Hindawi
Security and Communication Networks
Volume 2022, Article ID 8288855, 12 pages
https://doi.org/10.1155/2022/8288855

Research Article

Improving the Imperceptibility of Adversarial Examples Based on Weakly Perceptual Perturbation in Key Regions

Yekui Wang,[1,2] **Tieyong Cao**,[1] **Yunfei Zheng**,[1,3,4] **Zheng Fang**,[1] **Yang Wang**,[1] **Yajiu Liu**,[2] **Lei Chen**,[1] **and Bingyang Fu**[1]

[1]*Command and Control Engineering College, Army Engineering University of PLA, Nanjing, China*
[2]*Unit 31401, Changchun, China*
[3]*The Army Artillery and Defense Academy of PLA, Nanjing, China*
[4]*The Key Laboratory of Polarization Imaging Detection Technology, Hefei, China*

Correspondence should be addressed to Tieyong Cao; cty_ice@sina.com

Received 16 September 2022; Revised 4 December 2022; Accepted 8 December 2022; Published 21 December 2022

Academic Editor: Leandros Maglaras

Deep neural networks have been proved vulnerable to being attacked by adversarial examples, which have attracted extensive attention from researchers. Existing GAN-based object detection adversarial example generation methods are efficient in generating speed but ignore the visual imperceptibility of adversarial examples. In this paper, to improve the visual imperceptibility of adversarial examples, we propose an object detection adversarial example generation method based on weakly perceptual perturbations in key regions. First, a positioning module based on the gradient-weighted activation mapping method is designed to analyze the key region of the object from the perspective of gradient propagation and use the key region in order to limit the range and amplitude of the perturbation. Second, the deep feature of the convolutional network is introduced to constrain the content of adversarial perturbation and improve the similarity between adversarial examples and original images. Finally, a postprocessing method based on median filtering is introduced to further correct the color deviation of adversarial examples and improve imperceptibility. The experimental results for VOC datasets show that the attack success rate increased by 4%, the PSNR increased by 8.6%, and the MSE and LPIPS decreased by 92.3% and 59.5%, respectively. It demonstrates that the proposed method can significantly improve the imperceptibility of the adversarial example with a high attack success rate.

1. Introduction

Deep neural networks (DNNs) have achieved remarkable results in computer vision tasks such as image classification [1], object detection [2–9], image retrieval [10], and medical image analysis [11]. At the same time, several works show that the deep network is easy to be fooled when adding the adversarial perturbation to the input. Adversarial examples threaten the security performance of the deep network, which has raised many concerns recently.

Most research studies on adversarial examples focus on image classification networks, and there are only few methods concentrating on adversarial examples for object detection models. The existing object detection adversarial example generation methods can be divided into two categories, the gradient-based optimization method and the generative adversarial network (GAN)-based method. Gradient-based methods generate adversarial examples through hundreds of gradient iterative steps, and the generated adversarial examples have relatively good visual effects but need a lot of computing resources.

GAN-based methods train a generator network to generate adversarial perturbations, which can quickly generate adversarial examples in the inference stage. GAN-based methods significantly reduce the generation time of adversarial examples. However, they still have the following problems: (1) As shown in Figure 1, limited by the structural performance and optimization method of the generator network, adversarial examples generated by GAN-based methods have obvious red noise; (2) GAN-based methods

do not effectively limit the amplitude and region of perturbations, and many perturbations have been added to the background region; (3) Existing GAN-based methods usually use the L_p norm to constrain perturbation generation, but the L_p norm only limits the amplitude of the pixel, and the difference between the image structure and texture change is not restricted, which makes the generated adversarial examples unable to be "invisible" to human eyes.

To design an object detection adversarial example generation method with high efficiency and good imperceptibility, we optimize the above problems of GAN-based methods and propose a weakly perceptual adversarial example generation method called WPAE (weakly perceptual adversarial example).

Our contributions are summarized as follows:

(1) We design a positioning module combined with the gradient-weighted class activation mapping (Grad-CAM [12]) method to generate an attack region mask and limit the range and amplitude of the perturbation, which avoids the redundant perturbation in the background on the premise of ensuring the attack effect.

(2) Combined with the image content representation extracted by DNN, we design a perceptual loss to constrain perturbation and use a postprocessing method based on median filtering to eliminate the abnormal noise in the adversarial examples generated by the GAN-based method, which can effectively improve the subjective visual effect while retaining the attack ability of adversarial examples.

(3) We study and summarize the perceptual performance of object detection adversarial examples generated by GAN-based methods and propose a weakly perceptual object detection adversarial example generation method, which greatly improves the imperceptibility of adversarial examples generated by GAN-based methods. The experimental results for VOC datasets show that all objective perceptual indicators are greatly improved, and it is difficult to distinguish the difference between the original images and adversarial examples under human observation.

2. Related Work

2.1. Object Detection. Object detection is one of the essential branches of computer vision. It has been widely used in pedestrian detection, automatic driving, security monitoring, etc., and achieved notable results. The existing object detection models can be roughly divided into single-stage and two-stage algorithms. RCNN [2], faster RCNN [3], and cascade RCNN [4] are classic two-stage algorithms. These algorithms divide the detection process into two steps: first, extracting the region proposals and then classifying and regressing the extracted region proposals. Single-stage algorithms directly carry out classification and regression on feature maps. SSD [5] series, YOLO [6] series, and RetinaNet [7] are typical representatives. These algorithms use the anchor mechanism, which is also known as the anchor-based algorithm. Anchor-free algorithms such as FCOS [8] and CornerNet [9], which abandon the anchor mechanism and detect based on key points, have also achieved good results in object detection.

2.2. Adversarial Examples and Defense Methods. Adversarial examples are widely used in image classification, object detection, image retrieval [10], and medical image analysis [11]. Szegedy et al. [13]first proposed the formal definition of adversarial examples and designed the L-BFGS method to generate adversarial examples. Goodfellow et al. [14] considered that the high-dimensional linear characteristic of the deep network is the fundamental factor for the existence of adversarial examples and proposed a fast gradient sign method (FGSM). Existing methods such as BIM [15], MI-FGSM [16], DIM [17], and PGD [18], based on FGSM, improve the original method by optimization strategy and gradient calculation and further enhance attack ability. Carlini and Wagner [19] proposed three attack methods based on L_0, L_2, and L_∞ distance metrics, which can effectively improve the attack success rate of the distillation defense network. Moosavi et al. [20] took the minimum distance from the adversarial examples to the decision boundary as an adversarial perturbation. The authors proposed an attack algorithm based on hyperplane classification called DeepFool.

In recent years, many scholars have begun researching the adversarial example for object detection and proposed some gradient-based attack methods. Cihang et al. [21] extended the adversarial example from image classification to semantic segmentation and object detection and proposed DAG (dense advantage generation) attacks on faster RCNN. DAG preserves all region proposals in the region proposal network (RPN) marked as positive samples and discards the remaining region proposals. Then, it sets a threshold condition to manually select high-quality region proposals from the above regions to attack. Each region proposal is randomly assigned an error label, which leads to model error classification. Yuezun et al. [22] proposed RAP (robust advantageous perturbation) and designed a loss function combining classification loss and location loss to attack the detector by destroying the RPN network unique to the two-stage object detection model. Liao et al. [23] proposed a CA (category-wise attack) method for the anchor-free object detection model. In this method, key pixel areas rich in high-level semantic information are found in the heatmap generated by using the detector to carry out classification attacks. Derui et al. [24] proposed a new attack method called Daedalus, which makes the NMS (nonmaximum suppression) module invalid by reducing the area of the regression box and increasing the distance between different boxes. All the abovementioned attack methods have achieved certain effects, but they require more number of iterations, and it takes a long time for training to generate adversarial examples.

To address the problem of the high computational cost of the gradient-based optimization method, Xingxing et al. [25]

Figure 1: (a) The adversarial example generated by UEA and (b) the adversarial example generated by FA. We can see many red noises in adversarial examples and many perturbations in the irrelevant background.

proposed an efficient adversarial example generation method called UEA (unified and efficient adversary) based on the GAN framework. The method expresses the generation of adversarial examples as an image-to-image translation problem: inputting a clean image into the model and outputting adversarial examples. The model used in UEA consists of two parts: a generator and a discriminator. Adversarial examples are generated using the generator, and clean images are sent to the discriminator simultaneously to judge whether the input image is added with perturbations. Then, judgment results are used to assist in generator training. When the generator is trained, it can generate an adversarial example for the input image after single calculation, significantly improving the generation speed of the adversarial example.

UEA can be applied to the object detection task. It extracts region proposals from RPN, takes the region proposal score as the weight value of the region, and accumulates the weight as a mask to constrain the perturbation and to add it to the foreground region. The L_2 loss between the real image and the adversarial example is calculated to optimize the perturbation in network training. However, the L_2 loss only represents the difference between corresponding pixels, and it cannot fully reflect the characteristics of the human visual observation mechanism. Furthermore, simply taking the region proposal score as the weight value of all pixels in the region proposal cannot sufficiently reflect the importance of different regions for model detection, and there is the case of adding redundancy to perturbations, which results in the poor perceptual performance of adversarial examples. The contrasts between adversarial examples generated by UEA and the original images are shown in Figure 1(a). Under human eye observation, adversarial examples are significantly different from original images.

The FA [26] algorithm is similar to the UEA method. It only uses the L_2 loss between real images and adversarial examples to guide the training of the generator and ensure the basic image quality in the training process. Since there is no open-source code for this method, we intercepted the original images in the paper to show the difference between the adversarial examples generated by FA and the original images, as shown in Figure 1(b). The results show a significant difference between the adversarial examples generated by FA and the real images, and we can observe the disturbing texture clearly.

To make the model more robust to adversarial attacks, many adversarial defense methods have been proposed. Preprocessing-based methods such as image compression are an effective defense method against adversarial attacks. Previous studies [27, 28] have shown that antagonistic perturbations can partially be eliminated by JPEG compression. Comdefense [29] is an end-to-end image compression model to defend adversarial examples. The model consists of a compressed convolutional neural network (ComCNN) and a reconstructed convolutional neural network (RecCNN). ComCNN is used to keep the structure information of the original image and eliminate the adversarial perturbation. RecCNN is used to reconstruct the original image with high quality.

2.3. Low-Frequency Perturbation. Literature [30] generates adversarial examples by discarding the information from the original image. Experiments show that the generated

adversarial examples tend to retain low-frequency information and discard high-frequency details. A large number of experiments in the literature [31] have proved that all defense methods used in the NIPS 2017 adversarial example challenge cannot resist the attack of adversarial examples containing low-frequency perturbations, and the effect of low-frequency perturbations is even better than that of high-frequency perturbations on attacking the defense model. The studies in literature [32] show that filtering out high-frequency information from adversarial examples can still maintain good attack ability, and the processed adversarial example has better robustness. The abovementioned research studies demonstrate that filtering out the high-frequency noise, which changes violently in the perturbation, and retaining only the low-frequency perturbation will not affect the attack ability of the adversarial example.

3. Approach

In this paper, we propose a weakly perceptual object detection adversarial example generation method, which is used to improve the imperceptibility of adversarial examples generated by GAN-based methods. The key idea is to design a positioning module to constrain the added region and amplitude of perturbations; the perceptual loss is designed to constrain the generation of adversarial examples; we used a postprocessing method based on median filtering to eliminate the abnormal noise and improve the subjective visual effect while retaining its attack ability.

We first define the problem in Section 3.1; then, we introduce our positioning module and perceptual loss in Sections 3.2 and 3.3. Finally, we describe the proposed WPAE algorithm.

3.1. Problem Definition. Assuming that the original input is x, the adversarial example x' is crafted by adding a specific perturbation, and x' is used as the input of the object detection model to cheat the object detector. Unlike classification, the target of the attack object detector is to make the model misjudge the input's category or the IoU (intersection over union) value between the predicted position coordinates and the real label lower than the threshold. The specific process can be expressed as follows:

$$\begin{aligned} &\delta = x' - x, \\ &\text{minmize } \|\delta\|_p, \\ &\text{subjected.to. } C_i' \neq C_i \text{ or IoU}\left(b_i', b_i\right) < \alpha, \end{aligned} \tag{1}$$

where p can be equal to 0, 1, 2, and ∞. In most methods, p is taken as 2, and the L_2 norm is used to constrain the perturbation. The predicted output of the original input x is $B(x) = (C_i, b_i)$, and the predicted output of the adversarial example x' is $B(x') = (C_i', b_i')$. C_i' and C_i are the predicted categories of x' and x, respectively, and b_i' and b_i are the predicted boxes of x' and x, respectively. α is the detection threshold. In this paper, we set $\alpha = 0.5$, consistent with the target faster RCNN model.

3.2. Positioning Module. The attack performance of adversarial examples is highly related to the region of interest of the attacked network. Adding the perturbation to the prediction-sensitive region of the attacked network can achieve the effect of the adversarial attack, and the perturbation added to the background region is redundant. Grad-CAM is a gradient-weighted class activation mapping method to locate the important regions related to network prediction. Therefore, we combined this visual attention technology to design a positioning module and predict key attack regions and limit perturbation-added regions to avoid the redundant perturbation in the background on the premise of ensuring the attack effect.

The original Grad-CAM method is mainly used for classification networks, which are unsuitable for object detection models. Therefore, we modify this method according to the characteristics of the detection model. We extract region proposals from the RPN network after nonmaximum suppression processing and calculate their weight values, respectively, using gradient-weighted class activation mapping:

$$\alpha_{\text{if}} = \frac{1}{\left(x_{i2} - x_{i1}\right) \times \left(y_{i2} - y_{i1}\right)} \sum_{j=x_{i1}}^{x_{i2}} \sum_{k=y_{i1}}^{y_{i2}} \frac{\partial y_i}{\partial A_{ijk}^f}, \tag{2}$$

where α_{if} is the weight of the f-th channel of the feature map A to the i-th region proposal, y_i is the probability of containing objects in the i-th region proposal, and x_{i1}, x_{i2}, y_{i1}, and y_{i2} are the position coordinates of the i-th region proposal.

Second, the weight values of region proposals are processed by the ReLU function and accumulated as follows:

$$m = \sum_{g_i \in R} \text{ReLU}\left(\sum_f \alpha_{\text{if}} A_i^f\right), \tag{3}$$

where $R = \{g_1, g_2, \ldots, g_n\}$ is the set of region proposals in RPN after nonmaximum suppression, g_i is the i-th region proposal, and A_i^f is the feature map of the i-th region proposal mapped to the f-th channel of feature map A.

Finally, the attack region mask for adding perturbations is formed after normalization:

$$M = \frac{m - m_{\min}}{m_{\max} - m_{\min}}, \tag{4}$$

where M is the final generated attack region mask.

As shown in Figure 2, the first row is the original image and the second row is the visualization of the attack region in the original image. It can be seen in Figure 2 that the positioning module can accurately capture the key regions in the image that affect the performance of object detection and generate the perturbation-added mask. By adding the perturbation to the sensitive region concerned by model prediction, we reduce the perturbation in the irrelevant background region, making the generated adversarial examples closer to real images and improving the visual effect.

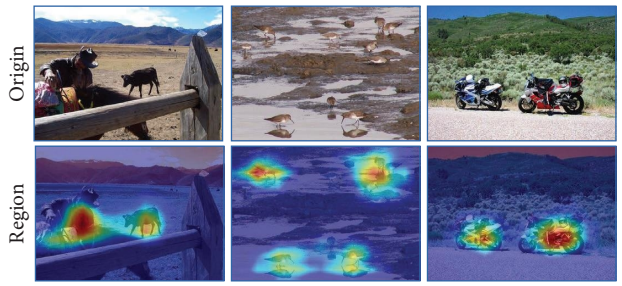


Figure 2: Attack regions.

3.3. Perceptual Loss. Gatys et al. [33] found that the pretrained deep convolutional network can effectively extract the content representation from the image. With the progressive processing level of the network, the input image is gradually converted into the content representation of the image. The low-level layers of CNN extract low-level features such as points and lines, and high-level layers tend to capture high-level semantic information. Different convolution kernels of the same convolutional layer extract different image features.

Based on this phenomenon, we designed the perceptual loss by constraining the content representation difference between original images and adversarial examples. We input the original image and the adversarial example into the pretrained convolutional neural network, respectively, and extract feature maps from different convolutional layers. Because feature maps extracted from different convolution kernels of the same convolutional layer differ, we average each layer's features in the spatial dimension, then sum them on the channel dimension, and calculate the L_2 distance between the content representation of the two as the perceptual loss for constraint perturbation generation. We choose the pretrained VGG16 [34] model as the convolutional neural network in the experiment. The perceptual loss is as follows:

$$L_{\text{perceptual}} = \sum_{f \in L} \sum_{f_l} \frac{1}{H_{f_l} \times W_{f_l}} \sum_{h=1}^{H_{f_l}} \sum_{w=1}^{W_{f_l}} \left\| \left(x_{\text{hw}}^{f_l} - y_{\text{hw}}^{f_l} \right) \right\|_2^2. \quad (5)$$

In Equation (5), H_{fl} and W_{fl} are the dimensions of the l-th channel of the f layer's feature map. $x_{\text{hw}}^{\text{fl}}$ and $y_{\text{hw}}^{\text{fl}}$ are the values of row h and column w of the l-th channel of the f layer's feature map output by the pretrained convolutional network with original inputs and adversarial examples. We set f as conv1-1, conv2-1, conv3-1, conv4-1, and conv5-1.

As shown in Figure 3(a), the first row is the perturbation generated by UEA that uses the L_p norm constraint. It can be seen that there are many irregular texture structures. These irregular texture structures heavily degrade the imperceptibility of adversarial examples. The second row is the perturbation generated by the WPAE method that uses the perceptual loss. Unlike using the L_p norm to constrain the pixel value difference between adversarial examples and original images, our perceptual loss can constrain the content representation difference between adversarial examples and original images and effectively reduce the subjective difference between adversarial examples and original images.

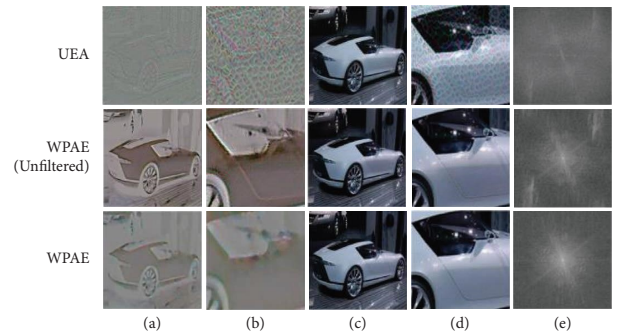


Figure 3: The perturbations generated by UEA and WPAE (unfiltered) have some abnormal red noise, and there is also some red noise in adversarial examples generated by the two methods. Red noise is basically eliminated in adversarial examples generated by WPAE. (e) The spectrum of perturbations. (a) Perturbation. (b) Detail. (c) Adv. (d) Detail. (e) Spectrum.

As shown in Figure 3, the imperceptibility of adversarial examples is significantly improved using the above perceptual loss. However, restricted by the generator of GAN, there are always some distortions in the generated perturbation. The red noise in the perturbation is a pixel with sharp changes in image intensity, which belongs to high-frequency noise. The human eye is more sensitive to the changes in high-frequency components than those in low-frequency components in the image. Therefore, filtering out the abnormal noise at high frequency and concentrating the perturbation on the low-frequency component can reduce the human eye's sensitivity to image changes and improve the subjective visual effect. In addition, the research in literature [30–32] has proved that filtering out high-frequency noise, which changes violently in the perturbation, and retaining the low-frequency perturbation will not affect the attack ability of the adversarial example. Therefore, we perform a postprocessing method based on median filtering to eliminate this red noise.

The median filter is a low-pass filter that can effectively eliminate isolated bright spots (dark spots), suppress noise, and retain edge contour information and image details. The red noise in the perturbation is similar to an isolated bright spot in the image. Therefore, we use the principle of median filtering to smooth the abnormal noise in the perturbation and replace the original pixel value with the median pixel value of eight adjacent points around it. The specific operation is as follows:

$$g(x, y) = \text{Med}\left[g(x-1, y-1), g(x, y-1), g(x+1, y-1), g(x-1, y), g(x+1, y), g(x-1, y+1), g(x, y+1), g(x+1, y+1) \right], \quad (6)$$

where $g(x, y)$ is the intensity value of the pixel at coordinates (x, y) after noise reduction and Med is the median function, which outputs the input's median.

As shown in Figure 3, the red noise is basically eliminated after smoothing and filtering the perturbation generated by WPAE, and there is no abrupt red noise in the final generated adversarial example, which significantly improves the subjective visual effect compared with the adversarial example without smoothing.

We obtain the spectrum of perturbations after the Fourier transform and shift processing to verify the influence of filtering perturbations on the attack effect. The specific operation is shown in the equations as follows:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M+vy/N)}, \tag{7}$$

$$F(u, v) = F\left(u - \frac{M}{2}, v - \frac{N}{2}\right), \tag{8}$$

where $F(u, v)$ is the value in the frequency domain (u, v), $f(x, y)$ is the pixel value in the spatial domain (x, y), and M and N are the sizes of the image.

As shown in Figure 3(e), the center point of the spectrum is the lowest frequency point, and the points on different radii represent different frequencies. The more outward the point is, the higher the frequency. After the perturbations generated by WPAE are smoothed and filtered, we can see that some abnormal noise in high frequency is filtered out, but its low-frequency information does not change, which can maintain the attack ability. We describe the specific changes in attack ability in Section 4.3.

3.4. Weakly Perceptual Adversarial Example. Combined with the ideas introduced in the above two sections, we propose the WPAE method, and its framework is shown in Figure 4. WPAE adopts the GAN structure consistent with AdvGAN [35], which is a relatively standard and straightforward GAN. We add the above ideas to this GAN to verify the effect of the above ideas on improving the imperceptibility of adversarial examples generated by GAN-based object detection attack methods.

The attack method uses the generator and the discriminator to train and update in turn. In the first stage, the generator is trained and the parameters of the discriminator are fixed. First, the original image is input into the generator and the positioning module to obtain the perturbation and attack region mask, respectively. Then, the perturbation is smoothed, and the Hadamard product with the attack region mask is performed to get the final perturbation, which is added to the original image to form the adversarial example. Finally, the adversarial example is inputted into the target model and the pretrained convolutional network to calculate the loss and update the parameters of the generator. In the second stage, the discriminator is trained and the parameters of the generator are fixed.

The Adam optimization method is used in WPAE. The specific process is shown in Algorithm 1. In terms of the loss function, in addition to the perceptual loss, we introduce the DAG loss and feature map loss to UEA to improve its attack ability. In summary, the loss function of the WPAE algorithm consists of the GAN loss, perceptual loss, feature map loss, and DAG loss:

$$L = L_{\text{GAN}} + \alpha L_{\text{perceptual}} + \beta L_{\text{feature}} + \gamma L_{\text{DAG}}, \tag{9}$$

where α, β, *and* γ are the weights of the perceptual loss, feature map loss, and DAG loss, respectively, which are used to balance losses. In the experiment, we set $\alpha = 1000$, $\beta = [0.0001, 0.0002]$, and $\gamma = 1$.

4. Experimental Results

4.1. Datasets and Evaluation Indicators. We choose VOC2007, which is commonly used in object detection, to verify the effectiveness of our method. The VOC2007 dataset contains 9963 images of 20 categories. The performance is evaluated by perceptual indicators and attack success rates.

4.1.1. Perceptual Indicators. To evaluate the perceptual indicators of adversarial examples, we use four indicators: the mean square error (MSE), peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and learned perceptual image patch similarity (LPIPS [36]). The PSNR is one of the most widely used objective image evaluation indicators, and it is based on the error between corresponding pixels but does not consider the visual recognition and perceptual characteristics of the human eye, so the PSNR often leads to different evaluation results from subjective perspectives. SSIM compares the brightness, contrast, and structure between the adversarial example and the real image. The closer the value is to 1, the more similar the structure between them. Both the PSNR and SSIM are simply compared at a low level, which still lags behind people's reality perception. LPIPS is a perceptual loss indicator closer to human perceptual behavior. The lower the value, the more similar the two images.

4.1.2. Attack Success Rate. The commonly used performance evaluation indicator of object detection is the mean average precision (mAP). Therefore, we take the mAP decline degree of the attacked detection model, namely, the attack success rate (ASR), as the performance indicator to evaluate the attack algorithm. The higher the ASR value, the greater the decrease in the mAP value of the object detection model and the better the attack algorithm's performance. We define the attack rate as follows:

$$\text{ASR} = 1 - \frac{\text{mAP}_{\text{adv}}}{\text{mAP}_{\text{ori}}}. \tag{10}$$

Among them, mAP_{adv} is the mAP value of the detection model when the adversarial example is input, mAP_{ori} is the mAP value of the detection model when the original image is input, and the range of the ASR value is between 0 and 1.

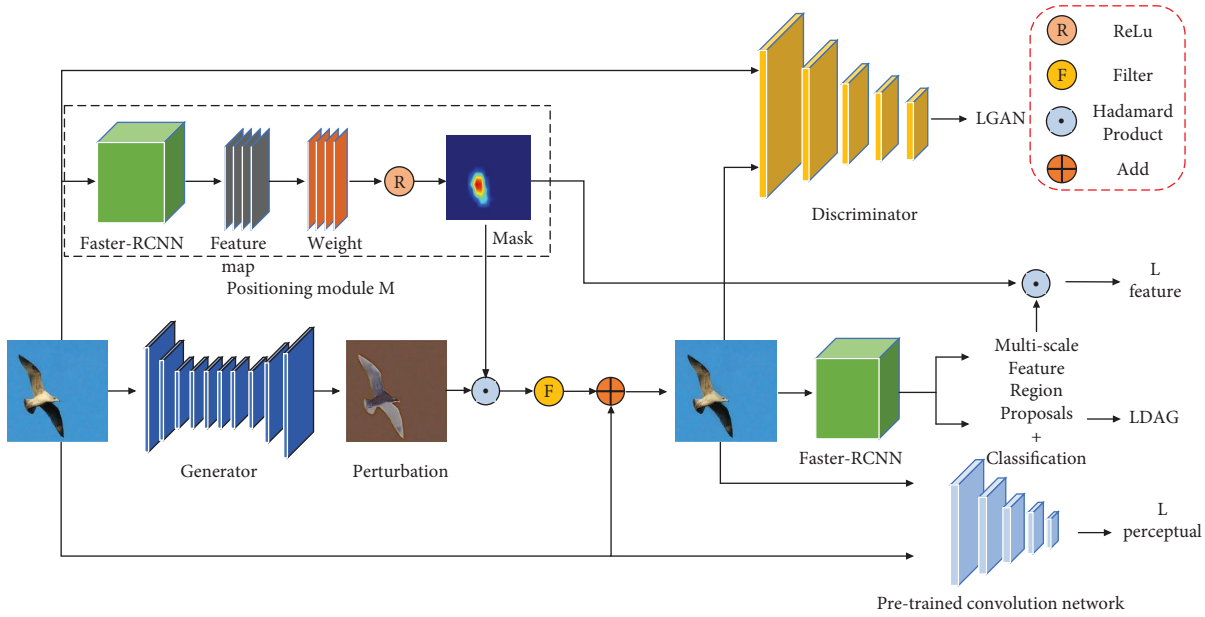


Figure 4: The framework of our WPAE consists of a generator, a discriminator, a filter, and a positioning module (M), attacking the target faster RCNN model. The pretrained convolutional network is used to extract the content representation from the image.

4.2. Implementation Details. The algorithm is implemented in the deep learning framework of Pytorch 1.7.1. The workstation is configured as NVIDIA RTX 2080ti 12G, the batch is set to 1, the number of iterations is 9, the image size is 300×300, the initial learning rate is set to 0.1, and the adaptive moment estimation (Adam) optimization method is used. The learning rate attenuation coefficient is 0.001, and the IoU threshold for mAP calculation is 0.5.

4.3. Analysis of Experimental Results. In this section, we conduct comparative experiments on the perceptibility and attack ability of the adversarial examples generated by WPAE and ablation experiments to verify the effects of our designed positioning module, perceptual loss, and smoothing filter on the perceptibility and attack ability.

4.3.1. Visual Perceptual Experiment

(1) Comparison of Subjective Visual Effects. As shown in Figure 5, we found that the subjective visual effect of the adversarial example generated by WPAE is greatly improved compared with the UEA method. The adversarial example of UEA has a visible perturbation in the foreground region, which is quite different from the original image. The subjective visual effect of the adversarial example generated by WPAE is equal to that of the DAG method based on optimization. It is difficult to find the difference between the adversarial example and the original image under naked eye observation, which dramatically improves the image quality of the adversarial example generated by GAN-based attack methods.

To better show the differences in the details of the adversarial examples generated by different methods, we magnify the local details of adversarial examples. We can clearly observe the disturbing texture of its surface in the UEA method, which is quite different from the original image. Compared with UEA, the subjective visual effect of the adversarial examples generated by WPAE has also been greatly improved in local details, and it is not easy to find the difference between the adversarial example and the original image under naked eye observation.

We further designed a subjective experiment to test the subjective visual effect of adversarial examples. We invited ten people who had never browsed the experimental datasets to classify images and adversarial examples. We randomly select 20 images from the testing dataset of VOC2007 and generate the corresponding adversarial examples used by DAG, UEA, and WPAE as experimental images. Ten experimenters were asked to classify the 20 groups of images, respectively. During the experiment, they scanned and confirmed each image by themselves and judged whether the image was a real image or an adversarial example. In the same group, multiple images can be judged as real images. The results are shown in Table 1.

According to the experimental results, experimenters cannot find the difference between the real image and the adversarial example generated by WPAE and DAG under the subjective observation of human eyes. Therefore, 96.5% of the experimenters mistakenly judged the adversarial example generated by WPAE for the real image, and 96% mistakenly judged the adversarial example generated by DAG for the real image. However, there are apparent differences between the adversarial examples generated by UEA and real images, and all ten experimenters successfully identified 20 adversarial examples generated by UEA. The subjective experimental results effectively verify that the perceptual performance of the adversarial example generated by WPAE is much better than that of UEA. It is difficult to judge the difference between the adversarial example generated by WPAE and the real image seen by human eyes.

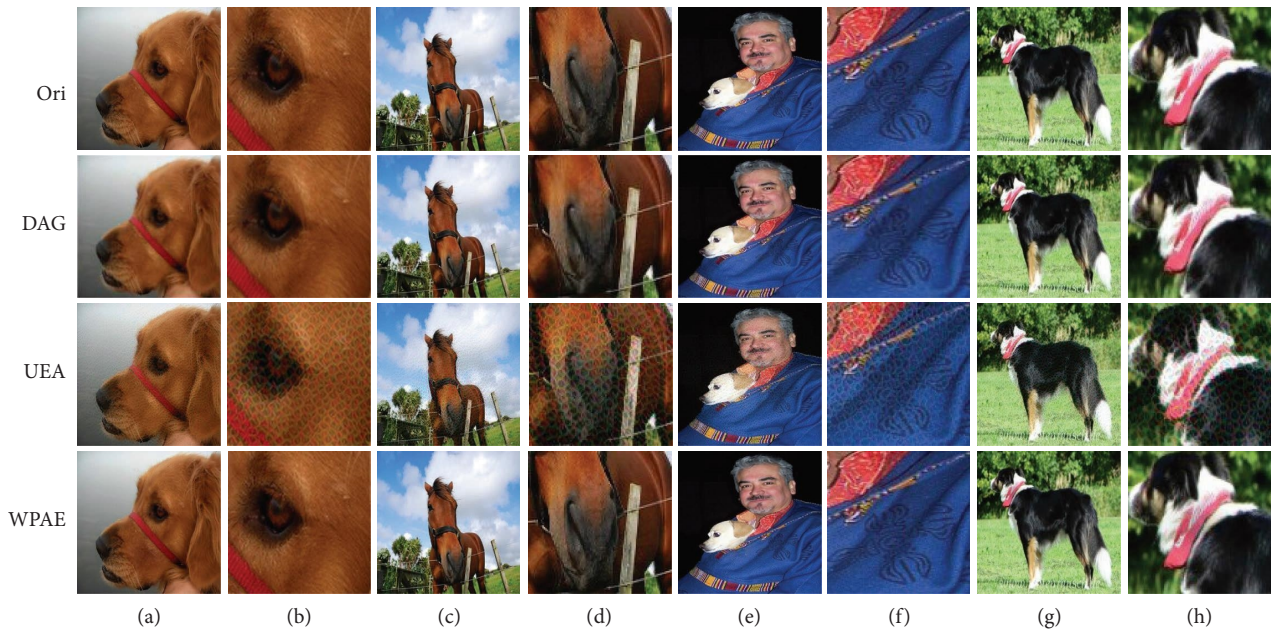


Figure 5: Comparison of adversarial examples.

Initialization: T = Faster RCNN, $f = R_f$, dataset = VOC 2007, batchsize = 1, epoch = 5, $\alpha = 1000$, $\beta = [0.0001, 0.0002]$, and $\gamma = 1$
While $t <$ epoch:
  for x **in** the dataset:
    Train G:
    $G(x) \longrightarrow r$, $M(x) \longrightarrow Mp = r \odot M$,
    $F(p) \longrightarrow P$
    $x' = x + P$
    $T(x') \longrightarrow f'$
    $G(x, p) \longrightarrow L_{GAN\_G}$
    Calculate $L_{perceptual}(x, x')$, $L_{feature}(f, f')$, and L_{DAG}
    $L_G = L_{GAN_G} + \alpha L_{perceptual} + \beta L_{feature} + \gamma L_{DA}$G
    Update G parameters
    Train D:
    D (img_real, ones) $\longrightarrow$ loss_real
    D (img_fake, zeros) $\longrightarrow$ loss_ fake
    $L_D = (\text{loss\_real} + \text{loss\_ fake}) * 0.5$
    Update D parameters
  end for
  $t \longrightarrow t{+}{+}$
end while

Algorithm 1: WPAE algorithm.

(2) Comparison of Objective Indicators. To further verify the perceptual performance of the adversarial examples generated by WPAE, we calculated the MSE, PSNR, SSIM, and LPIPS indicators of DAG, UEA, and WPAE, respectively. The results are shown in Table 2.

According to Table 2, the SSIM indicator of the three methods is above 0.999, basically consistent with the original image's structure. Regarding MSE and PSNR, WPAE is better than the other two methods. It has the slightest change in the original image, and the difference between the pixels in the adversarial examples generated by WPAE and the pixels in the original image is the smallest. These two of DAG are the worst because the DAG's adversarial example has been optimized for hundreds of iterations, and the final changed pixel value is quite different from the original image. In the LPIPS indicator, WPAE is the best and DAG is better than UEA. This result is also consistent with that shown in Figure 5. The subjective visual effect of WPAE and

Table 1: Subjective experimental results.

Method	Origin	DAG	UEA	WPAE
Real image	195	192	0	193
Adversarial example	5	8	200	7

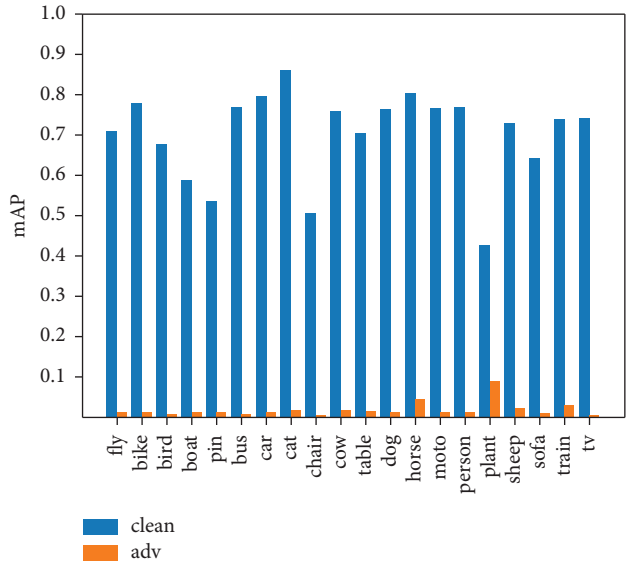


Figure 6: mAP of each category.

Table 2: Comparison of indicators for each method.

Method	MSE	PSNR	SSIM	LPIPS
DAG	970	19.4	0.999	0.311
UEA	114	27.9	0.999	0.353
WPAE	74	30.3	0.999	0.126

DAG is better than that of UEA, proving that MSE and PSNR indicators cannot accurately reflect the perceptual performance of adversarial examples. It also reflects that using the L_p norm distance to constrain the generation of adversarial examples cannot effectively improve the subjective visual effect.

4.3.2. Attack Experiment. We use WPAE to attack the target faster RCNN model in order to verify its attack ability. Figure 6 shows the mAP values of 20 categories detected by faster RCNN for the VOC2007 dataset, in which the blue bar is the original input and the red bar is the adversarial example. It can be seen that after the WPAE attack, the precision values of each category have been greatly reduced.

To further verify the attack capability of WPAE, we compare the attack performance with that of DAG and UEA. The comparative experimental results are shown in Table 3.

According to Table 3, the attack success rate of WPAE is the highest, reaching 0.97, which effectively fools the faster RCNN model. The results show that WPAE can improve the perceptual performance of adversarial examples while maintaining good attack capability.

4.3.3. Attack Experiment under Defense Method. In this section, we evaluate the proposed WPAE method against several image compression defense methods to verify the robustness of the proposed method.

Table 3: Comparison of the white box attack success rate.

Method	mAP_{ori}	mAP_{adv}	ASR
DAG	0.70	0.05	0.93
UEA	0.70	0.05	0.93
WPAE	0.70	0.02	0.97

Table 4: Attack performance under defense methods.

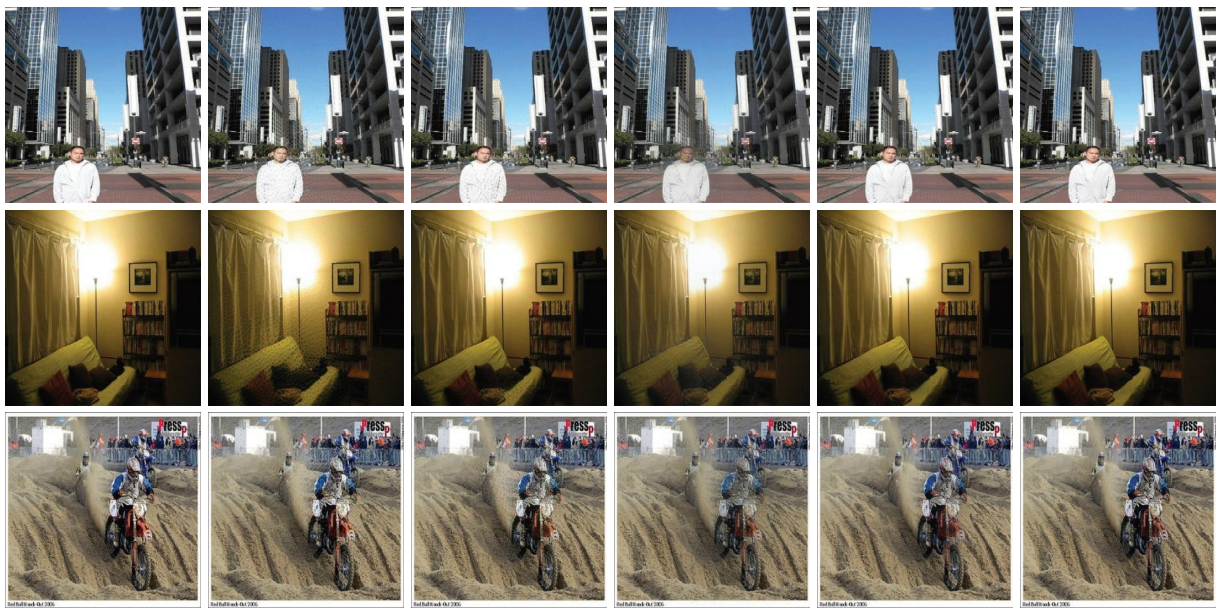
Method	mAP_{ori}	mAP_{adv}	ASR
No defense	0.70	0.018	0.974
JPEG20	0.70	0.024	0.966
JPEG50	0.70	0.025	0.964
JPEG80	0.70	0.027	0.961
Comdefense	0.70	0.031	0.956

We evaluate the performance of WPAE under JPEG compression with different quality factors to show the robustness of the adversarial example generated by WPAE. We used the Comdefense method to reconstruct the image and test the attack performance. The experimental results are shown in Table 4.

According to Table 4, after being compressed by the JPEG method with compression ratios of 20, 50, and 80 and processed by the Comdefense method, the ASR values of the adversarial example just have a slight decrement compared with those of the original image. Since the adversarial example generated by WPAE is processed by smooth filtering, there is almost no high-frequency perturbation, and the attack effect of WPAE depends more on low-frequency perturbation. Hence, WPAE shows strong robustness under the methods that reduce adversarial examples' attack ability by denoising.

4.3.4. Ablation Experiment. To verify the effect of the proposed positioning module, the perceptual loss, and the smooth filtering method on the enhancement of the imperceptibility of the adversarial example, we further conduct ablation experiments. The results are shown in Table 5.

According to Table 5, the MSE, PSNR, and LPIPS indicators of the positioning module added to the network are better than those of UEA. By limiting the scope and amplitude of perturbations, the imperceptibility of adversarial examples has been improved to a certain extent. However, using the L_2 loss to limit perturbation generation, the subjective visual effect still has room for improvement. When only using the perceptual loss, LPIPS increases more obviously. When only using the perceptual loss, the LPIPS increases obviously, and the MSE, PSNR, and SSIM indicators have an inevitable decline, which indicates that the perceptual loss can improve the subjective visual effect, but cannot limit the perturbation added region. When the positioning module and the perceptual loss are added simultaneously, perceptual indicators and subjective visual effects are improved compared with adding the positioning module or optimizing the perceptual loss alone. It further verifies the effectiveness of the proposed method in



(a) (b) (c) (d) (e) (f)

Figure 7: Adversarial examples generated by each comparison method. (a) Original. (b) UAE. (c) +M. (d) +P. (e) +P+M. (f) WPAE.

Table 5: Comparison of various indicators in the ablation experiment.

Method	MSE	PSNR	SSIM	LPIPS	ASR
UEA	114	27.9	0.999	0.353	0.93
Perceptual module	87.6	29.3	0.999	0.184	0.97
Perceptual loss	362.9	23.1	0.980	0.155	0.97
Perceptual module + loss	85.3	29.6	0.999	0.130	0.97
Perceptual module + loss + filtering	74	30.3	0.999	0.126	0.97

enhancing the imperceptibility of adversarial examples. After further filtering perturbation and the subjective visual effect, perceptual indicators have achieved the best result on the premise of ensuring the attack success rate, effectively improving the imperceptibility of adversarial examples.

Figure 7 shows adversarial examples generated by different comparison methods. According to Figure 7(b), the subjective visual effect of adversarial examples generated by UEA is poor, and many obvious perturbations are added to irrelevant background regions. In Figure 7(c), after adding the positioning module, the perturbation is mainly concentrated in the key regions, and the range and amplitude of perturbations are well controlled, but the added perturbation texture is still relatively abrupt. In Figure 7(d), after using the perceptual loss, the added perturbation texture is closer to the original image, which improves the imperceptibility of adversarial examples to a certain extent. Because it does not limit the range and amplitude of the perturbation, we can find many perturbations added to irrelevant background regions. As shown in Figure 7(e), when the positioning module and perceptual loss are used simultaneously, the subjective visual effect of adversarial examples has been greatly improved, but there is still some abnormal red noise. In Figure 7(f), we can see that the red noise in adversarial examples has been effectively eliminated after further using the smooth filtering method, and there is almost no difference between the adversarial examples and the original images under the observation of human eyes. The visual comparison results of adversarial examples generated by different methods also verify the conclusions of the above ablation experiment.

5. Conclusion

In this study, we first propose a positioning module, which is used to add perturbations to the region of interest of the model prediction, effectively reducing the addition of perturbations in unrelated background regions. Then, we design the perceptual loss to further constrain the amplitude and structure of perturbations, combined with the image content representation extracted from the DNN. Finally, we use a postprocessing method based on median filtering to smooth the perturbations and reduce the abrupt red noise. The experimental results for the VOC dataset demonstrate that the generated adversarial examples significantly improve the imperceptibility of the adversarial examples generated by GAN-based methods on the premise of maintaining the attack success rate and have strong robustness under the defense method of reducing the attack ability of the adversarial examples by denoising. In addition, simply integrating our proposed perceptual module, perceptual loss, and postprocessing method into any GAN-based object

detection attack method can improve the imperceptibility of the generated adversarial examples.

Data Availability

The data used to support the findings of this study are available at https://host.robots.ox.ac.uk/pascal/VOC/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Natural Science Foundation of Jiangsu Province (BK20180080) and the National Natural Science Foundation of China (62071484).

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Proc. Int. Conf. Neural Information Processing Systems, Lake Tahoe, Harrahs and Harveys*, pp. 1097–1105, December 2012.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the Proc. Int. Conf. Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[3] S. Ren, K. He, R. Girshick, J. Sun, and R.-C. N. N. Faster, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[4] Z. Cai and N. Vasconcelos, "Cascade R-CNN: delving into high quality object detection," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, pp. 6154–6162, Salt Lake City, UT, USA, June 2018.

[5] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot MultiBox detector," in *Proceedings of the Int. Conf. European Conference on Computer Vision*, pp. 21–37, October, 2016, Amsterdam, Netherlands.

[6] J. Redmon, S. Divvala, and R. Girshick, "You only look once: unified, real-time object detection," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, LV, USA, June 2016.

[7] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the Proc. Int. Conf. International Conference on Computer Vision*, pp. 2980–2988, Venice, Italy, October, 2017.

[8] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: fully convolutional one-stage object detection," in *Proceedings of the Int. Conf. International Conference on Computer Vision*, pp. 9626–9635, Seoul, Korea, October, 2019.

[9] H. Law and J. Deng, "Cornernet: detecting objects as paired key points," in *Proceedings of the Int. Conf. European Conference on Computer Vision*, pp. 734–750, Munich, Germany, October, 2018.

[10] N. Kayhan and S. Fekri-Ershad, "Content based image retrieval based on weighted fusion of texture and color features derived from modified local binary patterns and local neighborhood difference patterns," *Multimedia Tools and Applications*, vol. 80, no. 21-23, pp. 32763–32790, 2021.

[11] E. Jarallah and M. Al-Saffar, "Isolation and characterization of lytic bacteriophages infecting *Pseudomonas aeruginosa* from sewage water," *Journal of Innovation in Health Informatics*, vol. 9, no. 9, pp. 220–230, 2016.

[12] R. R. Selvaraju, M. Cogswell, and A. Das, "Grad-cam: visual explanations from deep networks via gradient-based localization," in *Proceedings of the Int. Conf. International Conference on Computer Vision*, pp. 618–626, Venice, Italy, October, 2017.

[13] C. Szegedy, W. Zaremba, I. Sutskever et al., "Intriguing properties of neural networks," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, Banff, Canada, February 2014.

[14] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, San Diego, CA, USA, March 2015.

[15] A. KurakinI. Goodfellow et al., "Adversarial machine learning at scale," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, Toulon, France, February 2017.

[16] D. Yinpeng, L. Fangzhou, P. Tianyu et al., "Boosting adversarial attacks with momentum," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, pp. 9185–9193, Salt Lake City, USA, June, 2018.

[17] L. Yanpei, C. Xinyun, L. Chang, and S. Dawn, "Delving into transferable adversarial examples and black-box attacks," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, Toulon, France, February 2017.

[18] A. Madry, A. Makelov, and L. Schmidt, "Towards deep learning models resistant to adversarial attacks," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, Toulon, France, February 2017.

[19] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proceedings of the Int. Conf. Proc of the IEEE Symp on Security and Privacy (EuroS&P)*, pp. 39–57, San Jose, CA, USA, April, 2017.

[20] M. S. Moosavi, A. Fawzi, and P. Frossard, "DeepFool: a simple and accurate method to fool deep neural networks," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, pp. 2574–2582, Las Vegas, LV, USA, June, 2016.

[21] X. Cihang, W. Jianyu, and Z. Zhishuai, "Adversarial examples for semantic segmentation and object detection," in *Proceedings of the Int. Conf. International Conference on Computer Vision*, pp. 1369–1378, Venice, Italy, October, 2017.

[22] L. Yuezun, T. Daniel, and C. MingChing, "Robust adversarial perturbation on deep proposal-based models," in *Proceedings of the Int. Conf. British Machine Vision Conference*, BMVC, Newcastle, UK, November 2018.

[23] Q. Liao, X. Wang, B. Kong et al., "Category-wise Attack: Transferable Adversarial Examples for Anchor Free Object Detection," 2003, http://arxiv.org/abs/2003.04367.

[24] W. Derui, L. Chaoran, W. Sheng, Q. L. Han, S. Nepal, and X. Zhang, "Daedalus: breaking nonmaximum suppression in object detection via adversarial examples," *IEEE Transactions on Cybernetics*, vol. 51, no. 1, pp. 1–14, 2021.

[25] W. Xingxing, L. Siyuan, and C. Ning, "Transferable adversarial attacks for image and video object detection," in *Proceedings of the Int. Conf. 28th International Joint Conference on Artificial Intelligence*, pp. 954–960, Macao, China, August, 2019.

[26] L. Yiwei, X. Guoliang, and L. F. A. Wanlin, "A fast method to attack real-time object detection systems," in *Proceedings of*

the Int. Conf. International Conference on Communications, Chengdu, China, December, 2020.

[27] D. Nilaksh, S. Madhuri, and C. Shang-Tse, "Shield: fast, practical defense and vaccination for deep learning using jpeg compression," in *Proceedings of the Int. Conf. 24th ACM SIGKDD International Conference*, pp. 196–204, London, UK, August, 2018.

[28] G. Chuan, R. Mayank, and C. Moustapha, "Countering adversarial images using input transformations," in *Proceedings of the Int. Conf. Int Conf on Learning Representations*, ICLR, Toulon, France, January 2017.

[29] J. Xiaojun, W. Xingxing, and C. Xiaochun, "ComDefend: an efficient image compression model to defend adversarial examples," in *Proceedings of the Int. Conf. Conference on Computer Vision and Pattern Recognition*, pp. 6077–6085, Long Beach, CA, USA, June, 2020.

[30] R. Duan, Y. Chen, and D. Niu, "AdvDrop: adversarial attack to DNNs by dropping information," in *Proceedings of the Int. Conf. International Conference on Computer Vision*, pp. 7486–7495, Montreal, Canada, October, 2021.

[31] Y. Sharma, G. W. Ding, and M. Brubaker, "On the Effectiveness of Low Frequency Perturbations," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 3389–3396, Macao, China, August, 2019.

[32] S. Song, Y. Chen, and N. M. Cheung, "Defense against adversarial attacks with saak transform," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June, 2018.

[33] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, Las Vegas, LV, USA, June, 2016.

[34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the Int. Conf. Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June, 2014.

[35] C. Xiao, B. Li, and J. Y. Zhu, "Generating adversarial examples with adversarial networks," in *Proceedings of the Int. Conf. 27th International Joint Conference on Artificial Intelligence*, pp. 3905–3911, Stockholm, Sweden, July, 2018.

[36] R. Zhang, P. Isola, and A. A. Efros, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the Int. Conf. Computer Vision and Pattern Recognition*, pp. 586–595, Salt Lake City, UT, USA, June, 2018.