Yang *et al. Journal of Cloud Computing* (2023) 12:169
https://doi.org/10.1186/s13677-023-00534-3

Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

# RPU-PVB: robust object detection based on a unified metric perspective with bilinear interpolation

Hao Yang[1], Xuewei Wang[2], Yuling Chen[1*], Hui Dou[1] and Yangwen Zhang[1]

**Abstract**

With the development of cloud computing and deep learning, an increasing number of artificial intelligence models have been applied to reality. Such as videos on cell phones can be uploaded to the cloud for storage, which is detected by cloud arithmetic. Nevertheless, achieving this goal requires frequent consideration of the security of the model, since videos or images that go to the cloud, it is very likely to receive an adversarial attack. Regarding object detection, there has however been slow advancement in robustness research in this area. This is because training a target detection model requires a lot of arithmetic and time. Moreover, the current research has only slightly reduced the gap between clean and adversarial samples. To alleviate this problem, we propose a uniform perspective object detection robustness model based on bilinear interpolation that can accurately identify clean and adversarial samples. We propose the robustness optimization based on uniform metric perspective (RPU) for feature learning of clean and adversarial samples, drawing on the fine-grained idea. Following this, we analyze the fragility of the adversarial samples and consequently use the proposed perturbation filtering verification (PVB) based on bilinear interpolation. With slightly degraded clean sample detection performance, it substantially improves the robustness of object detection. Extensive experiments on PASCAL VOC and MS COCO show that our model guarantees the detection performance of clean samples and increases the detection performance of adversarial samples. The work we did has been open-sourced on GitHub: https://github.com/KujouRiu/RPU-PVB.

## Introduction

Object detection [1], which is a core task in the field of computer vision, aims at identifying specific targets in images or videos and determining their locations. With the rapid development of deep learning, object detection has made significant progress and has been widely applied. Cloud computing [2], with a powerful way of centralized management of computing and storage resources, provides strong support for the implementation and application of object detection. While cloud computing provides powerful computing support for object detection, security [3] is also a concern. Adversarial perturbation is a perturbation attack based on the gradient of features generated by the image entry model. Moreover, the attacker can target the attack model accurately and imperceptibly. In other words, it is difficult to make correct predictions for this attack model where the attacker already knows the model structure or generates migratory counterattack samples. Currently, adversarial defense for image classification has received a lot of attention, and as a result, a large number of adversarial defense models [4, 5] have been created. There is a lack of research in this area for object detection tasks. This is because the datasets for object detection are rich in variety and complex in context, such as PASCAL VOC

*Correspondence:
Yuling Chen
ylchen3@gzu.edu.cn
[1] Guizhou University, State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guiyang 550000, China
[2] Weifang University of Science and Technology, College of Computer Science and Technology, Weifang 261000, China

[6], and Microsoft Common Objects in Context (MS COCO) [7]. Therefore, training robust object detection models also costs more resources than image classification models. In the blockchain arena [8], a great deal of security research [9–12] has been conducted. The traffic prediction for example uses blockchain [13, 14] to secure the model, and blockchain is used to evaluate the more sensitive reputation data in industry [15] to ensure fairness and security. For edge computing [16], there is also a great deal of research [17] to ensure security.

Although research on the defense of object detection models has progressed slowly, some scholars are trying hard for it. MTD [18] improves the robustness of object detectors against different types of attacks by generalizing the adversarial training framework from classification to detection. Meanwhile, CWAT [19] generates more reasonable adversarial samples by balancing the selection of attack categories. RobustDet [20] points out the drawbacks of the first two approaches, namely, introducing adversarial samples into training makes the model choose and compromise between the accuracy of clean samples and that of adversarial samples. A robust detection model with adversarial-aware convolution is proposed to learn the robust features of both clean and adversarial images, thus greatly improving the detection ability of the model for adversarial samples.

Throughout this paper, we first build a reasonable loss by controlling the distance before clean samples and adversarial samples, drawing on the study of loss functions in fine-grained classification. In detail, fine-grained image classification is based on distinguishing the basic classes and performing finer subclasses. Whereas, for the adversarial samples, the features extracted by the model are used to build a finer division between adversarial and normal samples, which is used to guide the model to generate dynamic convolution parameters with different weights. The distance features are established by aggregating the features in the samples with the cosine similarity between the clean and adversarial samples, which are used as supervisory information to control the model optimization parameters by minimizing the distance features of positive samples while maximizing the distance features of negative samples to further generate more reasonable weights. Furthermore, we propose a reconstructed image method based on bilinear interpolation with perturbation filtering verification, by which we enhance the correct labeling features of the image by sampling and filtering out the attack perturbations of the image to obtain a more robust object detection model. With the combination of the two methods, we obtain the lowest performance gap, as shown in Fig. 1 . Extensive experiments on PASCAL VOC and MS-COCO datasets demonstrate that our proposed method has excellent detection capability and also achieves a high level of performance in dealing with attacks from different adversarial samples.

- We propose a metric-based dynamic convolutional parameter update method by targeting fine-grained studies to migrate to a model for object detection against the defense, by measuring the distance between clean and adversarial samples by the feature gradient of the input image, which greatly improves the robustness of the object detection model.
- By analyzing the correlation between the pixels of the adversarial samples, we provide a basis for enhancing the adversarial robustness by analyzing the fragility of the adversarial samples.
- With a reconstruction image method based on bilinear interpolation for perturbation filtering

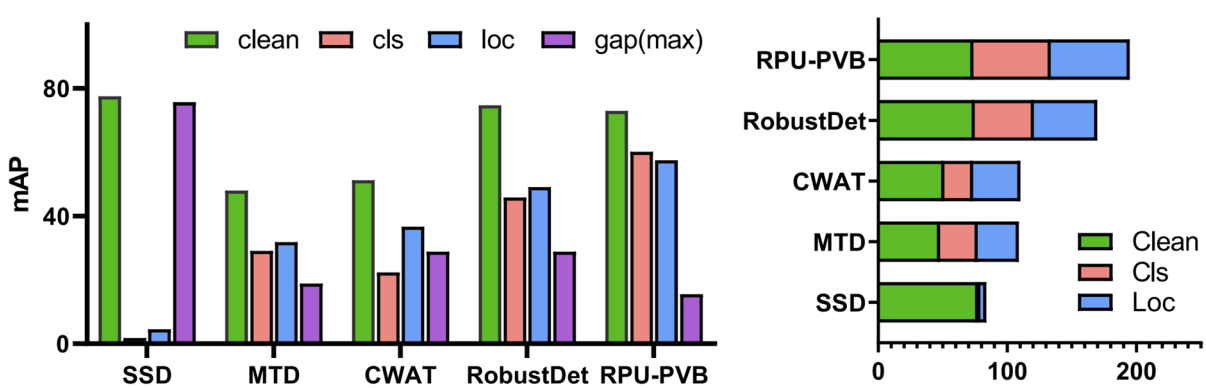


**Fig. 1** The left shows a comparison of the model's detection performance when subjected to different samples (clean vs. adversarial), in which we can see that there is almost no degradation in our accuracy for clean samples. In addition, we achieve the best detection performance against adversarial samples, realizing the smallest performance gap (highest vs. lowest performance difference). While the right figure shows a more intuitive comparison of the sum of performance detection, which shows that our proposed method achieves the highest performance

verification, we propose to filter the well-designed adversarial perturbations by up and downsampling to obtain better robustness of the object detection model.

- Our experiments test the performance of the model on top of the PASCAL VOC and MS-COCO datasets, where not only the detection ability of the object detection model for clean samples is not significantly degraded, but also the detection ability of the model for adversarial samples is improved, and the state-of-the-art performance is achieved in comparison with previous methods.

## Related work

### Image classification adversarial offensive and defensive development

Since the creation of adversarial perturbations against deep neural networks, more and more attack methods have been investigated. For example, a white-box attack where the attacker knows the target model information, i.e., the input image is perturbed to a certain limit, forcing the gradient obtained by the model to rise, resulting in a larger loss function. Common white-box attacks such as I-FGSM [21], PGD [21], DeepFool [22] etc.

While for black box attacks, the attacker only needs to generate adversarial samples with generalization through one or more neural networks. These adversarial samples are powerful for different models, such as MI-FGSM [23], SI-NI-FGSM [24], $V^2$MHI-FGSM [25], etc. Precisely because of the large number of attack methods studied in depth, numerous adversarial defenses have been generated to fight against them. Mostly, the approaches force the model to learn the information about the features brought by the adversarial samples and thus reduce the influence of the adversarial samples on the network.

Adversarial Training (AT) [26] is a technique applied in machine learning with the idea of improving the robustness of a model by adding samples with perturbative properties to the model. This technique can effectively counter attacks and improve the reliability and security of the model. The technique was first proposed by Goodfellow et al. The main idea is to add some artificially generated adversarial samples to the training data, which can deceive the model and thus force it to learn a more robust feature representation. Adversarial training is effective against various types of attacks, including input perturbation, target misdirection, etc. Several developments in adversarial training include defenses based on generative adversarial networks (GANs) [26], label-free adversarial training [27], to further improve the robustness of models using methods such as self-supervised learning [28].

### Adversarial attack and defense development based on object detection

The development of object detection has received great attention and models with different structures have been proposed one after another. For example, based on the first stage YOLO [1], SSD [29], RetinaNet [30], etc., these models perform regression and classification prediction on the prediction frame directly, which has a strong advantage in speed. The most representative of the two-stage models is Fast R-CNN [31], which first extracts the candidate frames for secondary correction based on images and then performs output prediction. And the transformer-based approach introduces the attention mechanism into the field of object detection, hoping to incorporate relational information in the features and achieve feature enhancement. The most representative models are Relation Net [32], DETR [33], etc.

Even though scholars continue to improve and enhance the models, the security of the models cannot be ignored. Adversarial attacks against object detection have also been proposed, and these methods can attack the models efficiently. Such as DAG [34], UEA [35], and GLH [36], the attacks on the model are achieved by adding to the image through computational perturbation for the features. Whereas Dpatch [37] and AdvTexture [38] affect the model output by training patches and making the model predict incorrectly by modifying the pixel points of the patches. Each of these attack methods without exception defeats the model judgment.

In comparison to attack methods, defense methods for object detection have been slow to develop. Although various methods have been proposed to improve model robustness, these models only mitigate the weaknesses of the models. The MTD [18] adversarial defense method proposed by Zhang et al. uses a multi-task supervised source for adversarial training, treating object detection as multi-task learning to train parameters. The CWAT [19] proposed by Chen et al. improves the sample quality of adversarial training by generating a generic adversarial attack to simultaneously attack all of the targets, jointly maximizing the respective loss of each object to improve the sample quality of adversarial training. RobustDet [20], on the other hand, uses another idea to improve model robustness by first using triple loss de-supervised dynamic convolution to learn the features of clean and adversarial samples, followed by enhancing the detection of adversarial samples by the distance between the VAE [39] reconstructed image and the clean samples.

Regarding the above object detection adversarial defense methods, none of them analyze the nature of the features of the adversarial samples, we detail our stronger dynamic convolution loss to obtain a more robust model in Method section, and we detail the comparison of

experiments and ablation experiments in Experiment section. And in Visualization section we will understand our method from a visualization perspective, and finally, in Conclusions section we conclude and look ahead.

## Method

### Adversarial attack settings for object detection

The adversarial attack for object detection and the adversarial attack for image classification share the same principle of adding noise to maximize the loss. The difference is that an adversarial attack for object detection requires targeting multiple objects in the image at the same time and is substantially more challenging to attack. While the targets of the attack are classification attack($A_{cls}$) and boundary regression box attack($A_{loc}$). We want to obtain the classification probability $C_i = \{c_i^{bg}, c_i^1, c_i^2, \ldots, c_n^{bg}\}$, where $bg$ represents the probability of the background, as well as $c_n^i$ represents the class prediction of $n$ targets in the image, under the detector $f$ with parameter $\theta$. The next obtained localization prediction $B_i = \{b_i^x, b_i^y, b_i^w, b_i^h\}$ represent the $b_i^x$ and $b_i^y$ coordinates of the boundary regression box for object detection, while $b_i^w$,$b_i^h$ represent the length and width of the boundary regression box.

For the attack mode, it is defined as follows:

$$A_{cls}(x) = \arg\max_{x\in[0,255]} L_{cls}(f(x;\theta), \hat{C}_i), \qquad A_{loc}(x) = \arg\max_{x\in[0,255]} L_{cls}(f(x;\theta), \hat{B}_i), \tag{1}$$

Where $\hat{C}_i$ $\hat{B}_i$ represent the labeled classification and localization information of clean samples, which we want to maximize the classification loss or localization loss to attack the model, as the basic principle of object detection against attack.

### Overall framework

A high-performance object detection defense model is proposed to address the problem that the performance gap between clean and adversarial samples is too large for object detection. Using the idea of fine-grained classification, the feature extraction of an image is first seen as a binary classification problem. The weights of the dynamic convolution are supervised through a uniform metric ground perspective so that the features extracted by the model are as similar as possible in the adversarial and clean samples. Furthermore, we use bilinear interpolation perturbation filtering to reconstruct the image in the validation phase, removing unnecessary adversarial perturbation information in the sampling process:

During the training phase, the *RPU* (robust optimization based on the perspective of uniform metrics) module gives the images the supervised information, which is used by Resnet18 [40] to extract the feature information of the model and assign different feature weights to the dynamic convolution to ensure that the clean image and the adversarial image of the same label are eventually extracted with the same features. In the validation phase, to obtain a more robust defense model, we use the *PVB* (Perturbation filtering verification based on bilinear interpolation) module to filter the adversarial noise. The overall process is shown in Fig. 2.

### Robust optimization based on the perspective of uniform metrics (*RPU*)

While previous adversarial defenses usually use adversarial training to learn the features of the adversarial samples, such an approach affects the detection accuracy of clean samples with little robustness improvement. Consequently, most methods [18] seek a balance between the accuracy of clean samples and adversarial samples. On the contrary, we refer to the Adversarially-Aware Convolution proposed by RobustDet [20] because this method can use a network with dynamic weights to achieve the same detection results for both clean and adversarial samples, as shown in Fig. 3:

Obtaining various weights (Conv1, Conv2, etc.) corresponding to different convolutional kernels ($\pi 1$, $\pi 2$, etc.) from the feature extraction network, calculating the output of the function y by summing up the summarized feature information using triplet loss [41] as the supervisory information. Unfortunately, this loss function does an inability to carefully discriminate between adversarial samples and clean samples.

To address this challenge, we utilize the idea of comparative learning for fine-grained identification, which in this paper, for the first time, proposes a robust optimization method based on a uniform metric perspective, where the distances of samples of the same class (clean samples to clean samples and adversarial samples to adversarial samples) are considered as intraclass distances, whereas the distances of samples of different classes (clean samples to adversarial samples) are considered as interclass distances.

Circle loss [42] has been extensively used as a loss function for classification problems in areas such as face recognition. An objective of the circle loss function is to cluster data points in the same category into tight clusters, but also keep the distance between different categories as much as possible. Consequently, Circle loss is a loss function based on margin control, which introduces two parameters, margin, and radius, where the margin is used to control the distance between different samples in the same category, and radius is used to control the size of clusters. For the

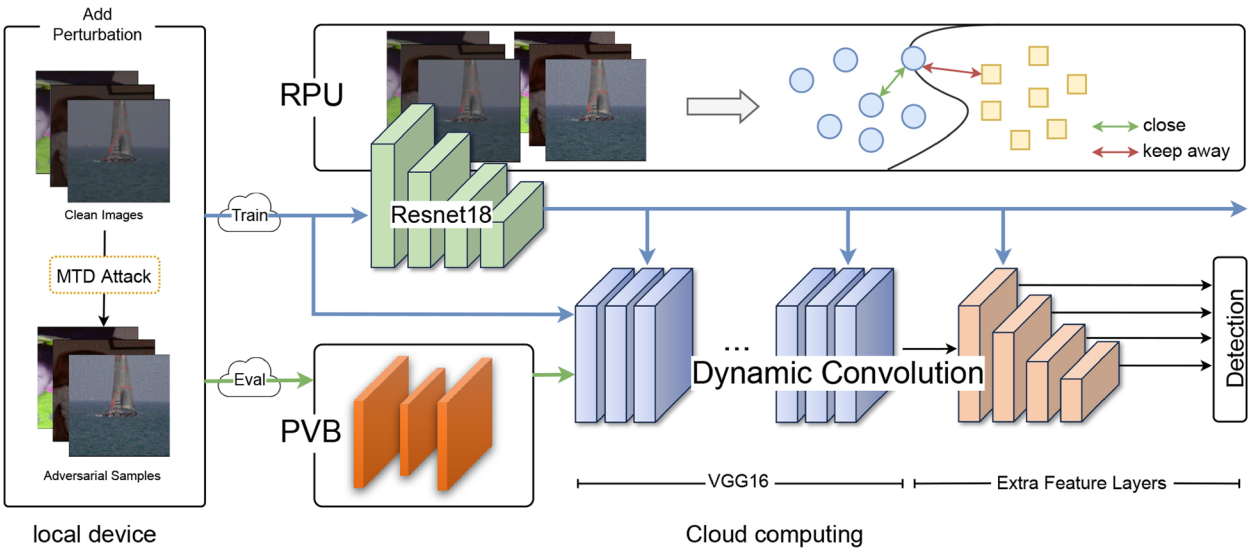


**Fig. 2** Overall structure: After adding perturbations to the local device, the input is fed into the cloud device for predicting the output results

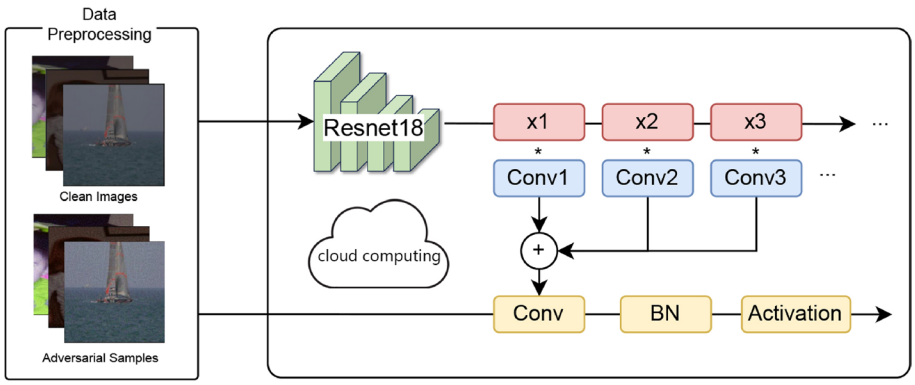


**Fig. 3** Dynamic Convolutional Computing Process in the Cloud

adversarial sample domain, we apply the circle loss function to both supervised clean samples and adversarial samples, so that the dynamic convolution of the defense model can correctly learn the features and assign different weights. Accordingly, our model can discriminate adversarial samples more carefully, thus improving the robust learning ability of the model.

The supervised loss function for dynamic convolution in the proposed method is shown as below:

$$L_{R_{cir}} = -\frac{1}{N_i}\Sigma_{y_i=p}^{N_i} log \frac{exp\{s(cos\theta_{ip}+m)\}}{exp\{s(cos\theta_{ip}+m)\}+\Sigma_{n=1,n\neq p}^{C} exp\{s(cos(\theta_{in})\}}, \tag{2}$$

where $s(cos\theta_{ip}+m)$, denotes intra-class robustness. $s(cos(\theta_{in})$, denotes inter-class robustness. Regarding the sample $i$, $N_i$ and $y_i$ are, separately, the respective sample size in the category in which the first $i$ sample is located, together with the category or location to which it belongs. $p$ represents the positive samples of the $i$th sample, whereas $n$ represents the negative samples of the $i$th sample, and thus satisfies $n \neq p$, which represents the total number of classes or locations, by calculating the intraclass cosine similarity, $\theta_{ip}$, and the interclass similarity, $\theta_{in}$, using the intraclass separation degree $m$ to controlling the distance between sample feature vectors. Simultaneously, $s$ is the feature scaling factor, which we compute using the default value.

As opposed to Triplet loss, the circle loss function introduces intra-class separation, which enforces the distance between samples of the same class through $m$ and increases the distance between samples of different classes through $s$. The circle loss function is a method for the network to learn the difference between samples of the same class and samples of a different class. Through

this method, the decision boundary between adversarial and clean samples is delineated, further strengthening the control of intra-class distance and enabling the network to learn their feature differences more accurately. The total loss function is as follows.

$$L = \beta(L_{det} + \alpha L_{R_{cir}}), \tag{3}$$

$L_{det}$ is the loss function of SSD, while $L_{R_{cir}}$ is the loss function we have applied. For $\alpha$ and $\beta$, we use the same parameters to compare RobustDet as more fairest

#### The fragility of the decision interval against the sample

The weakness of deep learning models is that adversarial samples may be intentionally created to perturb the input samples, which can produce incorrect classification results or misleading outputs. Adversarial samples are essentially gradient ascent processes, whereas the procedure of generating adversarial samples is incidentally the process of maximizing the model loss function. An example of an adversarial sample can be summarized as the min-max problem. Min-max refers to the attacker's attempt to find a perturbation to maximize the deception of the model, which expects to minimize the loss expectation of the entire data distribution in the case of adversarial samples, as shown in the following equation:

$$\min_{\theta} \rho(\theta), \rho(\theta) = \mathbb{E}(x, y)_{\sim D}[\max_{\theta \in S} L(\theta, x + \delta, y)], \tag{4}$$

Among them, $L$ denotes the loss function of the model, which is used to calculate the distance of the model output from the label. $x$ is the input image and $\delta$ is the additive perturbation vector. $y$ is the target label of the attack, which is the direction we have to mislead. Whereas $D$ is the distribution of the data $(x, y)$, $\theta$ is the network parameters of the model. To minimize the human eye's recognition of the adversarial samples, the attacker wants the perturbation $\delta$ to be in the $S$ range.

Consequently, we can consider the adversarial sample as a carefully computed image; in other words, the adversarial sample is equally fragile. As every pixel of the adversarial sample plays a crucial role in maximizing the loss of the model, this results in the adversarial sample also in a vulnerable decision interval. Accordingly, we can reduce the maximization of the loss function by simply altering the pixel values. In the adversarial sample, each pixel is highly connected, therefore we can destroy the effect of the attack on the adversarial sample by simply corrupting the correlation between the adversarial samples.

#### Perturbation filtering verification based on bilinear interpolation (*PVB*)

Disturbances for adversarial attacks are obtained based on gradient calculations, while the method allows the model to extract the wrong feature information by adding a disturbance that makes the model gradient rise. Such as PGD, by computing against the classification loss function of the model to get the adversarial samples that can make the model misclassify, or by targeting the localization loss function to make the objectives lost or add false targets. The equivalent of the attack is as follows:

$$x'_{t+1} = Clip_{(0,255)}(x'_t + \alpha sign(\nabla_x L(x'_t, y, \theta)), \tag{5}$$

where $x'_t$ represents the image that needs to be attacked iteratively, while y is the correct label of the image and $\theta$ is the network parameter of the model. Through deriving the loss $L$ of the network, the gradient information of this loss function is obtained by finding the partial derivative. By *sign* function to obtain the direction of the attack and add the perturbation amplitude $\alpha$ to get the perturbation of one attack, in addition to changing the perturbation to the original image and restricting the image to the normal range. The final attack image $x'_N$ is obtained by iterating N times.

Since the adversarial samples are carefully calculated images, in other words, each pixel of the adversarial sample has an essential role to fulfill. The correlation between individual pixels is so highly significant that we only need to disrupt the correlation between the adversarial samples to disrupt the attack's effect on the adversarial samples. Around this idea, we propose a perturbation filtering verification module based on bilinear interpolation to filter images without significant loss of image quality. Stated differently, we use bilinear interpolation as an interpolation method to up and down-sample the image, where the sampling process destroys the attacker's carefully designed perturbation by losing the association information of the adversarial samples.

Bilinear interpolation is an image scaling method that estimates the value of a new pixel by taking a weighted average of the four neighboring pixels. We downsample the equation as follows

$$f_{ds}(i, j, ds) = (1 - ds)^2 f(i, j) + ds \cdot (1 - ds) f(i + 1, j) + ds \cdot (1 - ds) f(i, j + 1) + ds^2 f(i + 1, j + 1), \tag{6}$$

where *f(i,j)* represents the pixel values in the original image with horizontal coordinate $i$ and vertical coordinate $j$, $ds$ is the scale of the image scaling, and $f_{ds}(i, j, ds)$ is the pixel value of the $(i, j)$ coordinate of the image after downsampling. By downsampling we lose some of the pixel values of the image, but this affects the model accuracy, therefore we need to upsample the image again to

restore the features of the image, so we zoom in at a magnification $us = (1 - ds)$ and we use the upsampling formula as follows:

$$f_{us}(i,j,ds) = (1-us)^2 f(i,j) + us \cdot (1-us) f(i+1,j) + us \cdot (1-us) f(i,j+1) + us^2 f(i+1,j+1). \quad (7)$$

To simplify the formula, we specify $s = ds$ and the simplified formula is as follows:

$$F_{PVB}(x_N^{'}) = \Sigma_{j=0}^{J-1} \Sigma_{i=0}^{I-1} \lfloor f_{us}[f_{ds}(i,j,s),(1/s)] \rfloor, \quad (8)$$

where $I$ equals the length of the image and $J$ equals the width of the image, we control the image sampling ratio by controlling the hyperparameter $s$. Since the image has pixels from 0 to 255, we use $\lfloor\rfloor$ to round down. The appropriate parameter is selected to improve the robustness of the model. Compared with RobustDet's way of reconstructing images using VAE, our proposed method is better at destroying the adversarial perturbation of images since VAE [39] needs to extract the feature information of images, while the sampling method can destroy the features carefully designed by the adversarial samples. The overall idea is shown in Fig. 4, where we want the reconstructed image to be far away from the wrong decision interval.

## Experiment

### Experimental environment settings

To demonstrate the effectiveness of our proposed method, with all experiments we trained on the same equipment.

Experimentation on MS-COCO [7] and PASCAL VOC [6] datasets was performed for our done work, training using the COCO2017 training set and the PASCAL VOC 2007 and PASCAL VOC 2012 training sets. As a comparison with the mainstream methods, the test sets we used for validation were the COCO2017 and PASCAL VOC 2012 test sets.

Our experiments were run on CPU: Intel(R) Xeon(R) Gold 5318Y CPU @ 2.10GHz and graphics card NVIDIA A100 Tensor Core GPU. The training was performed using an SGD [43] gradient descent strategy with a learning rate of 1e-3 and a momentum of 0.9.

Concerning the evaluation metrics of experimental robustness, we first tested the metrics of mainstream models using CWA, DAG, and PGD attacks for localization or classification, meanwhile, we also evaluated the performance metrics of CON, and MTD attacks to demonstrate the effectiveness of our work. Regarding the hyperparameters $\alpha$ and $\beta$, we set the same parameters as RobustDet to reflect the superior performance of our proposed method. For $m$ and $s$, we chose 0.85 and 0.8, for which we will demonstrate why this value was chosen in the subsequent ablation experiments.

### Evaluation indicators

*AP* (Average Precision) is one of the common metrics used to evaluate the performance of target detection or image classification algorithms. It represents the average precision of the model for different confidence thresholds. Suppose we have N categories, each with a different number of positive and negative samples. For each category, we sort the results predicted by the model in order of confidence from high to low, and then based on the different confidence thresholds, we calculate the precision

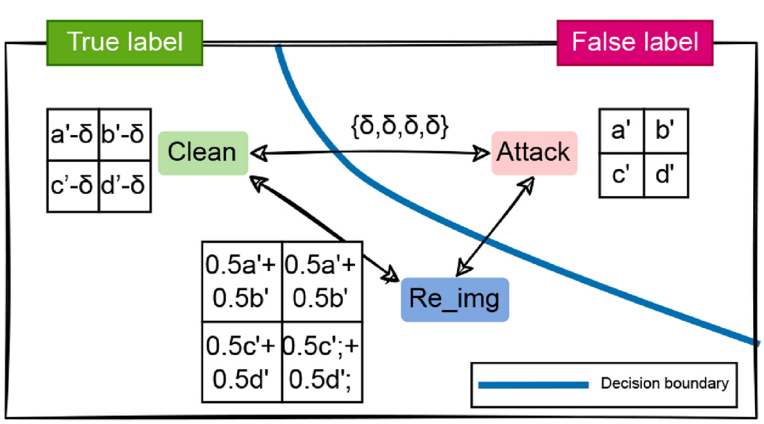


**Fig. 4** Blue is the decision boundary, in which the left side of the decision boundary is clean labels while the right side is wrong labels. We obtained the adversarial samples by adding a perturbation of $\delta$ to the clean samples ($a^{'}$-$\delta$, $b^{'}$-$\delta$, $c^{'}$-$\delta$, $d^{'}$-$\delta$) ($a^{'}$, $b^{'}$,$c^{'}$, $d^{'}$) are classified as mislabeled. Moreover, our reconstructed images by interpolation (0.5$a^{'}$+0.5$b^{'}$, 0.5$a^{'}$+0.5$b^{'}$, 0.5$c^{'}$+0.5$d^{'}$, 0.5$c^{'}$+0.5$d^{'}$) reverted to the correct category. In other words, our reconstructed image increases the distance to the adversarial sample and makes its escape from the interval of the adversarial sample

and the corresponding recall under each threshold. The AP under that category is then calculated based on this precision and recall.

Calculate the precision P and recall R for each category:

$$Precision = TP/(TP + FP), \qquad Recall = TP/(TP + FN). \tag{9}$$

Different APs were calculated for each category based on different confidence thresholds and then averaged:

$$AP = \frac{1}{n}\sum_{i=1}^{n} P(i) \cdot \Delta R(i), \qquad mAP = \frac{1}{N}\sum_{j=1}^{N} AP(j), \tag{10}$$

where $n$ denotes the total number of recall points, $P(i)$ denotes the precision at the ith recall point, and $\Delta R(i)$ denotes the difference between two recalls at the ith recall point. Eventually, the $mAP$ was calculated for all categories. in which $N$ denotes the total number of categories.

### Main experiments

Our major and subsidiary experiments are validated on the PASCAL VOC 2007 and COCO2017 datasets, as shown in Tables 1 and 2.

It is observed from Table 1 that the object detection model SSD performs remarkably well in detecting clean samples with 77.5% without the use of adversarial training. However, such a model lacks security as its performance drops to 1.8% and 4.5% under the classification attack $A_{cls}$ and localization attack $A_{loc}$ by PGD, while its performance is also extremely terrible under the *CWA* and *DAG* attacks. Whereas AT training involves adding adversarial samples to the training to improve robustness, this method significantly attenuates the performance of clean samples. As we can see in Table 1, the SSD-AT models trained by adding either $A_{cls}$ or $A_{loc}$ both significantly reduce the recognition rate of clean samples, *emph mAP* from 77.5% to 46.7% and 51.9%, although there is still some improvement in resistance for adversarial attacks. MTD is trained by selecting adversarial samples between $A_{cls}$ and $A_{loc}$ according to the loss size, which slightly improves the robustness although it reduces the performance of clean samples compared to the former. It has excellent clean sample detection performance and superior robustness as for RobustDet, however, the performance difference between clean samples and adversarial samples of this method is still substantial.

By comparison, the detection performance of our proposed method on clean samples is slightly lower than that of RobustDet, albeit only by 2.4%, in exchange for strong robustness. It is evident from Table 1 that for $A_{cls}$ and $A_{loc}$ adversarial attacks, the detection performance is

**Table 1** Results of *mAP* evaluation against adversarial attack methods on the PASCAL VOC 2007 test set

| Method | Conference | Clean | $A_{cls}$ [21] | $A_{loc}$ [21] | CWA [19] | DAG [34] |
|---|---|---|---|---|---|---|
| SSD [29] | ECCV2016 | **77.5** | 1.8 | 4.5 | 1.2 | 4.9 |
| SSD-AT($A_{cls}$) [29] | ICCV2019 | $46.7^{-30.8}$ | $21.8^{+20.0}$ | $32.2^{+27.7}$ | - | $28.0^{+23.1}$ |
| SSD-AT($A_{loc}$) [29] | ICCV2019 | $51.9^{-25.6}$ | $23.7^{+21.8}$ | $26.5^{+22.0}$ | - | $17.2^{+12.3}$ |
| MTD [18] | ICCV2019 | $48.0^{-29.5}$ | $29.1^{+28.1}$ | $31.9^{+27.4}$ | $18.2^{+17.0}$ | $28.5^{+23.6}$ |
| CWAT(PGD-10) [19] | CVPR2021 | $51.3^{-26.2}$ | $22.4^{+20.6}$ | $36.7^{+32.2}$ | $19.9^{+18.7}$ | $50.3^{+45.4}$ |
| RobustDet [20] | ECCV2022 | $75.4^{-2.1}$ | $41.5^{+40.7}$ | $45.2^{+40.7}$ | $42.4^{+41.2}$ | $52.0^{+47.1}$ |
| RobustDet* [20] | ECCV2022 | $74.8^{-2.7}$ | $45.9^{+44.1}$ | $49.1^{+44.6}$ | $48.0^{+46.8}$ | $56.6^{+51.7}$ |
| RPU-PVB | - | $73.0^{-4.5}$ | **60.2** $^{+58.4}$ | **57.5** $^{+53.0}$ | **60.9** $^{+59.7}$ | **63.2** $^{+58.3}$ |

* indicates that the model uses the CFR module

bold represents the best achievement for the indicator

**Table 2** Results of *mAP* evaluation against adversarial attack methods on the MS COCO 2017 test set

| Method | Conference | Clean | $A_{cls}$ [21] | $A_{loc}$ [21] | CWA [19] | DAG [34] |
|---|---|---|---|---|---|---|
| SSD [29] | ECCV2016 | **42.0** | 0.4 | 1.8 | 0.1 | 8.1 |
| MTD [18] | ICCV2019 | $24.2^{-17.8}$ | $13.0^{+12.6}$ | $13.4^{+11.6}$ | $7.7^{+7.6}$ | - |
| CWAT(PGD-10) [19] | CVPR2021 | $23.7^{-18.3}$ | $14.2^{+13.8}$ | $15.5^{+13.7}$ | $9.2^{+9.1}$ | - |
| RobustDet [20] | ECCV2022 | $36.7^{-5.3}$ | $20.6^{+20.2}$ | $19.4^{+17.6}$ | $20.5^{+20.4}$ | $24.5^{+16.4}$ |
| RobustDet* [20] | ECCV2022 | $36.0^{-6.0}$ | $20.0^{+19.6}$ | $19.0^{+17.2}$ | $19.9^{+19.8}$ | $16.6^{+8.5}$ |
| RPU-PVB | - | $36.2^{-5.8}$ | **24.5** $^{+24.1}$ | **27.1** $^{+25.1}$ | **25.3** $^{+25.2}$ | **26.6** $^{+18.5}$ |

* indicates that the model uses the CFR module

bold represents the best achievement for the indicator

almost invulnerable and is significantly similar to that of clean samples. Similarly, for CWAT and DAG, we obtain superb robustness of 60.9% and 63.2%. In summary, the detection difference between our clean and adversarial samples ranges from a minimum of only 9.8% to a maximum of only 15.5%, making our proposed method more valuable for real-world applications.

To demonstrate that our proposed method is not restrictive to a single dataset, we additionally finalized our experiments on the MS COCO2017 dataset, presented in Table 2:

It is observed that for larger datasets, the performance degradation of the attacked SSDs is more significant, with only a small improvement in robustness and a significant decrease in the theoretical ability to detect clean samples, even when using the MTD and CWAT training methods for adversarial defense. While for RobustDet there is a significant improvement in the detection ability for both clean images and adversarial samples, for adversarial samples most of the detection abilities are still below 20%. As a comparison, the detection ability of our proposed method for clean samples only declines by 5.8%, while the overall detection ability for adversarial samples enhances by more than 24%, which significantly narrows the gap between the quality of performance of clean samples and adversarial samples. In other words, our proposed method is well-portable, i.e., it achieves high performance on different datasets.

### Ablation experiments

As an intuitive representation of the effectiveness of our work, we demonstrate our conclusions by parameter and module ablation. At first, to show the rationality of our selected parameters, we set the evaluation index of the robustness of the adversarial sample (*RA*) and the overall robustness evaluation index that contains clean images (*OA*), which is formulated as follows:

$$RA = \frac{1}{N}\Sigma_{a=A_1[0]}^{A_1} mAP_a, \qquad OA = \frac{1}{N}\Sigma_{a=A_2[0]}^{A_2} mAP_a, \quad (11)$$

where $A_1$={cls, loc, con, cwat, dag, mtd}, $A_2$={clean, cls, loc, con, cwat, dag, mtd}, a traverses the *mAP* results starting from the first attack, in which $N$ is the number of $A_1$. This means that *RA* is the metric that reflects the fear of average robustness for adversarial samples, while *OA* is the average robustness metric with the addition of clean samples. The ablation experiments for our proposed *RPU*, which possesses a hyperparameter *m*, are shown in Table 3.

To more visually express the reasonableness of the selected values, we labeled the highest results in bright yellow, while we used light yellow for the second-best results. From the table, we can see that *mAP* obtains the highest results in each indicator when the m value is 0.85 among all parameters. The dominance of this parameter can be more intuitively seen in the *RA* and *OA* metrics, as *RA* and *OA* are the metrics that are evaluated together. In contrast, the model's performance reaches sub-optimal levels when the m value is 0.45. Thus the robustness of the model does not gradually increase as *m* increases.

Regarding the parameter *s* of the *PVB* module, we have experimented with most of the common interpolation methods, while s stands for the scaling multiplier, certainly, the optimal and sub-optimal results we have also used bright yellow and light yellow to show them more clearly. As shown in Table 4:

For SSD and RobustDet, where we can see that both achieved optimal and suboptimal performance on clean samples, respectively, whereas we saw an overall improvement in robustness after removing the *PVB* module. While we added the interpolation-based perturbation filtering verification method, we can see that different interpolations have very different effects under the control of scaling factors. It starts with *INTER_AREA* achieving optimal results against classification attacks with *s* equal to 0.5, and partial sub-optimal performance under other attack methods. The *INTER_LINEAR* method we use filters the adversarial perturbations more effectively and achieves the best performance for a total of four attack methods. Moreover, the advantages of our proposed method can be seen more clearly in the *RA* and *OA* metrics. Whereas for the other interpolation methods, the improvement is small or even inferior to the performance of removing the *PVB* module. Therefore, the effectiveness of our proposed *PVB* method is proved.

**Table 3** RPU module hyperparameter *m* ablation experiment

| m | clean | cls [21] | loc [21] | con [21] | cwat [19] | dag [34] | mtd [18] | RA | OA |
|---|---|---|---|---|---|---|---|---|---|
| 0.25 | 74.6 | 42.5 | 53.2 | 43.1 | 43.0 | 64.2 | 42.8 | 48.1 | 51.9 |
| 0.45 | 73.9 | 52.0 | 55.4 | 51.9 | 52.8 | 65.2 | 46.7 | 54.0 | 56.8 |
| 0.65 | 74.4 | 41.9 | 53.1 | 42.6 | 43.0 | 64.4 | 42.7 | 48.0 | 51.7 |
| 0.85 | 74.7 | 55.5 | 56.7 | 55.4 | 57.1 | 65.5 | 49.4 | 56.6 | 59.2 |

**Table 4** PVB module hyperparameter *s* ablation experiment

| | s | clean | cls [21] | loc [21] | con [21] | cwat [19] | dag [34] | mtd [18] | RA | OA |
|---|---|---|---|---|---|---|---|---|---|---|
| SSD [29] | - | 77.5 | 1.8 | 4.5 | - | 1.2 | 4.9 | - | 3.1 | 18.0 |
| RobustDet* [20] | - | 74.8 | 45.9 | 49.1 | - | 48.0 | 56.6 | - | 49.9 | 54.9 |
| Ours/-PVB | - | 74.7 | 55.5 | 56.7 | 55.4 | 57.1 | 65.5 | 49.4 | 56.6 | 59.2 |
| INTER_AREA | 0.2 | 16.0 | 42.2 | 39.5 | 41.8 | 33.5 | 36.9 | 33.5 | 37.9 | 34.7 |
| | 0.5 | 69.7 | 61.2 | 54.9 | 60.8 | 61.6 | 58.5 | 52.5 | 58.3 | 59.9 |
| | 0.8 | 73.4 | 59.6 | 57.1 | 59.6 | 60.3 | 63.5 | 52.3 | 58.7 | 60.8 |
| INTER_NEAREST | 0.2 | 11.3 | 26.7 | 28.2 | 26.9 | 28.1 | 63.8 | 45.2 | 36.5 | 32.9 |
| | 0.5 | 69.8 | 50.8 | 53.8 | 50.9 | 52.6 | 62.1 | 45.0 | 52.5 | 55.0 |
| | 0.8 | 72.6 | 50.9 | 54.4 | 51.3 | 52.5 | 63.8 | 45.1 | 53.0 | 55.8 |
| INTER_LINEAR | 0.2 | 40.3 | 51.1 | 41.1 | 50.6 | 51.2 | 37.7 | 39.9 | 45.3 | 44.6 |
| | 0.5 | 69.7 | 60.9 | 55.1 | 60.9 | 61.7 | 58.5 | 51.9 | 58.2 | 59.8 |
| | 0.8 | 73.0 | 60.2 | 57.5 | 59.9 | 60.9 | 63.2 | 52.6 | 59.0 | 61.0 |
| INTER_CUBIC | 0.2 | 38.1 | 50.0 | 41.8 | 49.7 | 50.0 | 36.1 | 39.3 | 44.5 | 43.5 |
| | 0.5 | 72.3 | 58.0 | 56.7 | 58.1 | 58.8 | 62.8 | 50.8 | 57.5 | 59.6 |
| | 0.8 | 74.4 | 56.4 | 57.0 | 56.9 | 57.7 | 65.2 | 50.7 | 57.3 | 59.8 |
| INTER_LANCZOS4 | 0.2 | 33.4 | 47.8 | 40.5 | 48.2 | 48.8 | 34.8 | 38.5 | 43.1 | 41.7 |
| | 0.5 | 71.1 | 54.7 | 55.2 | 54.6 | 55.7 | 62.1 | 48.0 | 55.1 | 57.3 |
| | 0.8 | 74.4 | 55.5 | 56.6 | 55.5 | 56.9 | 65.5 | 49.5 | 56.6 | 59.1 |

* indicates that the model uses the CFR module

At the same time, we performed module ablation experiments, as shown in Table 5: we can see that after adding the *RPU* module, we gained a huge overall improvement compared to the current most advanced method RobustDet, except for a 0.1 performance reduction on clean samples. While adding the *PVB* module sacrifices a small amount of clean sample detection performance and substantially increases the robustness against all attacks. Thus the above experiments demonstrate the effectiveness of our proposed approach.

Coming from Fig. 5 we can see that the green color represents the *AP* performance for images that did not receive any attacks, while the red and blue colors represent the *AP* performance that suffered from classification and localization. We can see that the bottled and potted plant performance degradation is extremely significant. It is believed that the reason for the relatively large performance degradation in this category is related to the fact that both categories are small targets and the modification of small target pixels by the adversarial perturbation is evident. Even though we filter the adversarial samples using interpolation, the normal features of the objectives are filtered out as well. Nevertheless, overall our approach reduces the model performance difference (the gap between clean and adversarial samples) to a minimum.

## Visualization

### Category performance show

#### *Training process*

To the selection of epoch, we refer to the convergence interval of the loss function, as shown in Fig. 6: we can see that the loss starts to stabilize as it approaches 100000. Therefore all our previous experiments used an iteration count of 100000.

**Table 5** Module ablation experiment

| Method | RPU | PVB | Clean | $A_{cls}$ [21] | $A_{loc}$ [21] | CWA [19] | DAG [34] |
|---|---|---|---|---|---|---|---|
| SSD [29] | | | **77.5** | 1.8 | 4.5 | 1.2 | 4.9 |
| RobustDet* [20] | | | $74.8^{-2.7}$ | $45.9^{+44.1}$ | $49.1^{+44.6}$ | $48.0^{+46.8}$ | $56.6^{+51.7}$ |
| SSD [29] | ✓ | | $74.7^{-2.8}$ | $55.5^{+53.7}$ | $56.7^{+52.2}$ | $57.1^{+55.9}$ | $\mathbf{65.5}^{+60.6}$ |
| SSD [29] | ✓ | ✓ | $73.0^{-4.5}$ | $\mathbf{60.2}^{+58.4}$ | $\mathbf{57.5}^{+53.0}$ | $\mathbf{60.9}^{+59.7}$ | $63.2^{+58.3}$ |

✓ represents the addition of the module

* indicates that the model uses the CFR module

bold represents the best achievement for the indicator

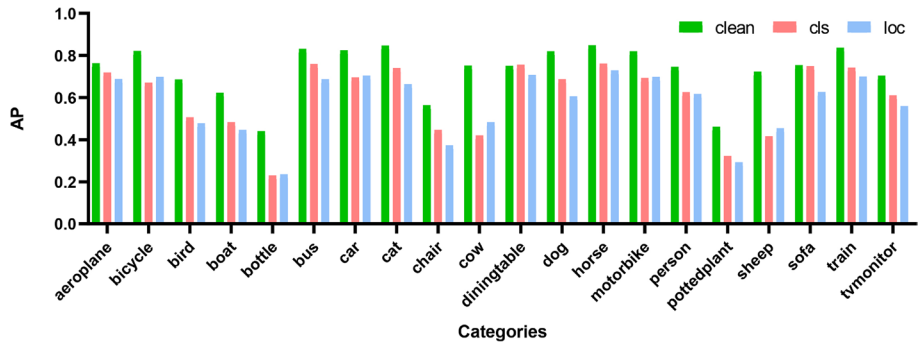


**Fig. 5** Classify and locate the performance of various categories of attacks

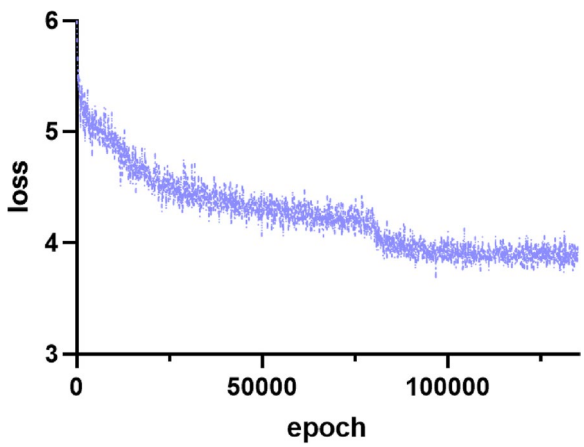


**Fig. 6** Loss function line graph

## Conclusions

In this paper, we propose a robust object detection model Based on a unified metric perspective with bilinear interpolation, starting from the perspective of bilinear interpolation, the fine-grained idea is first utilized to learn the adversary samples in comparison with the clean samples, thereby obtaining the correct dynamic convolutions parameters and a more robust model. Subsequently, a bilinear interpolation compression method is used, which, in combination with the former, drastically improves the robustness of the object detection model with an excellent performance difference (clean samples vs. adversarial samples). Our proposed method achieves the best results on different datasets, thus proving the relevance of the work we have done. Until now, research on methods for object detection against defense has been slow, and our future work aims to narrow the robustness performance difference of the model and use defense to facilitate the development of more advanced adversarial attacks.

### Additional information

The authors declare no conflict of interest. All images (including those taken in real life) were taken from a public dataset that is open source and was involved in the training of the models in this paper. This does not lead to the inclusion of information and images that could lead to the identification of research participants and therefore does not have any impact on the images themselves or the review process.

### Authors' contributions

H.Y. was responsible for the method proposal and implementation, experimental proof, and paper writing, Y.C and X.W. performed the experimental setting and grant support, and H.D and Y.Z revised the paper. All authors reviewed the manuscript.

### Funding

This research was partially supported by the National Natural Science Foundation of China (62202118, 61962009), the Natural Science Research Project of Guizhou Provincial Department of Education (Qian jiao ji [2022]073), and the Science and Technology Tackling Project of Guizhou Provincial Department of Education (Qian jiao ji [2023]003).

### Availability of data and materials

The dataset for generating weights and analysis in this study is available on GitHub("https://github.com/KujouRiu/RPU-PVB").

## Declarations

### Ethics approval and consent to participate

All datasets used in this paper are publicly available and there are no ethical issues involved.

### Competing interests

The authors declare no competing interests.

Received: 15 September 2023 Accepted: 25 October 2023
Published online: 02 December 2023

### References

1. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 779–788
2. Qian L, Luo Z, Du Y, Guo L (2009) Cloud computing: An overview. In: Cloud Computing: First International Conference, CloudCom 2009, Beijing, China, December 1-4, 2009. Proceedings 1. Springer, pp 626–631
3. Saiyeda A, Mir MA (2017) Cloud computing for deep learning analytics: A survey of current trends and challenges. Int J Adv Res Comput Sci 8(2):68–72

4. Papernot N, McDaniel P, Wu X, Jha S, Swami A (2016) Distillation as a defense to adversarial perturbations against deep neural networks. In: 2016 IEEE symposium on security and privacy (SP). IEEE, pp 582–597
5. Jia X, Wei X, Cao X, Foroosh H (2019) Comdefend: An efficient image compression model to defend adversarial examples. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019. Long Beach: Computer Vision Foundation/IEEE. pp. 6084–6092. https://doi.org/10.1109/CVPR.2019.00624
6. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A (2015) The pascal visual object classes challenge: A retrospective. Int J Comput Vis 111:98–136
7. Lin T-Y, Maire M, Belongie SJ, Hays J, Perona P, Ramanan D, Dollár P, Lawrence Zitnick C (2014) Microsoft COCO: common objects in context. In: Fleet DJ, Pajdla T, Schiele B, Tuytelaars T (ed) Computer Vision - ECCV 2014 - 13th European Conference. Proceedings, Part V, volume 8693 of Lecture Notes in Computer Science. Springer, Zurich, p 740–755
8. Han H, Fei S, Yan Z, Zhou X (2022) A survey on blockchain-based integrity auditing for cloud data. Digit Commun Netw 8(5):591–603
9. Guo L, Chen J, Li S, Li Y, Lu J (2022) A blockchain and iot-based lightweight framework for enabling information transparency in supply chain finance. Digit Commun Netw 8(4):576–587
10. Yan Z, Zheng Q, Wu Y, Zhao Y, Atiquzzaman M (2022) Guest editorial: Blockchain-enabled technologies for cyber-physical systems and big data applications. Digit Commun Netw 8(5):589-590
11. Sun Z, Wan J, Yin L, Cao Z, Luo T, Wang B (2022) A blockchain-based audit approach for encrypted data in federated learning. Digit Commun Netw 8(5):614–624
12. Huang Y, Yu Y, Li H, Li Y, Tian A (2022) Blockchain-based continuous data integrity checking protocol with zero-knowledge privacy protection. Digit Commun Netw 8(5):604–613
13. Wang F, Li G, Wang Y, Rafique W, Khosravi MR, Liu G, Liu Y, Qi L (2023) Privacy-aware traffic flow prediction based on multi-party sensor data with zero trust in smart city. ACM Trans Internet Technol 23(3):44:1–44:19
14. Miao Y, Bai X, Cao Y, Liu Y, Dai F, Wang F, Qi L, Dou W (2023) A novel short-term traffic prediction model based on svd and arima with blockchain in industrial internet of things. IEEE Internet Things J (99):1-1
15. Xu X, Gu J, Yan H, Liu W, Qi L, Zhou X (2022) Reputation-aware supplier assessment for blockchain-enabled supply chain in industry 4.0. IEEE Trans Ind Inform 19(4):5485–5494
16. He Q et al (2023) Edindex: Enabling fast data queries in edge storage systems. In Chen, H. et al. (eds.) Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023. Taipei: ACM. 675–685. https://doi.org/10.1145/3539618.3591676
17. Yuan L, He Q, Chen F, Zhang J, Qi L, Xu X, Xiang Y, Yang Y (2021) Csedge: Enabling collaborative edge storage for multi-access edge computing based on blockchain. IEEE Trans Parallel Distrib Syst 33(8):1873–1887
18. Zhang H, Wang J (2019) Towards adversarially robust object detection. In 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019. Seoul, Korea (South): IEEE. 421–430. https://doi.org/10.1109/ICCV.2019.00051
19. Chen P, Kung B, Chen J (2021) Class-aware robust adversarial training for object detection. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual. Computer Vision Foundation/IEEE. pp. 10420–10429. https://doi.org/10.1109/CVPR46437.2021.01028
20. Dong Z, Wei P, Lin L (2022) Adversarially-aware robust object detector. In: Avidan S, Brostow GJ, Cisśe M, Farinella GM, Hassner T (ed) Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part IX, volume 13669 of Lecture Notes in Computer Science. Springer, p 297–313
21. Kurakin A, Goodfellow IJ, Bengio S (2017) Adversarial examples in the physical world. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings. OpenReview.net
22. Moosavi-Dezfooli S, Fawzi A, Frossard P (2016) Deepfool: A simple and accurate method to fool deep neural networks. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016. Las Vegas: IEEE Computer Society. pp. 2574–2582. https://doi.org/10.1109/CVPR.2016.282
23. Dong Y, et al (2018) Boosting adversarial attacks with momentum. In 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018. Salt Lake City: Computer Vision Foundation / IEEE Computer Society. pp. 9185–9193. https://doi.org/10.1109/CVPR.2018.00957
24. Lin J, Song C, He K, Wang L, Hopcroft JE (2020) Nesterov accelerated gradient and scale invariance for adversarial attacks. In 8th International Conference on Learning Representations, ICLR 2020. Addis Ababa, Ethiopia, April 26-30, 2020 (OpenReview.net, 2020)
25. Huang Y, Chen Y, Wang X, Yang J, Wang Q (2023) Promoting adversarial transferability via dual-sampling variance aggregation and feature heterogeneity attacks. Electronics 12(3):767
26. Laykaviriyakul P, Phaisangittisagul E (2023) Collaborative Defense-GAN for protecting adversarial attacks on classification system. Expert Syst Appl 214:118957
27. Carmon Y, Raghunathan A, Schmidt L, Duchi JC, Liang PS (2019) Unlabeled data improves adversarial robustness. Adv Neural Inf Process Syst 32:11190-11201
28. Hendrycks D, Mazeika M, Kadavath S, Song D (2019) Using self-supervised learning can improve model robustness and uncertainty. Adv Neural Inform Process Syst 32:15637-15648
29. Liu W, Anguelov D, Erhan D, Szegedy C, Reed SE, Fu C-Y, Berg AC (2016) SSD: single shot multibox detector. In: Leibe B, Matas J, Sebe N, Welling M (eds) Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I, volume 9905 of Lecture Notes in Computer Science. Springer, p 21–37
30. Lin T, Goyal P, Girshick RB, He K, Dollár P (2020) Focal loss for dense object detection. IEEE Trans Pattern Anal Mach Intell 42:318–327. https://doi.org/10.1109/TPAMI.2018.2858826
31. Girshick RB (2015) Fast R-CNN. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago: IEEE Computer Society. 1440–1448. https://doi.org/10.1109/ICCV.2015.169
32. Hu H, Gu J, Zhang Z, Dai J, Wei Y (2018) Relation networks for object detection. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. p 3588–3597. https://doi.org/10.1109/CVPR.2018.00378 (Computer Vision Foundation / IEEE Computer Society, 2018)
33. Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: Vedaldi A, Bischof H, Brox T, Frahm J-M (eds) Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I, volume 12346 of Lecture Notes in Computer Science. Springer, p 213–229
34. Xie C, et al (2017) Adversarial examples for semantic segmentation and object detection. In: IEEE International Conference on Computer Vision, ICCV 2017. IEEE Computer Society, Venice, p 1378–1387. https://doi.org/10.1109/ICCV.2017.153
35. Wei X, Liang S, Chen N, Cao X (2019) Transferable adversarial attacks for image and video object detection. In: Kraus S (ed) Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019. p954–960. https://doi.org/10.24963/IJCAI.2019/134 (https://ijcai.org)
36. Chen Y, Yang H, Wang X, Wang Q, Zhou H (2023) Glh: From global to local gradient attacks with high-frequency momentum guidance for object detection. Entropy 25(3):461
37. Liu X, Yang H, Liu Z, Song L, Chen Y, Li H (2019) DPATCH: an adversarial patch attack on object detectors. In: Espinoza H, Éigeartaigh SÓ, Xiaowei Huang, Herńandez-Orallo S, Castillo-Effen M (eds). Workshop on Artificial Intelligence Safety 2019 co-located with the Thirty-Third AAAI Conference on Artificial Intelligence 2019 (AAAI-19), Honolulu, Hawaii, January 27, 2019, volume 2301 of CEUR Workshop Proceedings. https://CEUR-WS.org
38. Hu Z, Huang S, Zhu X, Sun F, Zhang B, Hu X (2022) Adversarial texture for fooling person detectors in the physical world. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022. IEEE, p 13297–13306
39. Kingma DP, Welling M (2014) Auto-encoding variational bayes. In: Bengio Y, LeCun Y (eds) 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings
40. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern

Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, p 770–778
41. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: A unified embedding for face recognition and clustering. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. IEEE Computer Society, p 815–823
42. Sun Y, Cheng C, Zhang Y, Zhang C, Zheng L, Wang Z, Wei Y (2020) Circle loss: A unified perspective of pair similarity optimization. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. Computer Vision Foundation / IEEE, p 6397–6406
43. Sinha NK, Griscik MP (1971) A stochastic approximation method. IEEE Trans Syst Man Cybern 1(4):338–344

## Publisher's Note