimization ability. To address this problems, Li et al. [50] proposed a novel problem formulation and designed three neighborhood search algorithm based on differential evolution, where the optimization variable is transformed from the pixel space to the weights space by introducing a neighborhood, and the dimension is only related to the neighbourhood size. Meanwhile, the weights are optimized by differential evolution to generated the final adversarial samples.

2.2 Sparse attacks

Jacobian-based saliency map attack (JSMA) Most attack methods rely on full pixels, but some studies have shown that adding perturbations to only a few pixels can mislead classifier predictions. In [51], the saliency heat map of images is first constructed using Jacobian matrices, then the adversarial samples are generated by iteratively changing some pixel points that are most critical to the model decision. Experimental results demonstrate that JSMA can only modify 4% pixels to produce adversarial samples.

Adversarial scratches Unlike the JSMA that performing attack on discrete pixels, Giulivi et al. [52] proposed an adversarial scratches-based sparse attack. Figure 2 provides the produced adversarial samples. Specifically, this method first initializes the position coordinates of three pixels, and then the Bezier curve is used to generate adversarial scratches. In addition, the pixel value of each color channel in the adversarial scratches is the same. Finally, the adversarial samples with scratches are produced by continuously optimizing the coordinates of the three pixels and the pixel values.

RP_2 The above methods can only be implemented in digital space. To attack the road sign classifier under different real physical conditions, such as angle, distance and light, Eykholt et al. [9] developed a robust physical perturbation. The perturbation consists of only black and white stickers, which can mislead the predictions of the classifier by pasting the perturbation to the road signs. The adversarial samples are presented in Fig. 3.

One-pixel attack Considering an extreme case, Su et al. [53] proposed a one-pixel attack algorithm, as shown in Fig. 4, One-pixel attack can attack the classification model by disturbing only one pixel. In detail, One-pixel attack

Fig. 3 Adversarial samples produced by RP_2, where the attack type is sparse attack

Fig. 2 Adversarial samples produced by adversarial scratches, where the attack type is sparse attack

Fig. 4 Adversarial examples generated by one-pixel, where the attack type is sparse attack

employs the coordinate position of a single pixel and the pixel value of the three channels as the optimization variable, and differential evolution is used to obtain the best attack pixel and perturbation value. In addition, the authors also verify the three-pixel and five-pixel attacks.

2.3 Universal adversarial perturbations

The universal adversarial attacks aim to generate a perturbation that is added to multiple images such that the majority of the adversarial samples mislead the predictions results [54].The universal adversarial attacks can be modeled as follows:

$$\max \sum I(y(x+v) \neq y(x)), x \in X, \tag{12}$$

where I represents the indicator that outputs 1 when the generated adversarial example is successful and 0 otherwise. y involves the output of the model. x and v denote the clean image and the generated universal perturbation, respectively, and X is the data set. Moosavi-Dezfooli et al. [55] discovered the existence of universal perturbation for the first time and revealed the important geometric correlation in the decision boundary of the classifier. Based on the above findings, the first universal adversarial perturbations attack is proposed, and this method proves that there is a potential security vulnerability in a single direction of the input space of the model, which may be exploited by an attacker to generate perturbations to mislead the prediction of the classifier on the majority of natural images. However, the proposed method is a white-box attack method. Furthermore, Das et al. [56] studied the black-box universal adversarial perturbations and developed a sparse universal black-box attack method. This method uses the differential evolution to optimize Eq. (12), where the optimization variable is composed of $(c+2) \times p$ elements, c is the RGB three-channel pixel value, and 2 represents the coordinate dimension of the pixel point. p is the number of optimized pixels, thus, the adversarial perturbation can be either a universal adversarial with few pixels or full pixels, depending on the setting of the user with respect to p.

2.4 Style transfer attacks

The existing attack methods produce adversarial samples by generating perturbations, while the style transfer attacks mainly generated adversarial examples by modifying the attributes of the image and introducing some stylized textures. For example, Wei et al. [57] used chroma, luminosity, sharpness and contrast in the image as optimization variables, and EA is introduced to optimize the four attributes and produce adversarial examples. The produced adversarial samples are presented in Fig. 5. Duan et al. [58] used a feature extractor to transfer the

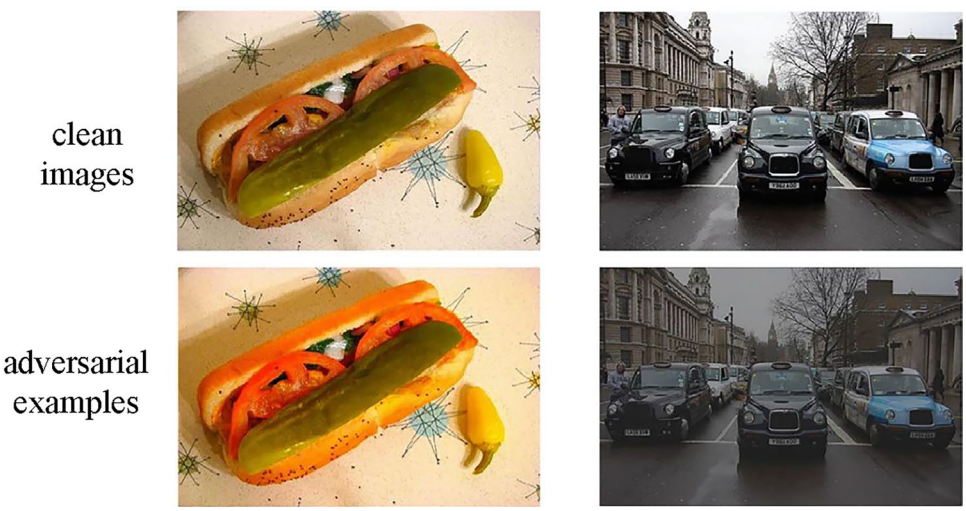


Fig. 5 Adversarial examples generated by [57], where the attack type is style transfer attack

texture information of a style image to the target image, and the loss function is designed to ensure that the content of the target image is not destroyed, as shown in Eq. (13):

$$L = (L_s + L_c + L_m)\lambda \cdot L_{adv}, \tag{13}$$

where L_{adv} represents the adversarial loss, and L_s is the style loss. L_c denotes the content loss that ensures the content does not change between the adversarial example and the original image. L_m is the smoothness loss. The adversarial examples are shown in Fig. 6.

3 Object detection-based adversarial attacks

The object detection task is to recognize and classify all objects appearing in the image, including the detection box of the object and the label of the corresponding object. Unlike classification-based adversarial attacks, object detection-based adversarial attacks tend to generate perturbations to make the detection box disappear or change the predicted label of the object. In addition, object detection-based adversarial attacks have two characteristics: first, the adversarial perturbation is usually an adversarial patch that is pasted into the image to facilitate implementation in the physical world. Second, the generated patch is generally universal, i.e., only one patch is generated for the whole data set. Currently, object detection-based adversarial attacks are broadly divided into patch attacks, 3D renderer camouflage attacks and infrared attacks.

3.1 Patch attacks

Patch attacks usually initialize a random patch, and the adversarial example is formed by pasting the patch to the object in the image. Then, different loss functions are designed in the optimization phase and the gradient descent method is applied to optimize the patch iteratively. For example, Thys et al.[59] carried out a study on adversarial patch attack for person detection, and the attack process is shown in Fig. 7, where the adversarial patch is randomly initialized, and then a series of transformations are performed on the adversarial patch, including rotation and scaling to enhance the robustness of the adversarial patch. Next, the adversarial examples are generated by pasting the patch to the object in the image and input to the target detector. Finally, based on the designed loss functions, the adversarial patch is updated using backpropagation, where the loss functions are as follows:

$$L = \alpha \cdot L_{nps} + \beta \cdot L_{tv} + L_{obj}, \tag{14}$$

where L_{obj} involves the output of the model and L_{nps} denotes the printable loss of the patch, which is used to constrain the color of the generated patch to be as printable as possible. The aim is to minimize the color distortion when the generated patch is printed. The detail is shown in Eq. 15:

$$L_{nps} = \sum_{p_{patch} \in P} \min_{c_{print} \in C} \left| p_{patch} - c_{print} \right|, \tag{15}$$

where p_{patch} is the pixel in the patch P and c_{print} is a color in a set of printable colors C. In addition, L_{tv} is used to constrain the generated patch to be smooth, and L_{tv} can be calculated as follows:

Fig. 6 Generated adversarial samples by [58], where the attack type is style transfer attack

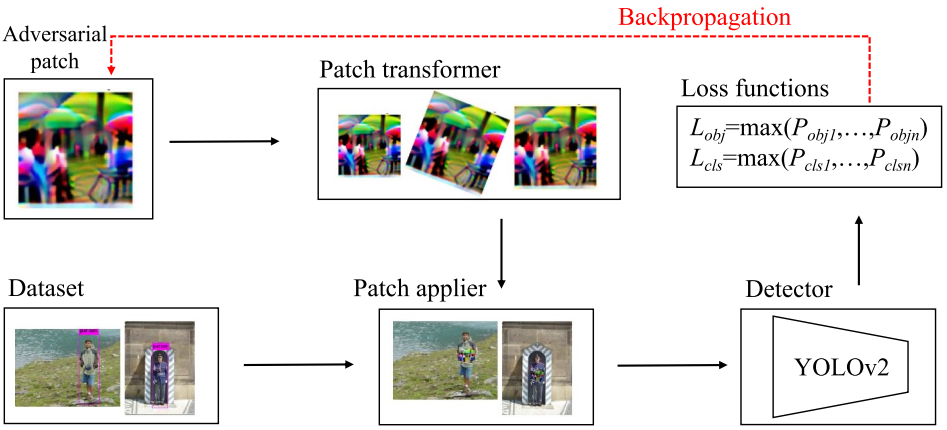


Fig. 7 Pipeline of person detection-based adversarial patches

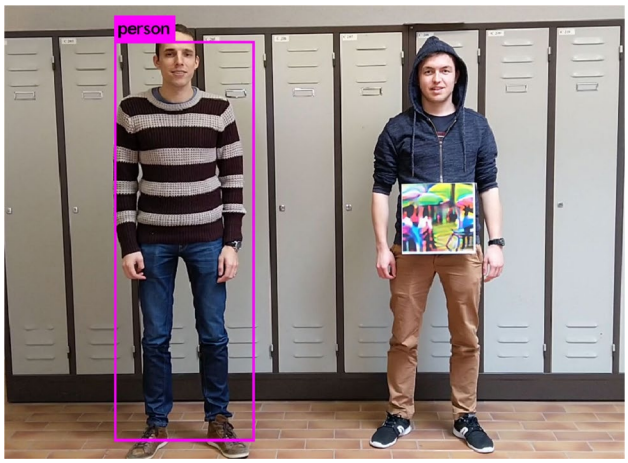


Fig. 8 Generated adversarial samples by [59], where the attack type is patch attack

$$L_{\text{tv}} = \sum_{i,j} \sqrt{(p_{i,j} - p_{i+1,j})^2 + (p_{i,j} - p_{i,j+1})^2}, \tag{16}$$

$p_{i,j}$ represents the pixel in the ith row and jth column of the patch, and the more similar the neighboring pixel values are, the smaller this loss value is. The final attack effect achieved by this method is shown in Fig. 8. Furthermore, Tang et al. [60] applied adversarial patches to the aerial imagery object detections by improving the loss function, and the generated adversarial examples are presented in Fig. 9. In addition, there are also some works for the naturalness [61, 62] and the location [63] of the patch, but most of the current patch attacks are the white-box, how to design a black-box-based patch attack method will be meaningful work.

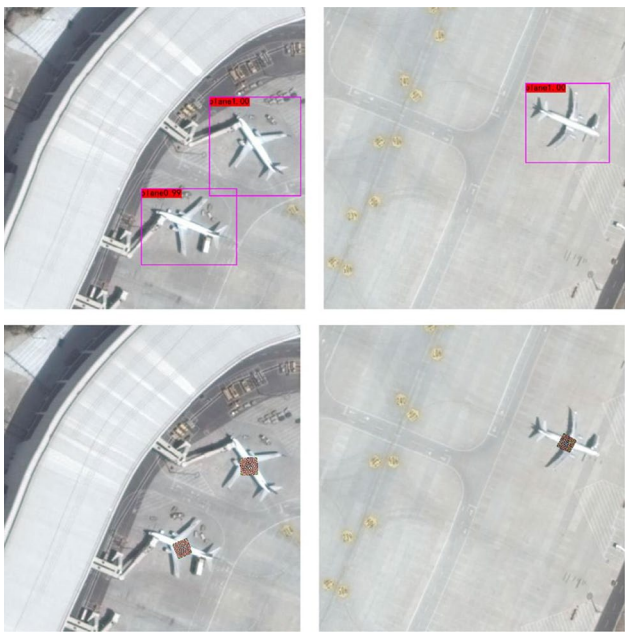


Fig. 9 Generated adversarial samples by [60], where the attack type is patch attack

3.2 3D renderer camouflage attacks

Patch-based adversarial attacks focus on hiding objects from detectors, but the patches only cover planar parts of the object surface, and the adversarial examples can not achieve successful attack in some physical scenes, such as multi-view and long distance. To address the above issue, some scholars have proposed 3D renderer camouflage attacks, where the texture of the whole surface of the 3D object in the 3D simulation environment is optimized to attack the object detector. For instance, Wang et al. [64] developed a full coverage camouflage adversarial attack (FCA), whose framework is presented in Fig. 10. First, to simulate vehicles driving in physical scenes, Carla, an open-source autonomous driving platform, is used as a simulation environment, which provides a variety of virtual scenes, such as cities and suburbs. The training data can be collected according to the task requirements. In the initial stage, the adversarial texture is randomly

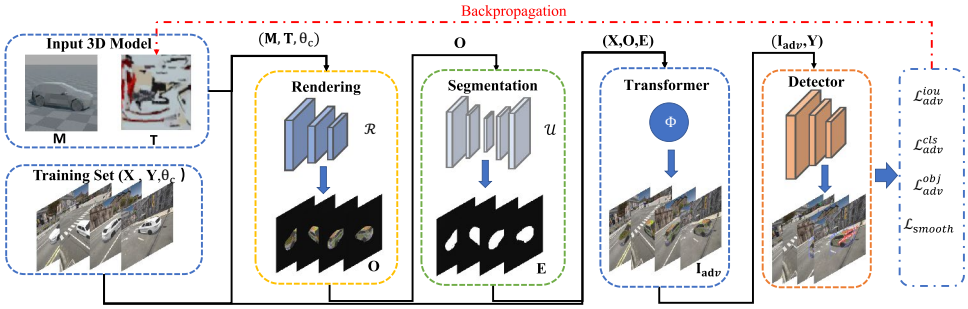


Fig. 10 Pipeline of FCA

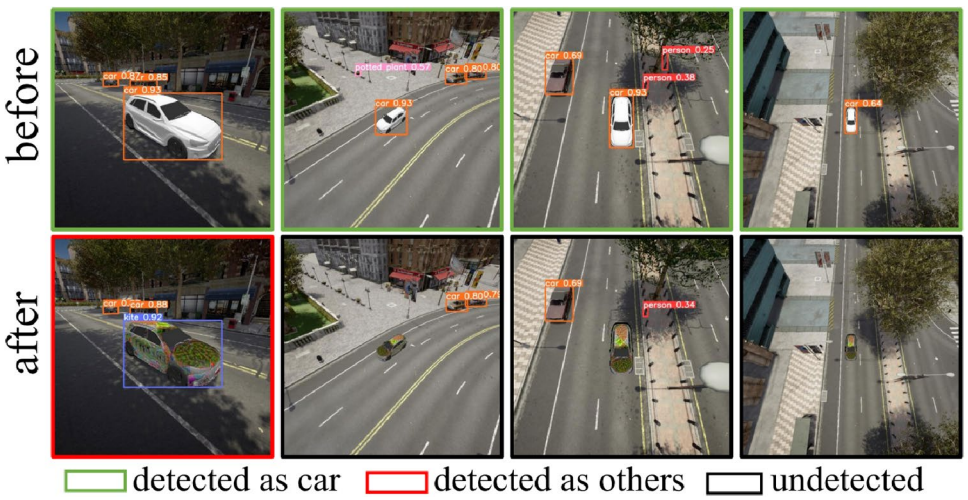


Fig. 11 Generated adversarial samples by FCA under different view angles, where the attack type is 3D renderer camouflage attack

initialized and rendered to the 3D car model. Furthermore, the rendered car is spliced with the background of the simulation environment to generate the camouflage car with the background. Finally, the adversarial example is input into the detector to obtain the output information, and the loss functions are optimized to update the adversarial texture. The generated adversarial samples are presented in Fig. 11.

The existing adversarial camouflages are easily detectable by the human eye due to their colourful nature, which restricts their practical application in the physical world. To address this issue, Sun et al. [65] developed a dual adversarial camouflage approach, where the proposed method devised a novel texture generation strategy. First, according to the environment, the global texture of the camouflage is obtained by optimizing the mean square error loss, so as to generate a camouflage that can adapt with the environment background for the purpose of deceiving the human eye. Then, this texture is optimized again with three different loss functions to achieve the generation of final camouflage that can fool the human eye and the object detector simultaneously. The generated adversarial samples are presented in Fig. 12.

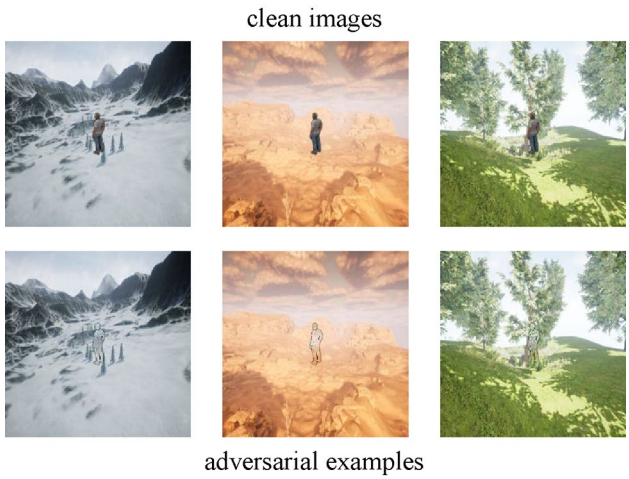


Fig. 12 Generated adversarial samples by [65], where the attack type is 3D renderer camouflage attack

3.3 Infrared attack

The existing adversarial attack methods are mainly based on the visible light domain, while the infrared domain is still in a blank stage. Infrared imaging is widely used in various fields, including temperature measurement and security

monitoring. Compared with the visible light images, infrared images have only one gray channel. Therefore, the amount of texture information in an infrared image is significantly lower than that in a visible light image. Some researchers use the heat map of the hot object as the basic module, and the position of the module is optimized to achieve the physical attack. Zhu et al. [66] found that the infrared imaging of the bulb is similar to the light spot, which is suitable as the basic module of the infrared attack and proposed a bulb attack. The generated bulb patches and adversarial examples are shown in Fig. 13; however, this method can only be executed at certain angles. To solve this issue, Zhu et al. [67] developed a wearable aerogel clothing based on "QR code" pattern using the thermal insulation characteristics of aerogel. As shown in Fig. 14, the method first optimizes the loss function to obtain an adversarial QR code. Then, the adversarial pattern is printed as the texture of the clothing, and the cylindrical aerogel is pasted on the corresponding black area of the aerogel clothing. Experiment results illustrate that the crafted clothing can effectively attack infrared pedestrian detectors at different angles. In addition, Hu et al. [68] designed a black-box infrared attack, named AdvIB, which uses cold and hot patches as physical perturbations, and the position of the patches is optimized using differential evolution to produce adversarial samples.

4 Adversarial attacks on other computer vision tasks

In addition to classification and detection tasks, a number of adversarial attack methods on other vision tasks have been proposed. This section will introduce adversarial attack methods on semantic segmentation [70], object tracking [71] and 3D point cloud tasks [72].

Attacks on semantic segmentation The purpose of semantic segmentation task is to segment the contents of the image through the model. As shown in Fig. 15, the left side is the original image, and the right side is the result of model segmentation, where each color represents a object in the image. Currently, semantic segmentation has been widely used in automatic driving, medical imaging and other fields. Arnab et al. [73] used adversarial samples for the first time to evaluate the robustness of the semantic segmentation model, in which FGSM was introduced as the attack method to generate adversarial samples. Figure 16 shows the segmentation results of different semantic segmentation models on adversarial samples, and it can be seen that only the right half of the image was perturbed, and the performance of each advanced segmentation models was greatly degraded. Furthermore, Nesti et al. [74] tested the effectiveness of adversarial samples on the semantic segmentation model in the real automatic driving scenario, and the experimental results clarified that the existing attack methods can achieve better attack performance in the digital space, but in the physical world, adversarial samples are less effective. In addition, Xie et al. [69] found that both semantic segmentation and object

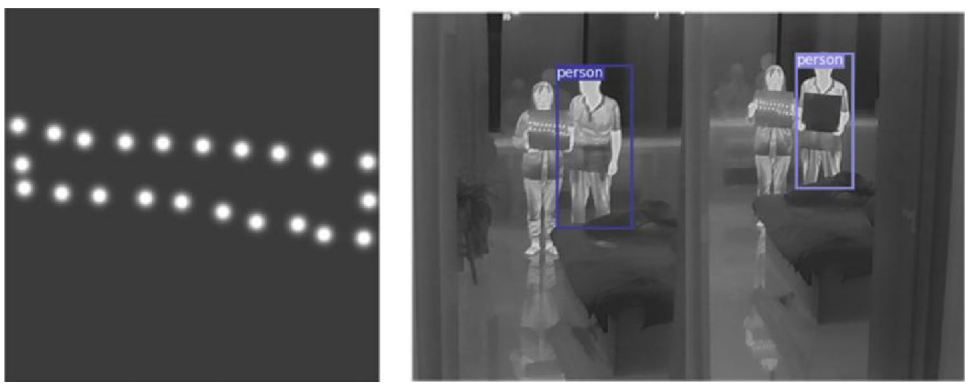


Fig. 13 Generated infrared adversarial patch and an example of physical infrared attack

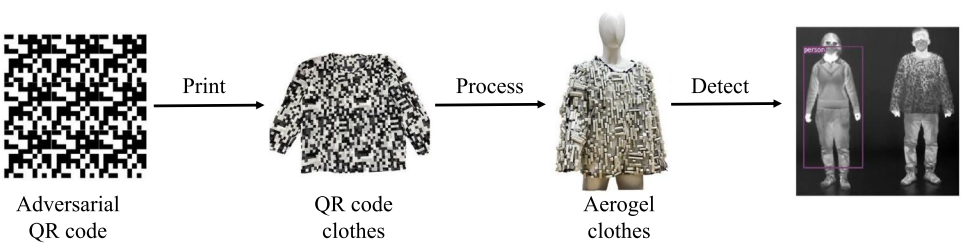


Fig. 14 Pipeline of "QR code"-based infrared attack

Fig. 15 Example of semantic segmentation

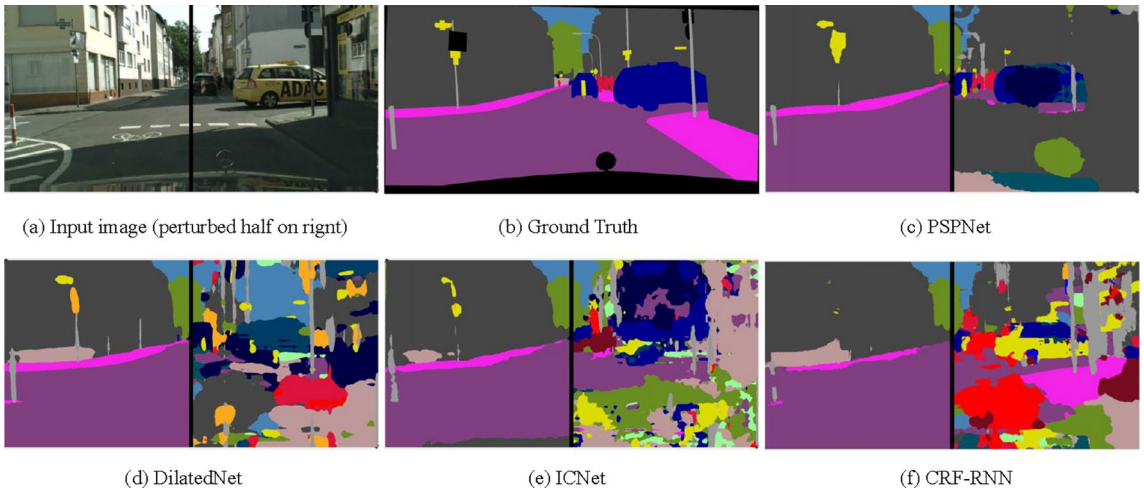


Fig. 16 Segmentation results of different semantic segmentation models on adversarial samples

detection tasks classify multiple objects in an image, hence, they proposed a novel algorithm for generating a adversarial perturbation to attack simultaneously the semantic segmentation model and the object detection model.

Attacks on object tracking The object tracking task is to accurately track any object or a specified class object over the entire time series after the object detector detects the initial object, and this technology is widely used in video surveillance, robot and other fields. Yan et al. [76] developed an attack technique for fooling single-object tracking models, where the proposed method constructs the adversarial frames by training the generator model with the designed loss function, causing the object to deviate from the search region and forcing the bounding box to shrink during tracking. Similarly, Chen et al. [77] only added small perturbations to the object in the initial frame, resulting in the tracking model losing the object in subsequent frames. Wiyatno et al. [75] proposed a physical attack method, where the adversarial patch was used to deceive a single-object tracking model, as shown in Fig. 17, when the tracked object moves in front of the adversarial patch, the adversarial texture made the tracker lock the adversarial patch behind the object instead of the object itself, resulting in object tracking failure.

Attacks on 3D point cloud The focus of the attack in the 3D scenarios is to deceive the model by changing the geometry of the 3D object, and the input of the model is changed from the 2D image to the 3D point cloud format or its corresponding mesh format data. The point cloud data uses 3D point coordinates to represent the 3D shape of the object, which is collected by the LiDAR sensor, and the corresponding mesh data uses the triangular facets to define the surface of the object. As shown in Fig. 18, from left to right are the 3D object and the corresponding 3D point cloud data and mesh data, respectively. In general, the attack methods in 3D scenarios modify the mesh data of the object to achieve the attack. When conducting physical experiments, 3D printing technology is used to print out physical objects and evaluate the performance of the attack in real scenarios. For example, Cao et al. [78] proposed an attack method to implement an attack on the Apollo self-driving detection system. As shown in Fig. 19, the proposed method generates a white adversarial obstacle, and the physical experiments show that

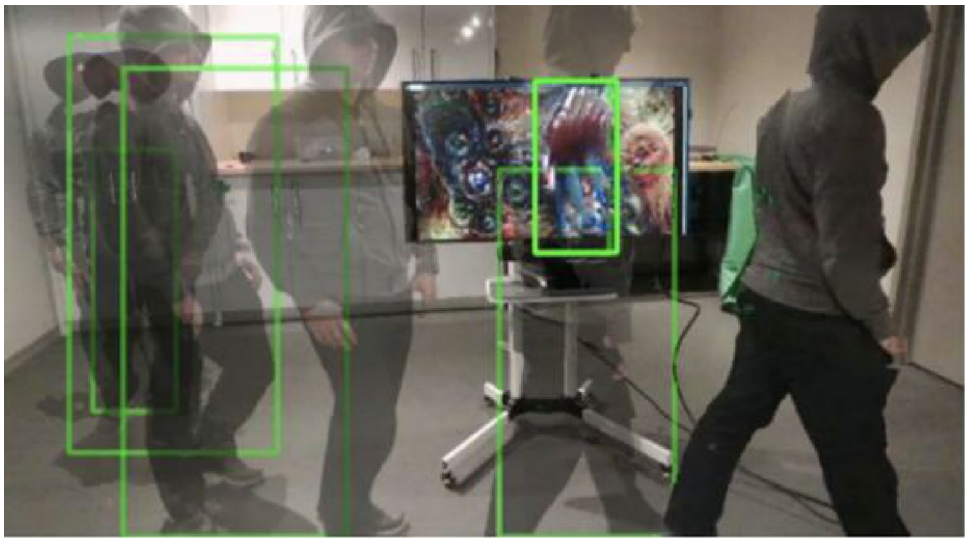


Fig. 17 Physical adversarial attack for object tracking

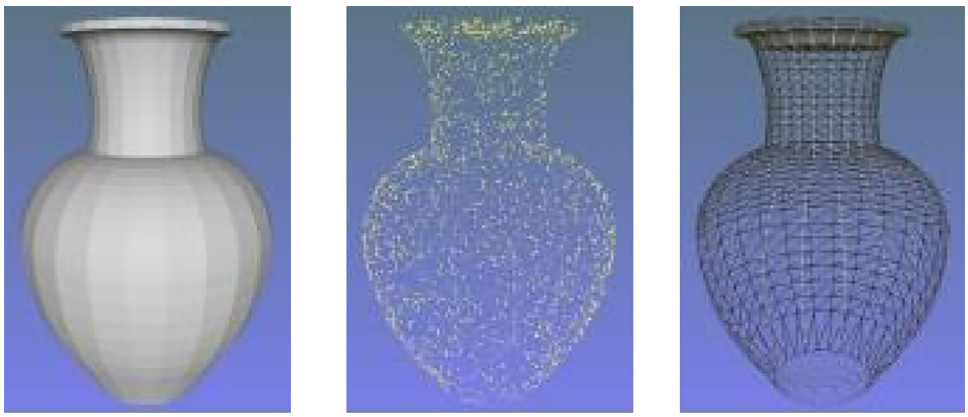


Fig. 18 Diagram of 3D objects, point cloud data and mesh data

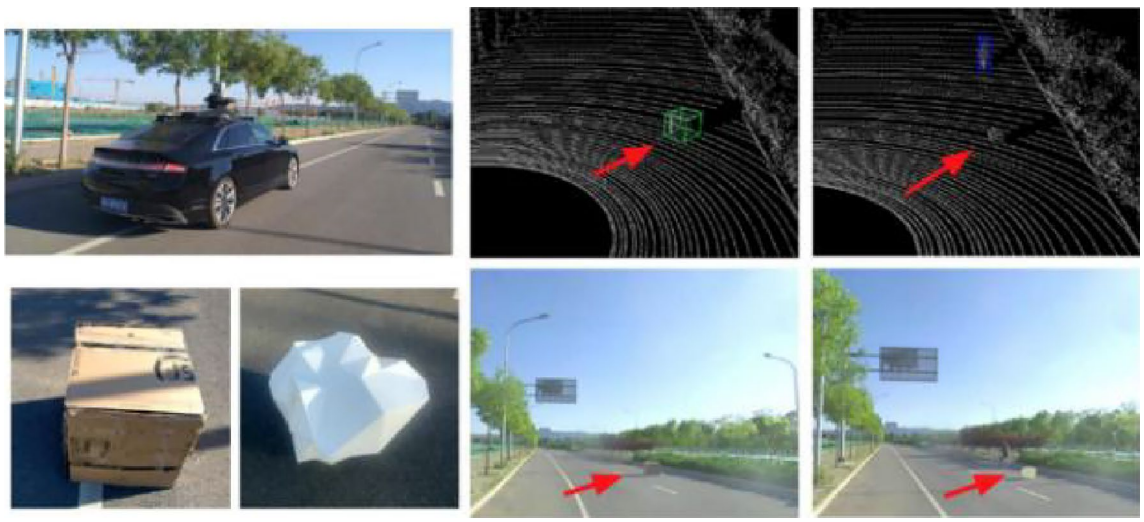


Fig. 19 Adversarial objects based on 3D point clouds

the self-driving car can successfully recognize the delivery box, but cannot detect the presence of the adversarial obstacle.

5 Adversarial defense

The goal of adversarial defense is to eliminate the impact of adversarial perturbation or enhance the robustness of the model. Currently, adversarial defense methods can be broadly classified into model-based and adversarial example-based methods. In the former, adversarial training [79] can be regarded as a classical defence, where the model is trained by mixing adversarial examples and clean examples for the purpose of regularizing the model parameters and improving the robustness of the model. In particular, it can be modeled as a minimum–maximization optimization problem shown in Eq. (17):

$$\min_{\theta} E_{(x,y)\in D}[\max \mathrm{L}(x_{\mathrm{adv}}, y, \theta)], \tag{17}$$

where D represents the data set, and x_{adv} is the adversarial example. y and θ are the true labels of the image and the model parameters, respectively, L denotes the loss function. Adversarial training first performs a maximum optimization internally to generate adversarial examples, where FGSM or PGD are usually employed as the attack methods. Then, a minimum optimization is executed externally to find a robust parameter y to approximate the distribution of the original data set. Gradient regularization [80] is another more classical defense method, which mainly introduces a penalty factor during model training to control the sensitivity of the model output to input changes, so that small adversarial perturbation will not cause drastic changes in the model output, but gradient regularization will cause a decrease in the accuracy of the model. The purpose of ENAS is to automatically search the optimal neural network structure on the specific tasks through evolutionary algorithms [81–83]. To resist the security threats posed by adversarial examples, researchers use ENAS to design robust neural network architectures. Specifically, in a given search space containing multiple model blocks, ENAS first randomly combines different model blocks as the initial population, where each individual in the population represents a neural network architecture. Then some new neural network architectures are produced using crossover and mutation operators, and these neural network architectures are evaluated on adversarial examples to obtain the robust accuracy as the fitness value. Finally. the optimal robust neural network architecture is updated according to the fitness value. For example, Liu et al. [84] took the model prediction accuracy on clean examples and the robust accuracy on adversarial samples as two optimization objectives. Through the designed multi-objective evolutionary algorithm, a robust neural network structure that can still achieve good classification accuracy under different attack methods is obtained. Although the above methods have been proved to obtain satisfactory defense effects, these methods are usually time-consuming.

Adversarial example-based methods usually preprocess the generated adversarial examples before model prediction, including adding some randomization operations or denoising. Xie et al. [85] pointed out that randomly adjusting the size of adversarial examples or adopting padding operation would reduce the aggressiveness of adversarial samples. In [86], JPEG compression method was proposed, and the authors stated that when the intensity of adversarial perturbation is small, JPEG compression can eliminate the effect of perturbations on model performance. However, with the increase of the intensity of adversarial perturbation, the defense performance of JPG compression will gradually decline. Xu et al. [87] proposed a feature compression defense mechanism to reduce the image feature space available for attackers, where the proposed method explored two simple mechanisms to compress image features. Furthermore, the model evaluates the input and that of performing feature compression, respectively, and if the predicted variance exceeds the set threshold, the input is considered as an adversarial example.

6 Challenges and promises

In the previous sections, we present an extensive literature survey on adversarial attack methods for image classification and object detection tasks. Meanwhile, we also review some common defense methods. It is concluded that adversarial attacks have become a hot topic in AI field. In this section, we discuss the current challenges in the adversarial attacks field and some emerging research directions.

Black-box attack efficiency A number of query-based attacks have been proposed, but they still have a common problem that requires excessive query numbers. Currently, although there have been some promising studies to address the mentioned issues, they destroy the quality of adversarial examples to improve the efficiency of attacks, or the attack performance of the methods is not satisfactory. Therefore, how to enhance the attack performance and ensure the quality of adversarial samples within a limited number of queries will be a meaningful research direction.

Performance of targeted attacks In white-box attacks, both non-targeted and targeted attacks are able to achieve better attack performance due to the use of gradients. However, in black-box attacks, compared with non-target attacks, the performance of targeted attacks needs to be improved, since the

attacker needs to mislead the model to identify the examples as a specific class, which increases the difficulty of the attack.

Transferability of adversarial examples As a primary black-box attack paradigm, transfer-based attack has attracted the attention of many scholars. Compared with the query-based attack, the transfer-based attack can quickly produce adversarial examples by exploiting gradients, but the poor transferability makes their attack performance inferior to query-based attacks. In recent years, various methods are designed for the purpose of enhancing the transferability, but the transferability has not been greatly improved, especially for the transferability between different model structures. Therefore, it remains a research hotspot to enhance the transferability of adversarial samples.

Black-box attack in object detection In object detection-based adversarial attacks, the existing works are generally white-box methods, that is, the gradient of the model is required to be obtained, however, in the physical world, it is almost impossible for the attacker to obtain the relevant knowledge of the model, resulting in a limitation on the application of the approach. Although a few black-box methods have been proposed so far, there remains a huge distance between their performance and that of white-box attacks. Furthermore, object detection-based attacks are also more difficult. Hence, how to implement object detection-based black-box attacks is a challenge.

Attack on other vision tasks At present, most of the existing work is based on classification and detection tasks, and there is little research on other fields, such as 3D point cloud, object tracking. These vision tasks have been broadly applied in the physical world. For instance, object tracking and 3D point cloud have been maturely used in automated driving, and the adversarial examples also poses potential risk to these applications. Therefore, developing adversarial attack research for these tasks is an important research direction.

Advanced defense methods Due to different attack technologies that have been proposed, some traditional defense technologies can not resist the attack of adversarial examples. Currently, adversarial training is regarded as the strongest defence technique, however, it needs to retrain the model when it faces new attack methods, and this process is time-consuming. Therefore, how to accelerate adversarial training is important. In addition, how to use some emerging technologies, such as automatic machine learning, to develop a general adversarial training model to resist various attacks is also a potential research direction.

7 Conclusions

DNNs have achieved satisfactory results in various fields, however, their vulnerability brings risks and challenges to the application of artificial intelligence, which has aroused the interest of many scholars. Hence, a large amount of adversarial attacks methods have been proposed in recent years. In this survry, based on different computer vision tasks, a comprehensive survey on adversarial attacks is conducted. Furthermore, the existing mainstream defense methods are introduced. Finally, we propose some challenges and future research directions with respect to the current development on adversarial attacks.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (No. 62376202).

Author contributions Chao Li wrote the main manuscript text and Handing Wang, Wen Yao, and Tingsong Jiang revised it. All authors reviewed the manuscript.

Declarations

Competing interests The authors declare no competing interests.

References

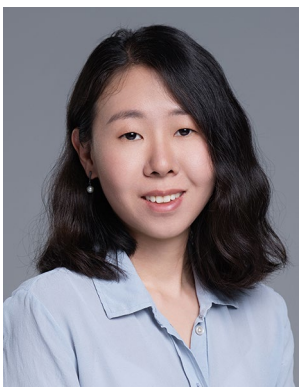
1. Kim, H. E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M. E., & Ganslandt, T. (2022). Transfer learning for medical image classification: A literature review. *BMC Medical Imaging, 22*(1), 69.
2. Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. Proceedings of the IEEE.
3. Li, C., Yao, W., Wang, H., Jiang, T., & Zhang, X. (2023). Bayesian evolutionary optimization for crafting high-quality adversarial examples with limited query budget. *Applied Soft Computing, 142*, 110370.
4. Wong, E., Schmidt, F., & Kolter, Z. (2019). Wasserstein adversarial examples via projected Sinkhorn iterations. In: International Conference on Machine Learning (pp. 6808–6817). PMLR.
5. Ilyas, A., Engstrom, L., & Madry, A. (2018). Prior convictions: Black-box adversarial attacks with bandits and priors. arXiv:1807.07978.
6. Komkov, S., & Petiushko, A. (2021). Advhat: Real-world adversarial attack on arcface face id system. In: 2020 25th International Conference on Pattern Recognition (ICPR) (pp. 819–826). IEEE.
7. Li, J., Ji, S., Du, T., Li, B., & Wang, T. (2018). Textbugger: Generating adversarial text against real-world applications. arXiv:1812.05271.
8. Wang, D., Yao, W., Jiang, T., Li, C., & Chen, X. (2023). Rfla: A stealthy reflected light adversarial attack in the physical world. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 4455–4465).
9. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., & Song, D. (2018). Robust physical-world attacks on deep learning visual classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1625–1634).
10. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing properties of neural networks. arXiv:1312.6199.
11. Akhtar, N., Mian, A., Kardan, N., & Shah, M. (2021). Advances in adversarial attacks and defenses in computer vision: A survey. *IEEE Access, 9*, 155161–155196.
12. Sun, H., Zhu, T., Zhang, Z., Jin, D., Xiong, P., & Zhou, W. (2023). Adversarial attacks against deep generative models on data: A

survey. *IEEE Transactions on Knowledge and Data Engineering, 35*(4), 3367–3388. https://doi.org/10.1109/TKDE.2021.3130903
13. Goodfellow, I.J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv:1412.6572.
14. Wang, Y., Liu, J., Chang, X., Rodríguez, R. J., & Wang, J. (2022). Di-aa: An interpretable white-box attack for fooling deep neural networks. *Information Sciences, 610*, 14–32.
15. Bai, Y., Wang, Y., Zeng, Y., Jiang, Y., & Xia, S.-T. (2023). Query efficient black-box adversarial attack on deep neural networks. *Pattern Recognition, 133*, 109037.
16. Feng, W., Xu, N., Zhang, T., & Zhang, Y. (2023). Dynamic generative targeted attacks with pattern injection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 16404–16414)
17. Reza, M.F., Rahmati, A., Wu, T., & Dai, H. (2023). Cgba: Curvature-aware geometric black-box attack. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 124–133)
18. Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine, 29*(6), 141–142.
19. Krizhevsky, A., & Hinton, G. et al. (2009). Learning multiple layers of features from tiny images.
20. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (pp. 248–255). Ieee
21. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C.L.: (2014). Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13 (pp. 740–755). Springer
22. Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision, 88*, 303–338.
23. Kurakin, A., Goodfellow, I., & Bengio, S, et al. (2016). Adversarial examples in the physical world.
24. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv:1706.06083.
25. Moosavi-Dezfooli, S.-M., Fawzi, A., & Frossard, P. (2016). Deepfool: A simple and accurate method to fool deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2574–2582).
26. Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In: 2017 Ieee Symposium on Security and Privacy (sp) (pp. 39–57). IEEE.
27. Wang, X., He, X., Wang, J., & He, K. (2021). Admix: Enhancing the transferability of adversarial attacks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 16158–16167).
28. Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., & Li, J. (2018). Boosting adversarial attacks with momentum. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 9185–9193).
29. Wang, X., & He, K. (2021). Enhancing the transferability of adversarial attacks through variance tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1924–1933).
30. Li, C., Yao, W., Wang, H., & Jiang, T. (2023). Adaptive momentum variance for attention-guided sparse adversarial attacks. *Pattern Recognition, 133*, 108979.
31. Xie, C., Zhang, Z., Zhou, Y., Bai, S., Wang, J., Ren, Z., & Yuille, A.L. (2019). Improving transferability of adversarial examples with input diversity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 2730–2739).
32. Dong, Y., Pang, T., Su, H., & Zhu, J. (2019). Evading defenses to transferable adversarial examples by translation-invariant attacks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4312–4321).
33. Lin, J., Song, C., He, K., Wang, L., & Hopcroft, J.E. (2019) Nesterov accelerated gradient and scale invariance for adversarial attacks. arXiv:1908.06281.
34. Liu, Y., Chen, X., Liu, C., & Song, D. (2016). Delving into transferable adversarial examples and black-box attacks. arXiv:1611.02770.
35. Chen, S., He, Z., Sun, C., Yang, J., & Huang, X. (2020). Universal adversarial attack on attention and the resulting dataset damagenet. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
36. Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., & Hsieh, C.-J. (2017). Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (pp. 15–26).
37. Brendel, W., Rauber, J., & Bethge, M. (2017). Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. arXiv:1712.04248.
38. Andriushchenko, M., Croce, F., Flammarion, N., & Hein, M. (2020). Square attack: a query-efficient black-box adversarial attack via random search. In: European Conference on Computer Vision (pp. 484–501). Springer.
39. Shukla, S. N., Sahu, A.K., Willmott, D., & Kolter, J. Z. (2019). Black-box adversarial attacks with Bayesian optimization. arXiv:1909.13857.
40. Li, Z., Cheng, H., Cai, X., Zhao, J., & Zhang, Q. (2022). Sa-es: Subspace activation evolution strategy for black-box adversarial attacks. *IEEE Transactions on Emerging Topics in Computational Intelligence*.
41. Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197.
42. Vidnerová, P., & Neruda, R. (2020). Vulnerability of classifiers to evolutionary generated adversarial examples. *Neural Networks, 127*, 168–181.
43. Alzantot, M., Sharma, Y., Chakraborty, S., Zhang, H., Hsieh, C.-J., & Srivastava, M. B. (2019). Genattack: Practical black-box attacks with gradient-free optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference (pp. 1111–1119).
44. Lin, J., Xu, L., Liu, Y., & Zhang, X. (2020). Black-box adversarial sample generation based on differential evolution. *Journal of Systems and Software, 170*, 110767.
45. Wang, J., Yin, Z., Jiang, J., Tang, J., & Luo, B. (2022). Pisa: Pixel skipping-based attentional black-box adversarial attack. *Computers & Security, 123*, 102947.
46. Tian, Y., Pan, J., Yang, S., Zhang, X., He, S., & Jin, Y. (2022). Imperceptible and sparse adversarial attacks via a dual-population-based constrained evolutionary algorithm. *IEEE Transactions on Artificial Intelligence, 4*(2), 268–281.
47. Zhang, Q., Wang, K., Zhang, W., & Hu, J. (2019). Attacking black-box image classifiers with particle swarm optimization. *IEEE Access, 7*, 158051–158063.
48. Ilyas, A., Engstrom, L., Athalye, A., & Lin, J. (2018). Black-box adversarial attacks with limited queries and information. In: International Conference on Machine Learning (pp. 2137–2146). PMLR.
49. Qiu, H., Custode, L.L., & Iacca, G. (2021). Black-box adversarial attacks using evolution strategies. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion (pp. 1827–1833).
50. Li, C., Wang, H., Zhang, J., Yao, W., & Jiang, T. (2022). An approximated gradient sign method using differential evolution

for black-box adversarial attack. *IEEE Transactions on Evolutionary Computation*.
51. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. In: 2016 IEEE European Symposium on Security and Privacy (EuroS &P) (pp. 372–387). IEEE.
52. Giulivi, L., Jere, M., Rossi, L., Koushanfar, F., Ciocarlie, G., Hitaj, B., & Boracchi, G. (2023). Adversarial scratches: Deployable attacks to cnn classifiers. *Pattern Recognition, 133*, 108985.
53. Su, J., Vargas, D. V., & Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation, 23*(5), 828–841.
54. Mopuri, K. R., Ganeshan, A., & Babu, R. V. (2018). Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 41*(10), 2452–2465.
55. Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1765–1773).
56. Ghosh, A., Mullick, S. S., Datta, S., Das, S., Das, A. K., & Mallipeddi, R. (2022). A black-box adversarial attack strategy with adjustable sparsity and generalizability for deep image classifiers. *Pattern Recognition, 122*, 108279.
57. Wei, X., Guo, Y., & Li, B. (2021). Black-box adversarial attacks by manipulating image attributes. *Information Sciences, 550*, 285–296.
58. Duan, R., Ma, X., Wang, Y., Bailey, J., Qin, A.K., & Yang, Y. (2020). Adversarial camouflage: Hiding physical-world attacks with natural styles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1000–1008).
59. Thys, S., Van Ranst, W., & Goedemé, T. (2019). Fooling automated surveillance cameras: adversarial patches to attack person detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.
60. Tang, G., Jiang, T., Zhou, W., Li, C., Yao, W., & Zhao, Y. (2023). Adversarial patch attacks against aerial imagery object detectors. *Neurocomputing, 537*, 128–140.
61. Hu, Y.-C.-T., Kung, B.-H., Tan, D.S., Chen, J.-C., Hua, K.-L., & Cheng, W.-H. (2021). Naturalistic physical adversarial patch for object detectors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 7848–7857).
62. Tang, G., Yao, W., Jiang, T., Zhou, W., Yang, Y., & Wang, D. (2023). Natural weather-style black-box adversarial attacks against optical aerial detectors. *IEEE Transactions on Geoscience and Remote Sensing*.
63. Liu, X., Yang, H., Liu, Z., Song, L., Li, H., & Chen, Y. (2018). Dpatch: An adversarial patch attack on object detectors. arXiv: 1806.02299.
64. Wang, D., Jiang, T., Sun, J., Zhou, W., Gong, Z., Zhang, X., Yao, W., & Chen, X. (2022). Fca: Learning a 3d full-coverage vehicle camouflage for multi-view physical adversarial attack. In: Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 36, pp. 2414–2422).
65. Sun, J., Yao, W., Jiang, T., Wang, D., & Chen, X. (2023). Differential evolution based dual adversarial camouflage: Fooling human eyes and object detectors. *Neural Networks, 163*, 256–271.
66. Zhu, X., Li, X., Li, J., Wang, Z., & Hu, X. (2021). Fooling thermal infrared pedestrian detectors in real world using small bulbs. In: Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, pp. 3616–3624).
67. Zhu, X., Hu, Z., Huang, S., Li, J., & Hu, X.(2022). Infrared invisible clothing: Hiding from infrared detectors at multiple angles in real world. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 13317–13326).
68. Hu, C., Shi, W., Jiang, T., Yao, W., Tian, L., Chen, X., Zhou, J., & Li, W. (2023). Adversarial infrared blocks: A multi-view black-box attack to thermal infrared detectors in physical world. Available at SSRN 4532269.
69. Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., & Yuille, A. (2017). Adversarial examples for semantic segmentation and object detection. In: Proceedings of the IEEE International Conference on Computer Vision (pp. 1369–1378).
70. Strudel, R., Garcia, R., Laptev, I., & Schmid, C. (2021). Segmenter: Transformer for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (pp. 7262–7272).
71. Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *ACM Computing Survey, 38*(4), 13. https://doi.org/10.1145/1177352.1177355
72. Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M. (2021). Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*(12), 4338–4364. https://doi.org/10.1109/TPAMI.2020.3005434
73. Arnab, A., Miksik, O., & Torr, P. H. S. (2018). On the robustness of semantic segmentation models to adversarial attacks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
74. Nesti, F., Rossolini, G., Nair, S., Biondi, A., & Buttazzo, G.(2022). Evaluating the robustness of semantic segmentation for autonomous driving against real-world adversarial patch attacks. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) (pp. 2280–2289).
75. Wiyatno, R. R., Xu, A.(2019). Physical adversarial textures that fool visual object tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).
76. Yan, B., Wang, D., Lu, H., & Yang, X. (2020). Cooling-shrinking attack: Blinding the tracker with imperceptible noises. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).
77. Chen, X., Yan, X., Zheng, F., Jiang, Y., Xia, S.-T., Zhao, Y., & Ji, R. (2020). One-shot adversarial attacks on visual tracking with dual attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).
78. Cao, Y., Xiao, C., Yang, D., Fang, J., Yang, R., Liu, M., & Li, B. (2019). Adversarial Objects Against LiDAR-Based Autonomous Driving Systems.
79. Zheng, S., Song, Y., Leung, T., & Goodfellow, I.(2016). Improving the robustness of deep neural networks via stability training. In: Proceedings of the Ieee Conference on Computer Vision and Pattern Recognition (pp. 4480–4488).
80. Ross, A., & Doshi-Velez, F.(2018). Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In: Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 32).
81. Sun, J., Yao, W., Jiang, T., & Chen, X. (2024). Efficient search of comprehensively robust neural architectures via multi-fidelity evaluation. *Pattern Recognition, 146*, 110038.
82. Zhou, X., Qin, A. K., Sun, Y., & Tan, K. C. (2021). A survey of advances in evolutionary neural architecture search. In: 2021 IEEE Congress on Evolutionary Computation (CEC) (pp. 950–957). https://doi.org/10.1109/CEC45853.2021.9504890.
83. Zhou, X., Qin, A. K., Gong, M., & Tan, K. C. (2021). A survey on evolutionary construction of deep neural networks. *IEEE Transactions on Evolutionary Computation, 25*(5), 894–912. https://doi.org/10.1109/TEVC.2021.3079985
84. Liu, J., & Jin, Y. (2021). Multi-objective search of robust neural architectures against multiple types of adversarial attacks. *Neurocomputing, 453*, 73–84.
85. Xie, C., Wang, J., Zhang, Z., Ren, Z., & Yuille, A. (2017) Mitigating adversarial effects through randomization. arXiv:1711.01991.

86. Dziugaite, G. K., Ghahramani, Z., & Roy, D. M. (2016). A study of the effect of jpg compression on adversarial images. arXiv:1608.00853.
87. Xu, W., Evans, D., & Qi, Y. (2017). Feature squeezing: Detecting adversarial examples in deep neural networks. arXiv:1704.01155.

Chao Li received the M.Eng. degree from Zhongyuan University of Technology, Zhengzhou, China, in 2020. He is pursuing the Ph.D. degree with the School of Artificial Intelligence, Xidian University, Xi'an, China. His research interests include adversarial machine learning, surrogate-assisted evolutionary optimization, and multi-objective multi-modal optimization.



Handing Wang received the B.Eng. and Ph.D. degrees from Xidian University, Xi'an, China, in 2010 and 2015, respectively. She is currently a professor with School of Artificial Intelligence, Xidian University, Xi'an, China. Dr. Wang is an Associate Editor of IEEE Transactions on Evolutionary Computation, IEEE Computational Intelligence Magazine, Evolutionary Computation, Swarm and Evolutionary Computation, Memetic Computing, and Complex & Intelligent Systems. Her research interests include nature-inspired computation, multiobjective optimization, multiple-criteria decision-making, surrogate-assisted evolutionary optimization, and real-world problems.

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 Defense Innovation Institute, Chinese Academy of Military Science, Beijing, China. Her current research interests include spacecraft systems engineering, multidisciplinary design optimization, uncertainty-based optimization, data-driven surrogate modeling, and evolutionary optimization.

Tingsong Jiang received the B.E. degree and the Ph.D. degree from School of Electronics Engineering and Computer Science, Peking University. He is currently an assistant professor with Defense Innovation Institute, Chinese Academy of Military Science. His research interests include adversarial machine learning and knowledge graph.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com