



A survey on adversarial attacks and defenses for object detection and their applications in autonomous vehicles

Abdollah Amirkhani¹ · Mohammad Parsa Karimi¹ · Amin Banitalebi-Dehkordi²

Accepted: 22 August 2022 / Published online: 12 September 2022

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract

Object detection is considered as one of the most important applications of deep learning. However, the object detection techniques lose their effectiveness and reliability when they fall victim to adversarial attacks. This big flaw has made it challenging to fully adopt the object detection applications in important products and essential industries such as autonomous vehicles. While the field of adversarial robustness has witnessed a great deal of achievement in building sophisticated methods of attack and defense, the majority of the work has been focused on the task of image classification due to its simplicity in theory and practice. In this paper, we provide an up-to-date survey of recent advancements in the field of adversarial robustness for object detection. We review the prominent attack and defense mechanisms presented in the research community and provide discussions and insights on their strengths and weaknesses. In addition, we review the recent literature on adversarial robustness for applications related to autonomous vehicles, as a critical aspect of this high-impact emerging industry, in which the robustness of models is of vital importance.

Keywords Object detection · Adversarial attack · Robust detector · Adversarial defense

1 Introduction

Deep learning has helped solving many crucial artificial intelligence problems over the past decade, some of which were outstanding for a long time [1]. Nowadays, deep neural networks (DNNs) are employed to solve complex problems in various fields such as machine vision [2], natural language processing [3], big data processing [4], DNA analysis [5] and autonomous vehicles [6–9]. The rapid development and progress of the deep learning field is rooted not only in the high capability and performance of the deep learning approaches, but it also stems from other factors such as the concentrated efforts of researchers and the rapid improvement of the deep learning models [10], discovering the importance of the different deep learning applications [11], hardware advancements [12], the increased power of the computer processors and the graphical processing units [13],

and the development of various software libraries and platforms [14]. Today, DNNs play a vital role in our lives, which can even overshadow our health and well-being [15]. Owing to the high precision and the low error rate of these networks and, thus, the amount of trust they have been able to earn, the DNNs have been reliably employed in the sensitive areas [16] such as health [17], face recognition [18], autonomous flying vehicles [19], and many other fields. The deep learning approaches, especially those associated with machine vision, play a significant role in our day-to-day activities, and this role will become even more prominent in the future [20].

Machine vision is the field in which most of the deep learning breakthroughs are rooted [21]. Before the advent of the deep learning, the conventional machine vision techniques had a much lower performance [22], and the deep learning concept tremendously evolved the precision of the machine vision systems [23]. DNNs helped improve machine perception to the point that it surpassed the natural human vision precision in some cases [24]. After a short while, deep learning models were employed in different machine vision applications such as classification [25], segmentation [26], semantic segmentation [27], face recognition [28], object detection [29], and tracking [30].

✉ Abdollah Amirkhani
amirkhani@iust.ac.ir

¹ School of Automotive Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran

² Big Data & Intelligence Platform Lab, Huawei Technologies Canada Co. Ltd., Markham, Ontario, Canada

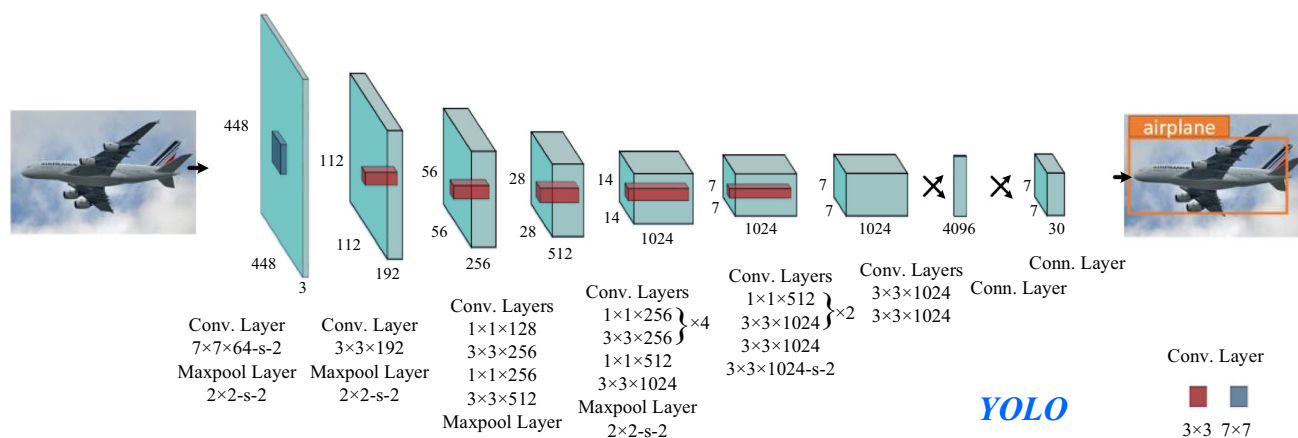


Fig. 1 The basic operation pipeline of an object detector

Many high-precision object detectors based on the DNN models have been introduced in the recent years. The main task and challenge of these object detectors is to detect the target objects and their positions in different classes and images [31]. Object detectors are first trained on a number of labeled images, and then a trained DNN performs inference on unlabeled input images. The output will be in the form of bounding boxes and class categories for object types seen during training [32]. Figure 1 shows a basic overview of an object detection pipeline.

As it can be observed in Fig. 1, after passing an image through the different layers of an object detection neural network, the target objects and their positions in the input image are detected by the object detector. The detection of objects in video recordings is an essential task related to autonomous vehicles, and due to the importance of this field, it has been the subject of numerous investigations [33]. The processing of objects in videos is a complex task, because the quality of every frame which is isolated from a video recording deteriorates and needs to be boosted and enhanced independently [34]. In certain applications, sometimes it is necessary to perform object detection and object tracking simultaneously. For this purpose, a new task called the “Video Instance Segmentation” has been introduced in the field of video processing [35].

In the last several years, many high-precision and fast object detection models have been presented [36]. Some of the most important object detection models include the different versions of the YOLO model [37], the FRCNN model [38], and the SSD model [39]. These models are considered as the generic object detectors, and their task is to detect all objects in an image and to outline their positions by means of the bounding boxes [40]. Another form of object detection is the salient object detection, in which the detectors try to find the visually dominant objects in an image [41].

A critical flaw of the DNNs was discovered by Szegedy et al. [42], when they showed that these networks are

highly vulnerable against adversarial attacks. The adversarial attacks are small perturbations which normally are imperceptible to the human eye, but they can completely mislead the DNNs [43]. Since the publishing of the findings by Szegedy et al. [42], a significant part of the research works in the field of deep learning was devoted to the adversarial attacks/defenses [44–57].

While the field of adversarial robustness has witnessed a great deal of achievement in building sophisticated methods of attack and defense, the majority of the work has been focused on the task of image classification due to its simplicity in theory and practice [58]. As a result, there has been little coherent effort to survey the state-of-the-art research in adversarial robustness of object detectors. In the real world however, object detectors are much more adopted than classifiers, and thus dedicated studies on robustness of object detectors are required.

Considering the significance of the adversarial robustness in detector DNNs, in this paper, as seen in Fig. 2, we review the most important articles on the subject of adversarial attacks and defenses and generally adversarial robustness in object detection. We try to show the progress of the attacks over time in the study of attacks, and in the discussion of defenses, the advantages and disadvantages of each method have been presented. An overall comparison of different adversarial attacks is carried out in Table 1 based on the reduction of the mAP values. The mAP analysis [59] is a method of measuring the performance of detection models. Since most of the examined attacks can be applied on the FRCNN, this model was considered in comparing the mAP values.

Table 2 shows an overall comparison of the defense techniques against the adversarial attacks. In this table, for improving the precision, we have used the YOLO model and considered the DAG attack, because most of the defense strategies have presented their results based on this model and type of attack.

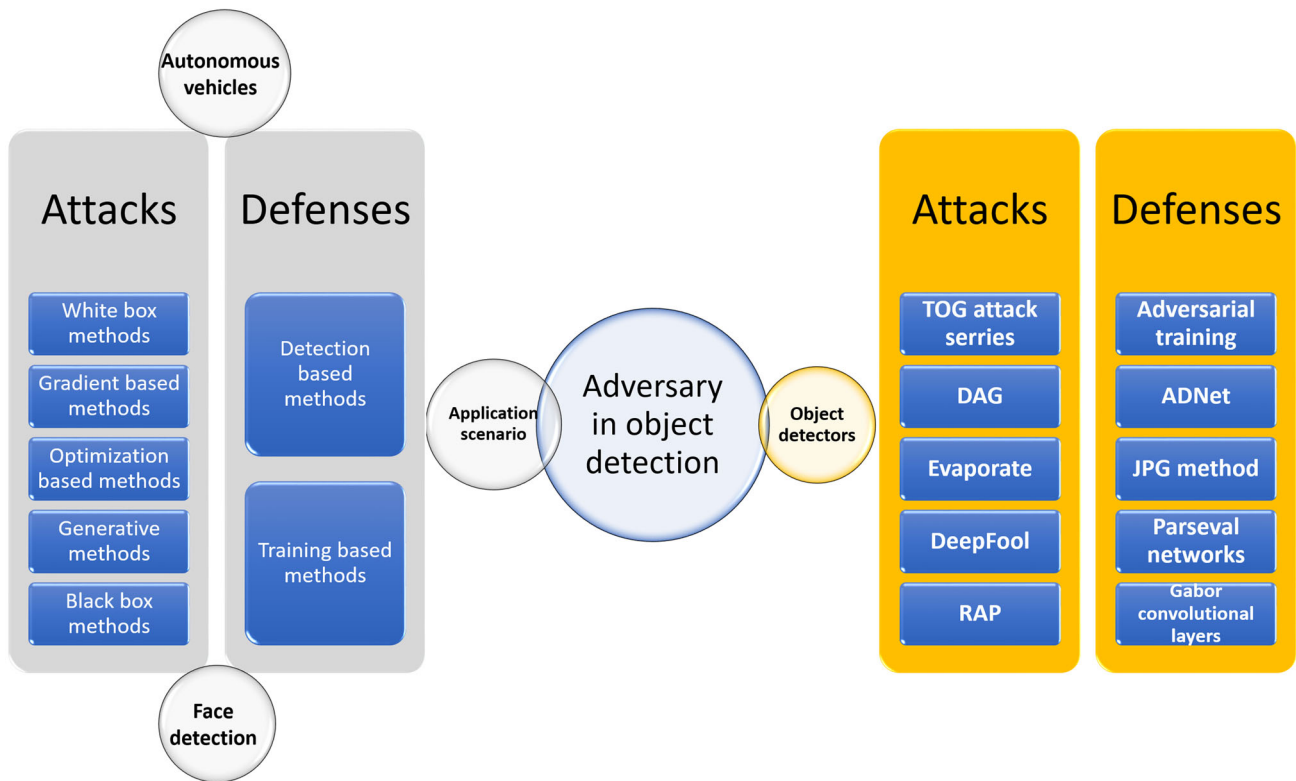


Fig. 2 A categorization of the methods discussed

We discuss the recent works on adversarial attacks and defense for object detection. Problem and terminology are discussed in Sect. 2. In Sect. 3, a general description of different types of adversarial attacks is presented. Some of the more prominent adversarial defenses for robustifying the DNNs in object detection are introduced in Sect. 4. The performances of the adversarial attacks in the field of sample application scenarios like autonomous vehicles and face detection are investigated in Sect. 5, and the conclusion of the paper is presented in Sect. 6.

2 Problem formulation and terminology

Adversarial attacks include small perturbations which are usually unrecognizable to the human eye but can be mixed into clean images and contaminate them. As stated in introduction section, these attacks are able to mislead the deep learning models and reduce their accuracy [66]. Figure 3 shows some example images perturbed by the adversarial attacks. As observed in this figure, the detection models have been deceived by these perturbations to a large extent and have made wrong detections. Let us formally give a definition of an adversarial attack. Suppose $O(\cdot)$ is an object detection model and let x be a clean input image. We expect the output to be object labels that can be displayed with a set like $L = \{l_1, l_2, \dots, l_n\}$ where n is the number of detected

objects in the input image. Normally we expect the object detector to act according to the following equation:

$$O(x) \rightarrow L = \{l_1, l_2, \dots, l_n\}. \tag{1}$$

Now suppose we add a small amount of perturbation like ρ to the image. The output of the detector changes to:

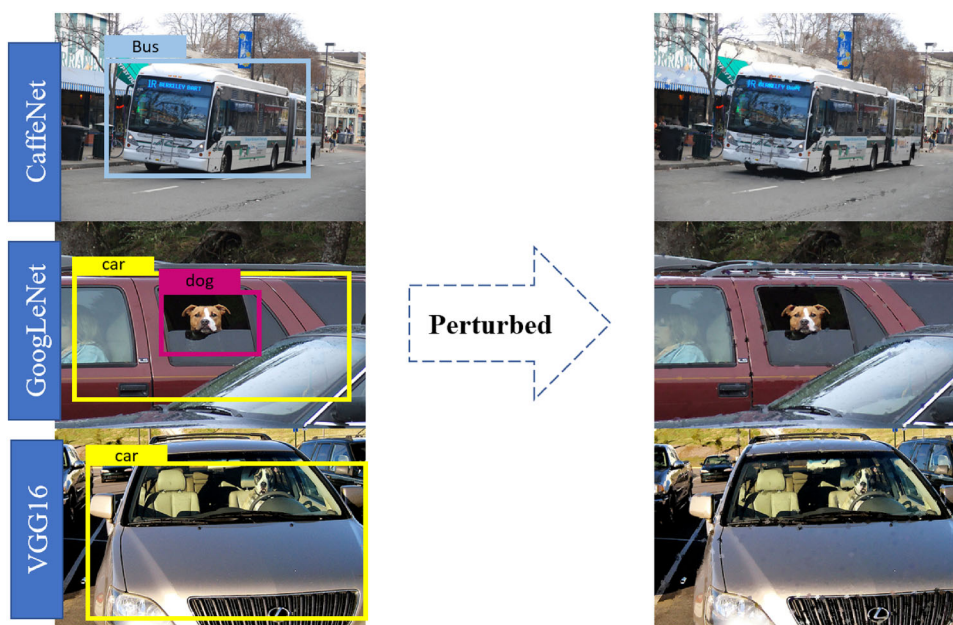
$$O(x + \rho) \rightarrow \bar{L} = \{\bar{l}_1, \bar{l}_2, \dots, \bar{l}_n\}. \tag{2}$$

Experimental results show that usually $L \neq \bar{L}$. And also in some cases \bar{L} and L are mutually exclusive. That means, in some attacks, the model is deceived in such a way that it does not recognize even a single object in the input image.

There are numerous technical expressions pertaining to the subject of the adversarial attacks, and we are just going to define some of them here.

Literally, the expression ‘‘adversarial perturbations’’ refers to the disturbances that are embedded into a clean image to turn it into an adversarial example [66]. As a perturbed version of a clean image, an adversarial example is intended to mislead or deceive a machine learning technique such as a DNN [67]. In the literature related to adversarial attacks, the expression ‘‘adversarial training’’ refers to a method of network training by means of the images perturbed by such attacks [68]. At a high level, adversarial attacks can be divided into several types: ‘targeted’ or ‘untargeted’

Fig. 3 Some examples of clean images and those perturbed by adversarial attacks, and the outcomes of the DNNs used to detect objects within these images. After perturbation, the network is not able to detect the objects anymore



attacks and ‘black-box’ or ‘white-box’ attacks. The untargeted attacks do not care about the final label and the misled labels; they just want to deceive an object detection model. What is important in the untargeted attacks is that objects get wrong labels [69], while the targeted attacks want to deceive a model so that it designates a particular label for a specific object. In fact, the targeted attacks are devised for a certain class of objects [70]. As mentioned earlier, there are different methods for adversarial defense as well. Generally, these defensive techniques can be divided into ‘one-shot’ and ‘iterative’ methods. The one-shot methods produce the adversarial disturbances by performing a one-shot computation (e.g., a one-time computation of a model’s loss gradient to generate a perturbation [71]), whereas the iterative methods perform the computations several times in order to generate a single disturbance. This operation is usually more costly than the one-shot procedure [72]. In general, these are the most common terminologies used in the literature published on the subject of the adversarial attacks, and we are going to use these expressions in the following sections.

3 Different types of attacks on object detection

In this section, we review the most common and frequently used adversarial attacks introduced in the field of object detection. We review the effects of these attacks on numerous datasets and models. An adversarial attack is considered to be more effective if it has a higher fooling rate and is able to reduce the accuracy of a model to a greater extent [73]. In the following subsections, we will explore these attacks.

According to [45], there are three tasks in the object detectors: detecting an object, forming a bounding box, and allocating a label to the bounding box. In one training sample \hat{x} we have n bounding boxes that refer to objects in the training sample. The objectness score $\hat{C}_i \in [0, 1][0, 1]$ which determines the presence of an object in an image, can be obtained by minimizing a binary cross-entropy (L_{BCE}) [74]. The objectness loss can then be formulated as:

$$L_{obj}(\hat{x}; O, W) = \sum_{i=1}^n [L_i L_{BCE}(1, \hat{C}_i) + (1 - L_i) L_{BCE}(0, \hat{C}_i)]. \tag{3}$$

In Eq. (3) it is assumed that $L_i = 1$ if there is an object in the i th candidate bounding box and $L_i = 0$ if the i th boundary box does not have any objects.

Regressing the bounding boxes: $(\hat{b}_i^x, \hat{b}_i^y)$ and $(\hat{b}_i^W, \hat{b}_i^H)$ denote the center, width, and height of the bounding box i and are obtained by minimizing a sum of box coordinates least square error (L_{SE}) as follows:

$$L_{bbox}(\hat{x}; O, W) = \sum_{i=1}^n L_i \left[L_{SE}(b_i^x, \hat{b}_i^x) + L_{SE}(b_i^y, \hat{b}_i^y) + L_{SE}\left(\sqrt{b_i^W}, \sqrt{\hat{b}_i^W}\right) + L_{SE}\left(\sqrt{b_i^H}, \sqrt{\hat{b}_i^H}\right) \right] \tag{4}$$

And finally, the object-type classification loss term is defined as:

$$L_{class}(\hat{x}; O, W) = \sum_{i=1}^n L_i \sum_{c=1}^C L_{BCE}(p_i^c, \hat{p}_i^c) \tag{5}$$

where K-class probability vector $\hat{p}_i = (\hat{p}_i^1, \hat{p}_i^2, \dots, \hat{p}_i^k)$ approximates the label of a box.

Thus, the overall loss function of the deep detector introduced by Chow et al. [45], is formulated by combining Eqs. (3), (4), and (5):

$$L_{\text{object-detector}}(\hat{x}; O, W) = L_{\text{obj}}(\hat{x}; O, W) + L_{\text{bbox}}(\hat{x}; O, W) + L_{\text{class}}(\hat{x}; O, W) \quad (6)$$

3.1 The targeted adversarial objectness gradient attacks (TOG) series of attacks

In this series of attacks, six attacks have been introduced by Chow et al. [45], who also made the relevant software available to the public. An evaluation of the obtained results shows that the TOG attacks have done well in different cases and were able to reduce the accuracy of the considered models on a variety of datasets. Here, we will touch briefly on the mathematical approaches used by the various models and also explain the attack strategies. The algorithm implemented in the TOG attacks is relatively simple.

The TOG carries out its adversarial attacks by reversing the training process. Chow et al. [45] are able to generate the x' images as the adversarial examples by using the following equation:

$$x' = \prod_{x', \epsilon} \left[x'_{t-1} - \alpha_{\text{TOG}} \Gamma \left(\frac{\partial L^*(x'_{t-1}; O^*, W)}{\partial x'_{t-1}} \right) \right] \quad (7)$$

Here, α_{TOG} denotes the learning rate of the attack, and $\Gamma(\cdot)$ is the sign function. By strategically manipulating L^* and finding the auxiliary target O^* , not only does the TOG support the random and arbitrary attacks, but it also generates 3 types of exclusive targeted attacks in order to fool its victims. We survey these attacks in the following subsections.

(a) *The TOG-untargeted attack* This is a type of random attack which tries to deceive the detection models in a way that they are unable to correctly detect the objects in various classes. This type of attack does not target an exact class of

objects. Also, the aim of this attack is not to use a wrongful technique, and such a mistake may occur due to the concealing of an object from an object detector, the allocation of the wrong label or the fabrication of a particular detection [75].

(b) *The TOG-vanishing attack* The main goal of this targeted attack is to create some noise in an image so that an object detector would be unable to detect and recognize any of the object classes in that image [76]. In fact, the main objective in this attack is to have an empty detection vector at the output of an object detector.

(c) *The TOG-fabrication attack* Contrary to the TOG-vanishing attack, the main objective of the TOG-fabrication attack is to add incorrect detections to the output detection vector. In this type of attack, at a detector's output, one can see an image with a lot of wrongly detected objects.

(d) *The TOG-mislabeling attack* In this attack, the existing positions of the objects in an image are correct, and they are detected correctly, but the object detector chooses the wrong labels for the images.

The outcomes of the TOG attacks on the example images and the performance of the object detector confronting these attacks are illustrated in Fig. 4.

3.2 The DAG attack

The DAG attack, with the full name of Dense Adversary Generation attack, has been devised for the object detection and semantic segmentation tasks [46]. According to the authors, this attack has a high transferability and is quite effective on numerous datasets and architectures. The algorithm for generating this type of attack is a simple one. In order to generate a DAG attack, the following procedure should be carried out:

Suppose \hat{x} is an image that contains n desired objects to be detected as $D = \{d_1, \dots, d_n\}$. Each desired object has been designated with a real class label of $l_n \in \{1, 2, \dots, C\}$ where C indicates the number of classes. The values of L will be assigned according to $L = \{l_1, l_2, \dots, l_n\}$.

For a specific task in a DNN, we will use $f_{l_n}(X, d_n) \in \mathbb{R}^C$ to show the classification score vector (before normalizing the maximum smooth function) on the n^{th} target of \hat{x} . In generating an adversarial example, we should make the prediction of

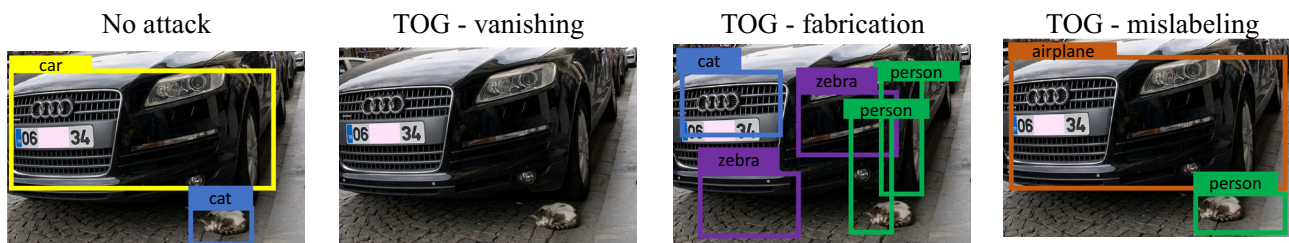


Fig. 4 The outcomes of the TOG attacks on the considered images

all the target objects erroneous, i.e., $\forall n, ARGmax_c \{f_c(X+r, d_n)\} \neq l_n$. Here, r indicates an adversarial perturbation which has been added to the image \hat{x} . Thus, an adversarial label l'_j is assigned for each target object and adversarial label vector $L' = \{l'_1, l'_2, \dots, l'_n\}$ is a set of adversarial labels. Therefore, the relevant loss function for all the targets will be

$$\text{loss}(X, D, L, L') = \sum_{n=1}^N [f_{l_n}(X, d_n) - f_{l'_n}(X, d_n)] \quad (8)$$

The value of loss can be minimized by causing an error in the prediction of each target object. This can be done by lowering the confidence level of the original correct class $f_{l_n}(X+r, d_n)$ and raising the confidence level of the considered incorrect class (the adversarial type) $f_{l'_n}(X+r, d_n)$.

In this approach, the gradient descent algorithm is employed to optimize the results. In the m th iteration, the current image (probably, after adding several perturbations) is displayed as X_m and the set of the correctly predicted target objects, i.e., the set of active targets, is obtained by mean of $\mathbf{D}_m = \{d_n | \text{argmax}_c \{f_c(X_m + d_n)\} = l_n\}$. Then, the gradient is computed based on the input data, and the sum of all the perturbations, which we call r_m , is determined. Thus, the final perturbation will be obtained as

$$r = \sum r_m \quad (9)$$

The DAG attack is demonstrated in Fig. 5.

3.3 The composite evaporate attack

This is a black-box type of attack and it can conceal the class of the target objects from an object detector without knowing anything about the network [47]. Figure 6 shows the overall strategy of the Evaporate attack.

The attacks generated in the Evaporate method are iterative in nature [53], and they will be repeated in a next steps if the obtained images in the previous steps are not good enough. In this approach, an attack is initiated by obtaining the adversarial example x' through the following optimization equation:

$$\min_{x'} L(x') = d(x', x) - \delta(D(x')) \quad (10)$$

Here, $d(x', x)$ denotes the MSE distance, and $\delta(D(x'))$ indicates an adversarial criterion. The value of the criterion will be zero when the conditions of a satisfactory attack are satisfied, and it can go down to $-\infty$ when such conditions are not met. According to [47], this attack has been able to fool the YOLOv3 model 84% of the time.

Fig. 5 Demonstration of the DAG attack

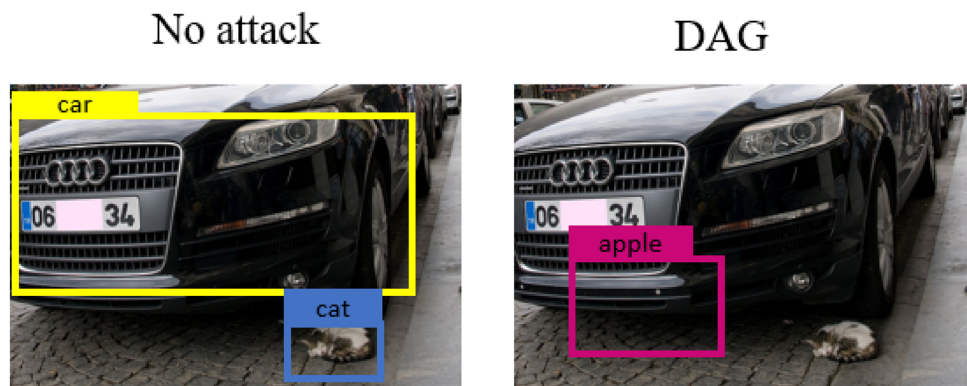


Fig. 6 The overall design of an Evaporate attack

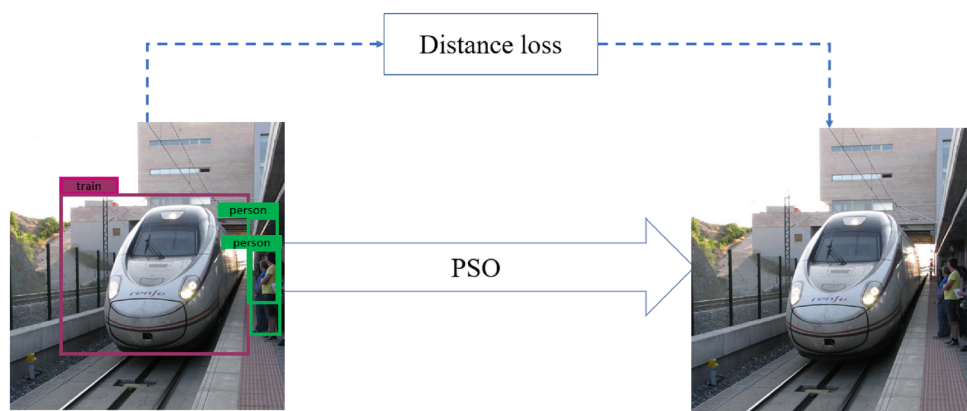
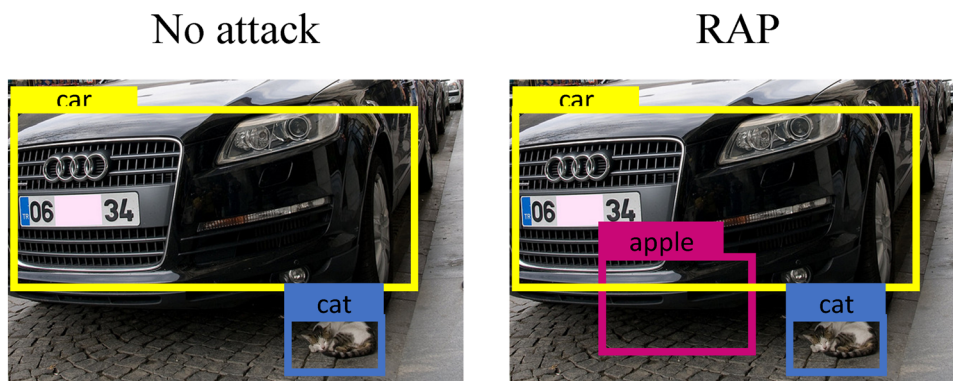


Fig. 7 A demonstration of the RAP attack



3.4 The DeepFool attack

This type of attack is carried out by adding a minimum number of perturbations. In this method, a clean image is fed into an iterative algorithm which adds some perturbations to it in every step. Eventually, by reaching a fooling threshold, the iterative algorithm is stopped. This algorithm is repeated to the point that an object detector changes its decision with respect to the original detection. This attack can be a universal attack and is used in many types of DNNs [48].

3.5 The RAP attack

This attack, which is called the Robust Adversarial Perturbation, is a black-box type of attack. This attack is also designed based on the solution of an optimization equation, and it will go on until the intended effect is achieved. In this scheme, the proposal-based object detectors and the instantaneous segmentation algorithms are attacked by adding minimal adversarial noises to an input image. In this approach, with an input image and a pre-trained RPN, a special objective function is designed, and then a technique based on the iterative gradients is employed to optimize the objective function with respect to the input image [49].

In this method, the generation of the adversarial perturbations has been considered as an optimization problem. The outcome of this attack is displayed in Fig. 7.

3.6 The generative adversarial training (GAT) method

This method is designed based on accurate identification of weaknesses and strengths of the target network. This algorithm is designed to be able to repeatedly eliminate the weaknesses of the adversarial example and improve its strengths to make the adversarial example more efficient.

In the course of each training step, the GAT scheme learns to produce the best perturbation for each input. Simultaneously, a classification network is trained by the GAT to

correctly classify the original and the adversarial examples [50]. The loss function of the GAT method is expressed as

$$L_F = \alpha \cdot J(\theta_f, x, y) + (1 - \alpha) \cdot J(\theta_f, x + G(\Delta), y), \quad (11)$$

where

$$J(\theta_f, x, y) = -\log F(X; \theta_f)_y \quad (12)$$

Typical values of α and k could be 0.5 and 1.0, respectively. Also, F is a classifying network. The schematic of this technique is illustrated in Fig. 8.

4 The different types of defense

4.1 The adversarial training method

In this approach, the adversarial training [54] is employed as a defensive mechanism against the adversarial attacks. The basic strategy of this method is to use the perturbed images to train a network. Zhang et al. [50] have presented the following formula as an adversarial training method for achieving robust object detectors.

$$\min_{\theta} [\max_{\bar{x} \in S_{cls} \cup S_{loc}} L(f_{\theta}(\bar{x}), y_k, b_k)] \quad (13)$$

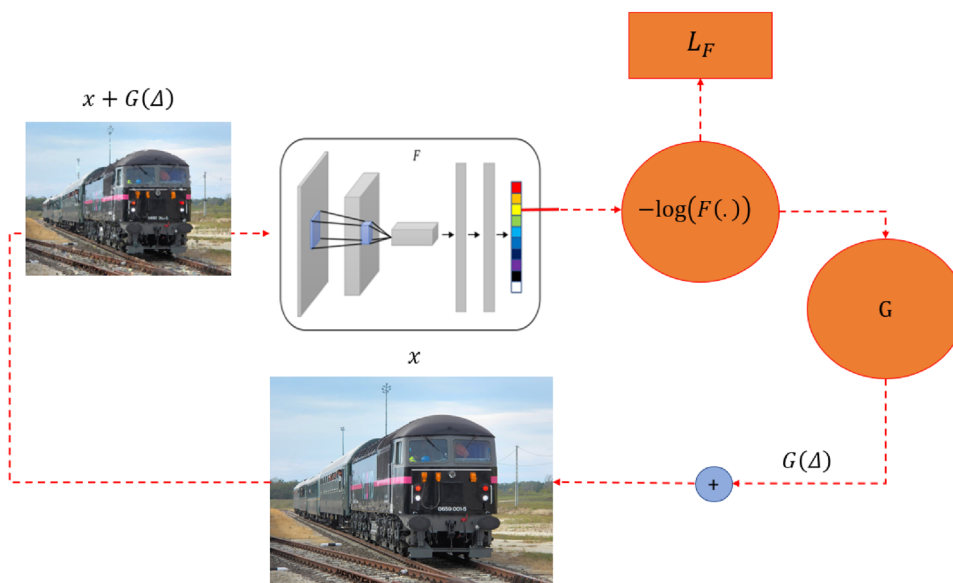
in which

$$s_{cls} = \{\bar{x} | \arg \max_{\bar{x} \in S_x} loss_{cls}(f(\bar{x}), \{y_k\})\} \quad (14)$$

$$s_{loc} = \{\bar{x} | \arg \max_{\bar{x} \in S_x} loss_{loc}(f(\bar{x}), \{b_k\})\} \quad (15)$$

As one of the first attempts to robustify object detectors against the adversarial attacks, this method has been able to achieve good results. Zhang et al. [51] have tested this technique on the PASCAL VOC and MS COCO datasets.

Fig. 8 The GAT method



Amirkhani and Karimi [65] also tested this method on different architectures, and on the average, it was able to improve the adversarial accuracy of the models by about 20%.

4.2 The ADNet method

The detection strategy of this method is based on the adversarial detection network (ADNet). The ADNet learns the detection abilities from the input images in a hierarchical fashion. In this process, the input images pass through the convolutional and the composite layers. The first convolutional layer has 6 ability maps of size 5×5 and a step size of 1. Next, the ADNet performs the secondary sampling by using a 2×2 size filter and a step size of 2. Then, there is a second convolutional layer with 16 feature maps of size 5×5 and a step size of 1. In this layer, only 10 of the 16 feature maps are connected to the 6 feature maps of the preceding layer. The fourth layer in the proposed network is again a medium cumulative layer with a 2×2 size filter and a step size of 2. This is similar to the second layer, except that it has 16 feature maps. The fifth layer is a fully connected convolutional layer with 120 feature maps of size 1×1 . Each of the 120 elements in the FC5 is connected to all the 400 nodes ($5 \times 5 \times 16$) in the fourth layer. The sixth layer is a fully connected layer with 84 neurons. Finally, we have one fully connected SoftMax output layer with 2 possible values corresponding to the perturbed images or the original images. Our convolutional layers use the “ReLU” activation function and the “Adam” optimization algorithm.

A desirable characteristic of the ADNet method is that it can fool deep models in the test phase [55]. Functioning as a separate module, it is able to detect the adversarial examples

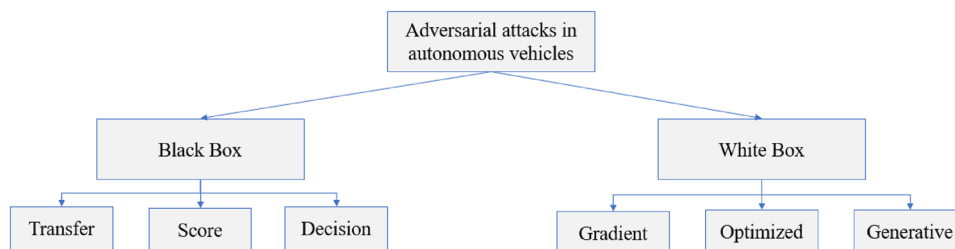
independently of a considered model; it can also act as a hidden component of an overall intelligent system. This makes the ADNet inherently strong against the attacks on itself. Note that this feature of the ADNet is contrary to most of the existing decision networks that have to rely on the internal states of a network during the test phase, which makes them exposed to a potential attackers. It is also worth noting that these methods are also incapable of dealing with the pixel-level attacks.

The only dependency of the ADNet is on a considered network during the training phase, in which the adversary examples for a model are produced by attacking the model. In this work, the ResNet has been employed to train the ADNet. However, any other network or group of networks could be used to train the ADNet and to further improve its ability of detecting the adversarial examples [52].

4.3 The JPG compression method

This method has demonstrated that by reducing the data volume of images via converting them to the JPG files, the effects of the adversarial attacks on the detection ability of the DNNs can sometimes be eliminated. Of course, this approach alone cannot be considered as a complete defensive strategy [56]. The authors of this paper have pointed out that most of the image classification datasets contain the images in the JPG format. In view of this observation, they have studied the effects of the JPG compression on the perturbations generated by the FGSM [63]. They have reported that the JPG compression technique can significantly reverse the loss of the classification accuracy for the FGSM perturbations [63]. That being said, heavy compression itself can reduce the performance of neural networks.

Fig. 9 A brief review of the adversarial attacks in the autonomous vehicles



4.4 The parseval networks

Cisse et al. [64] have presented the Parseval networks as a defensive method against the adversarial attacks. These networks use a Lipschitz constant. Since a network can be considered as a combination of functions, it can be robustified against the small input perturbations by keeping the Lipschitz constant small for these functions. They have developed this method by controlling the position norm of the network weight matrices and parameterizing them by means of the hard Parseval frameworks; thus, they have called their method, the Parseval networks.

4.5 Gabor convolutional layers

Amirkhani and Karimi [65] recently proposed a new method in order to robustify the object detectors against adversarial attacks based on Gabor convolution layers. In this method, the images are first decomposed into their own RGB channels. Then they enter a Gabor filter bank. Due to its high ability to extract low-level image features, Gabor filters can increase network robustness at this stage. The authors of this study have been able to provide considerable improvements on the performance of object detection models against images infected with adversarial attacks. In [65], five robust models of object detection against adversarial attacks are presented and these models have been evaluated using different attacks. The method presented in this paper has been able to improve the performance of object detectors against adversarial attacks up to 50%.

5 The application scenarios

5.1 Adversarial robustness in autonomous vehicles

Because of the vital importance of object detection in autonomous vehicles, we will discuss it separately in this section.

5.1.1 Adversarial attacks in autonomous vehicles

In autonomous vehicles, by adding small intangible perturbations, deep learning models are fooled into making wrong detections and predictions [65]. In self-driving car applications, depending on the capability of an attacker, these attacks are divided into the white-box and the black-box attacks. In the white-box attacks, the attackers have all the information about the model being attacked. This information may include the training and the validation data, the model's architecture and all its parameters, the way the model is trained, and the status of the model's gradient during training [77]. Conversely, the black-box models have no information about the models [78]. The adversarial attacks in the autonomous vehicles are briefly reviewed in Fig. 9. In general, there are two types of adversarial attacks: the fleeing attacks and the poisoning attacks. The fleeing or deceptive attacks occur during the inference process, and the poisoning attacks take place during the model training. These attacks were initially tested on classifier models.

The white-box methods White-box attacks are designed with full knowledge of the target model and its parameters. For example, three different white-box methods for producing the adversarial examples are introduced below:

(a) The Gradient-Based Method: In this method, all the attacks such as those in [79] and [80] are based on the Fast Gradient Sign Method (FGSM). In these methods, the adversarial examples are created directly by increasing the value of the cost function gradient for every pixel of an original image.

(b) The Optimization-Based Methods: These techniques ([81] and [82]) generate the adversarial examples by solving an optimization problem such as the following equation.

$$\arg \min \alpha \left\| x - x' \right\|_p + L(J_{\theta, c}(x')) \quad (16)$$

In the first part of this equation, L denotes the distance between the original and the adversarial images and the second part is the cost function restriction of the adversarial image [83].

(c) The Generative Methods: These types of attacks ([84]) exploit the advantages of the generative methods to produce the adversarial examples. These techniques create a generative model ζ by optimizing the following function.

$$L = L_\gamma + \alpha L_\zeta \quad (17)$$

In this equation, L_γ represents the cross-entropy cost function for the adversarial examples and the target object class, and L_ζ indicates the degree of similarity between the adversarial examples and the original images.

The black-box method In the black-box attacks, the attackers have no information about the model being attacked; they can only feed an input to the model and then evaluate its output [85]. Three example approaches that are used in the black-box attacks to generate the adversarial examples are as follows:

(a) The Transformation Method: It has been shown that the adversarial examples that are produced for a model by this approach are more effective than those generated by the other methods [82]. Therefore, in this approach, the attackers can use the input/output results to create a model similar to the target model and then apply the white-box techniques to generate the adversarial examples for this model. They can then use these adversarial examples to attack the target model.

(b) The Score Method: In this approach, the score of the gradient output can be estimated by knowing the target model results and accuracy and, based on this information, the adversarial examples can be generated [86].

(c) The Decision Method: In this approach, a model's final results are used to generate the adversarial examples with large random perturbations. Then the perturbations are reduced in magnitude so that they go along with the characteristic of the adversarial examples, i.e., the intangible perturbations.

It is worth noting that the black-box attacks are more realistic than the white-box ones. The white-box attacks need the full information about the driving models of the autonomous vehicles, which is not available for most of the commercial vehicles.

In [87], a real-world adversarial attack on the traffic signs is implemented. Zhang et al. [88] presented a physical camouflage for an adversarial attack that was similar to the camouflage in the simulation programs. This technique performed as good as the detectors in leading to wrong detections. A perturbed stop sign in [89] could not be detected by the best detectors, such as the one in [90]. A technique called the DeepBillboard has been formulated in [91], which causes a deviation in the steering of the autonomous vehicles from their original paths by creating adversarial advertising billboards. This attack causes a maximum deviation of 26.44° in the steering of a vehicle. In [92], an end-to-end driving model was attacked by means of the adversarial perturbations in the driving environment. This attack caused the vehicle to crash in the CARLA simulator. A decision method for producing the adversarial textures for attacking the autonomous vehicle systems was introduced in [93]. This method leads to the wrong detection in these vehicles.

5.1.2 Adversarial defenses in autonomous vehicles

There are numerous defensive methods against the adversarial attacks on classification models. However, many of these approaches cannot be applied to the regression models used in autonomous vehicles. Figure 10 illustrates a breakdown of the defensive techniques for the adversarial attacks in autonomous vehicles.

In the following, we will review some of the defensive techniques against the adversarial attacks in the autonomous vehicles.

The detection-based approaches In these methods, robust models try to detect the presence of the potential attacks. Zheng et al. [94] have presented a detection-based method in which an iterative algorithm detects the presence of an attack in an input sample and tries to robustify the network with respect to this attack. The iterative methods are interesting approaches for robustifying the models against the adversarial attacks, but their effectiveness in the white-box or the image-based attacks is questionable. In some works [95, 96], the responsibility for detecting the presence of attacks has been laid on the preprocessing systems that exist in the

Fig. 10 Different types of defenses against the adversarial attacks in autonomous vehicles

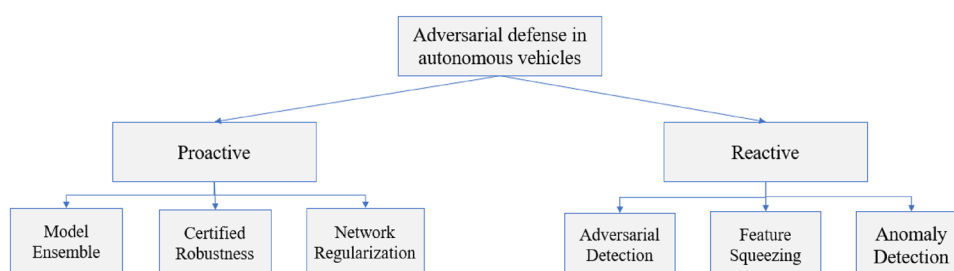


Table 1 Comparing the adversarial attacks explored in this paper

Attack name	Year of presentation	mAP reduction rate	Attack objective	Primary target model
TOG [45]	2020	64.73	Has both targeted and untargeted attack series	Object detection
DAG [46]	2017	63.81	Targeted/Untargeted type of attack	Semantic segmentation
Evaporate attack [47]	2020	48.00	targeted	Object detection
DeepFool attack [48]	2016	–	untargeted	Classification
RAP [49]	2018	62.59	targeted	Object detection and segmentation
U-DOS [60]	2021	61.2	targeted	Object detection
AO ² AM [61]	2021	53.3	untargeted	Object detection
Bidirectional [62]	2022	55.1	targeted	Object detection

autonomous vehicles, and these systems are expected to perform satisfactorily in detecting the adversarial attacks [97].

The training-based approaches In these approaches, like in the training-based methods in the field of object detection, the adversarial training technique is employed to robustify the autonomous vehicles against the adversarial attacks [57]. In the training phase of the adversarial training process, a combination of the clean and perturbed images is given to a network for the training purposes, and since the network has already been exposed to the adversarial examples, its adversarial accuracy is expected to improve [98]. Yan et al. [99] have presented an efficient training-based method. In this approach, first, the input images are perturbed by different adversarial attacks and then combined with the clear images, and the new dataset thus obtained is used during the network training.

5.2 Face recognition/detection

Face recognition is one of the most important applications of object detection in the deep learning models. Face recognition is used in a vast spectrum of human–computer interfaces, cameras, and biometric detectors [100]. Therefore, adversarial attacks in face recognition applications are studied and surveyed in the following subsections.

5.2.1 Adversarial attacks in face recognition

Adversarial attacks in the face recognition applications have been investigated in many works. For example, using an adversarial attack generating network, a method of DNNs was presented in [101]. This technique is based on solving an optimization problem that can be scaled and applied to other networks as well. This method was applied specifically

Table 2 Comparing the defense techniques considered in this paper

Defense name	Year of presentation	Defense type
Adversarial training [51]	2019	Iterative
ADNet [52]	2020	Iterative
JPG compression [63]	2016	One shot
The Parseval networks [64]	2017	One shot
Gabor convolutional layers [65]	2022	Iterative

on the FRCNN face recognition model and was able to considerably reduce the precision of the network on the 300-W dataset (the effective precision of the FRCNN was reduced to 0.5%).

5.2.2 Defense strategies in face recognition

Different defensive techniques are also employed in the face recognition models. For example, to prove the efficacy of its adversarial attack, the defense strategy presented in Sect. 4.3 has been adopted in [101]. This method, which uses image compression to boost the resistance against adversarial attacks, has been able to enhance the network precision by 5%. Various defense strategies have been presented in this field, and the defenses outlined in Sect. 3 can be extended to this section as well.

6 Conclusion

This paper surveyed the adversarial attacks, defenses, and the related research works in the fields of object detection and autonomous vehicles. Despite the high precision of the DNNs in various computer vision tasks, these networks are

vulnerable against the small imperceptible input perturbations and produce totally different outputs when exposed to such disturbances. The formulation of the effective adversarial attacks and appropriate defenses against these attacks has become an important subject in the deep learning research articles. In this review paper, we have introduced and compared the most significant attacks and defenses in the fields of object detection and autonomous vehicles. The current deep learning techniques can be easily attacked, but owing to the tremendous research efforts in this field, it is hoped that in the near future, the deep learning methods will be able to achieve great robustness against the devised adversarial attacks.

Declarations

Conflict of interest All authors declare that they have no conflict of interest.

References

- Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of Advance Neural Information Processing Systems, pp. 1097–1105 (2012)
- Dodge, S., Karam, L.: Understanding how image quality affects deep neural networks. Eighth International Conference on Quality of Multimedia Experience (QoMEX) **2016**, 1–6 (2016). <https://doi.org/10.1109/QoMEX.2016.7498955>
- A. R. Sharma and P. Kaushik, "Literature survey of statistical, deep and reinforcement learning in natural language processing," *2017 International Conference on Computing, Communication and Automation (ICCCA)*, 2017, pp. 350–354, doi: <https://doi.org/10.1109/ICCA.2017.8229841>.
- Hu, H., Tang, B., Gong, X., Wei, W., Wang, H.: Intelligent fault diagnosis of the high-speed train with Big Data based on deep neural networks. *IEEE Trans. Industr. Inf.* **13**(4), 2106–2116 (2017)
- Deng, L., Wu, H., Liu, H.: D2VCB: a hybrid deep neural network for the prediction of in-vivo protein-DNA binding from combined DNA sequence. *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* **2019**, 74–77 (2019). <https://doi.org/10.1109/BIBM47256.2019.8983051>
- Ackerman, E.: How Drive.ai is Mastering Autonomous Driving With Deep Learning, Dec. 2017, [online]. Available: <https://spectrum.ieee.org/cars-that-think/transportation/self-driving/how-driveai-is-mastering-autonomous-driving-with-deep-learning>
- Cococcioni, M., Rossi, F., Ruffaldi, E., Saponara, S., Dupont de Dinechin, B.: Novel arithmetics in deep neural networks signal processing for autonomous driving: challenges and opportunities. *IEEE Signal Process. Mag.* **38**(1), 97–110 (2021)
- Cococcioni, M., Ruffaldi, E., Saponara, S.: Exploiting posit arithmetic for deep neural networks in autonomous driving applications. *International Conference of Electrical and Electronic Technologies for Automotive* **2018**, 1–6 (2018). <https://doi.org/10.23919/EETA.2018.8493233>
- Okuyama, T., Gonsalves, T., Upadhyay, J.: Autonomous driving system based on deep Q learning. *International Conference on Intelligent Autonomous Systems (ICoIAS)* **2018**, 201–205 (2018). <https://doi.org/10.1109/ICoIAS.2018.8494053>
- Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks, 2016, [online]. Available: <https://arxiv.org/abs/1608.06993>
- Szegedy, C., Vincent, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
- Sze, V., Chen, Y., Yang, T., Emer, J.S.: Efficient processing of deep neural networks: a tutorial and survey. *Proc. IEEE* **105**(12), 2295–2329 (2017)
- Xu, J., Wang, B., Li, J., Hu, C., Pan, J.: Deep learning application based on embedded GPU. *First International Conference on Electronics Instrumentation & Information Systems (EIIS)* **2017**, 1–4 (2017). <https://doi.org/10.1109/EIIS.2017.8298723>
- Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding, 2014, [online]. Available: <https://arxiv.org/abs/1408.5093>
- Akhtar, N., Mian, A.: Threat of adversarial attacks on deep learning in computer vision: a survey. *IEEE Access* **6**, 14410–14430 (2018)
- Deng, Y., Zheng, X., Zhang, T., Chen, C., Lou, G., Kim, M.: An Analysis of adversarial attacks and defenses on autonomous driving models. *IEEE International Conference on Pervasive Computing and Communications (PerCom)* **2020**, 1–10 (2020). <https://doi.org/10.1109/PerCom45495.2020.9127389>
- Rajan, J.P., Rajan, S.E., Matris, R.J., Panigarhi, B.K.: Fog computing employed computer aided cancer classification system using deep neural network in internet of things based health-care system. *Image Sign. Process.* (2019). <https://doi.org/10.1007/s10916-019-1500-5>
- Su, H., Qi, W., Yang, C., Sandoval, J., Ferrigno, G., Momi, E.D.: Deep neural network approach in robot tool dynamics identification for bilateral teleoperation. *IEEE Robot. Autom. Lett.* **5**(2), 2943–2949 (2020)
- Zhu, J., et al.: Urban traffic density estimation based on ultrahigh-resolution UAV video and deep neural network. *IEEE J. Select. Top. Appl. Earth Observ. Remote Sens.* **11**(12), 4968–4981 (2018)
- LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**, 436–444 (2015)
- Mollahosseini, A., Chan, D., Mahoor, M.H.: Going deeper in facial expression recognition using deep neural networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)* **2016**, 1–10 (2016). <https://doi.org/10.1109/WACV.2016.7477450>
- Seifer, C., Aamir, A., Balagopalan, A., Jain, D., Sharma, A., Grottel, S., Gumhold, S.: Visualizations of deep neural networks in computer vision: a survey. *Transparent Data Mining Big Small Data* **32**, 123–144 (2017)
- Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
- Pérez, J.C., Alfarra, M., Jeanneret, G., Bibi, A., Thabet, A., Ghanem, B., Arbeláez, P.: Gabor layers enhance network robustness. In: *Computer Vision – ECCV 2020 Lecture Notes in Computer Science*, pp. 450–466 (2020)
- Aprilyone, M., Kinoshita, Y., Kiyu, H.: Adversarial robustness by one Bit double quantization for visual classification. *IEEE Access* **7**, 177932–177943 (2019)
- Liskowski, P., Krawiec, K.: Segmenting retinal blood vessels with deep neural networks. *IEEE Trans. Med. Imaging* **35**(11), 2369–2380 (2016)
- Arnab, A., Miksik, O., Torr, P.H.S.: On the robustness of semantic segmentation models to adversarial attacks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 888–897 (2018)

28. Arora, S., Bhatia, M.P.S., Mittal, V.: A robust framework for spoofing detection in faces using deep learning. *Vis. Comput.* (2021). <https://doi.org/10.1007/s00371-021-02123-4>
29. Liu, Z., Xiang, Q., Tang, J., Wang, Y., Zhao, P.: Robust salient object detection for RGB images. *Vis. Comput.* **36**, 1823–1835 (2020)
30. Zhou, X., Xie, L., Zhang, P., Zhang, Y.: An ensemble of deep neural networks for object tracking. *IEEE International Conference on Image Processing (ICIP)* **2014**, 843–847 (2014). <https://doi.org/10.1109/ICIP.2014.7025169>
31. Shah, M., Kapdi, R.: Object detection using deep neural networks. *International Conference on Intelligent Computing and Control Systems (ICICCS)* **2017**, 787–790 (2017). <https://doi.org/10.1109/ICCONS.2017.8250570>
32. Li, G., Yu, Y.: Contrast-oriented deep neural networks for salient object detection. *IEEE Trans. Neural Netw. Learn. Syst.* **29**(12), 6038–6051 (2018)
33. Liu, D., et al.: Video object detection for autonomous driving: motion-aid feature calibration. *Neurocomputing* **409**, 1–11 (2020)
34. Cui, Y., et al.: TF-blender: temporal feature blender for video object detection. In: 2021 IEEE International Conference on Computer Vision (ICCV) (2021)
35. Liu, D., et al.: Sg-net: spatial granularity network for one-stage video instance segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9816–9825 (2021)
36. Li, X., et al.: DeepSaliency: multi-task deep neural network model for salient object detection. *IEEE Trans. Image Process.* **25**(8), 3919–3930 (2016)
37. Wu, F., Jin, G., Gao, M., He, Z., Yang, Y.: Helmet detection based on improved YOLO V3 deep Model. In: *IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*, Canada, pp. 363–368 (2019)
38. Nsaif, A.K., et al.: FRCNN-GNB: cascade faster R-CNN With gabor filters and Naïve Bayes for enhanced eye detection. *IEEE Access* **9**, 15708–15719 (2021)
39. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: Ssd: single shot multibox detector. *European Conference on Computer Vision (ECCV)* (2016)
40. Xu, H., Lv, X., Wang, X., Ren, Z., Bodla, N., Chellappa, R.: Deep regionlets: blended representation and deep learning for generic object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(6), 1914–1927 (2021)
41. Han, J., Zhang, D., Hu, X., Guo, L., Ren, J., Wu, F.: Background prior-based salient object detection via deep reconstruction residual. *IEEE Trans. Circuits Syst. Video Technol.* **25**(8), 1309–1321 (2015)
42. Szegedy, C., et al.: Intriguing properties of neural networks, 2014, [online]. Available: <https://arxiv.org/abs/1312.6199>
43. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples, 2015, [online]. Available: <https://arxiv.org/abs/1412.6572>
44. Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 86–94 (2017)
45. Chow, K.H., Liu, L., Loper, M., Bae, J., Gursoy, M.E., Truex, S., Wei, W., Wu, Y.: Adversarial objectness gradient attacks in real-time object detection systems (2020). [Online]. Available: https://khchow.com/media/TPS20_TOG.pdf
46. Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., Yuille, A.: Adversarial examples for semantic segmentation and object detection. In: 2017 IEEE International Conference on Computer Vision (ICCV) (2017)
47. Wang, Y., Tan, Y., Zhang, W., Zhao, Y., Kuang, X.: An adversarial attack on DNN-based black-box object detectors. *J. Netw. Comput. Appl.* **161**, 102634 (2020)
48. Moosavi-Dezfooli, S., Fawzi, A., Frossard, P.: DeepFool: a simple and accurate method to fool deep neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582 (2016)
49. Li, Y., Tian, D., Bian, X., Lyu, S.: Robust adversarial perturbation on deep proposal-based models. In: *British Machine Vision Conference (BMVC)* (2018)
50. Lee, H., Han, S., Lee, J.: Generative adversarial trainer: defense to adversarial perturbations with GAN, 2017, [online]. Available: <https://arxiv.org/abs/1705.03387>
51. Zhang, H., Wang, J.: Towards adversarially robust object detection. In: *Proceedings of IEEE International Conference on Computer Vision*, pp. 421–430 (2019)
52. Shah, S.A.A., Bougre, M., Akhtar, N., Bennamoun, M., Zhang, L.: Efficient detection of pixel-level adversarial attacks. *IEEE International Conference on Image Processing (ICIP)* **2020**, 718–722 (2020). <https://doi.org/10.1109/ICIP40778.2020.9191084>
53. Han, D., et al.: DeepAID: Interpreting and Improving Deep Learning-based Anomaly Detection in Security Applications, 2021, [online]. Available: <https://arxiv.org/abs/2109.11495>
54. Mahmood, F., Chen, R., Durr, N.J.: Unsupervised reverse domain adaptation for synthetic medical images via adversarial training. *IEEE Trans. Med. Imaging* **37**(12), 2572–2581 (2018)
55. Husnoo, M.A., Anwar, A.: Do not get fooled: defense against the one-pixel attack to protect IoT-enabled deep learning systems. *Ad Hoc Netw.* (2021). <https://doi.org/10.1016/j.adhoc.2021.102627>
56. Prakash, A., Moran, N., Garber, S., DiLillo, A., Storer, J.: Protecting JPEG images against adversarial attacks. *Data Compression Conference* **2018**, 137–146 (2018)
57. Liu, A., Liu, X., Yu, H., Zhang, C., Liu, Q., Tao, D.: Training robust deep neural networks via adversarial noise propagation. *IEEE Trans. Image Process.* **30**, 5769–5781 (2021)
58. Manville, K., Merkhofer, E., Strickhart, L., Walmer, M.: Apricot: a dataset of physical adversarial attacks on object detection. In: *Proceedings of Eur. Conference on Computer Vision*, in *Lecture Notes in Computer Science*, vol. 12366. Springer, Cham, pp. 35–50 (2020). https://doi.org/10.1007/978-3-030-58589-1_3
59. Everingham, M., et al.: The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vis.* **111**(1), 98–136 (2015)
60. Li, D., Zhang, J., Huang, K.: Universal adversarial perturbations against object detection. *Pattern Recogn.* **110**, 107584 (2021)
61. Xiao, Y., Pun, C., Liu, B.: Fooling deep neural detection networks with adaptive object-oriented adversarial perturbation. *Pattern Recognit.* **115**, 107903 (2021)
62. Li, X., Jiang, Y., Liu, C., Liu, S., Luo, H., Yin, S.: Playing against deep-neural-network-based object detectors: a novel bidirectional adversarial attack approach. *IEEE Trans. Artif. Intell.* **3**(1), 20–28 (2022)
63. Dziugaite, G.K., Ghahramani, Z., Roy, D.M.: A study of the effect of JPG compression on adversarial images, 2016, [online]. Available: <https://arxiv.org/abs/1608.00853>
64. Cisse, M., Bojanowski, P., Grave, E., Dauphin, Y., Usunier, N.: Parseval networks: improving robustness to adversarial examples. In: *International Conference on Machine Learning*, pp. 854–863. PMLR (2017)
65. Amirkhani, A., Karimi, M.P.: Adversarial defenses for object detectors based on Gabor convolutional layers. *Vis. Comput.* **38**(6), 1929–1944 (2022)
66. Lu, J., Issaranon, T., Forsyth, D.: SafetyNet: detecting and rejecting adversarial examples robustly. In: *Proceedings of the IEEE International Conference on Computer Vision* (2017)

67. Zhang, Y., Tian, X., Li, Y., Wang, X., Tao, D.: Principal component adversarial example. *IEEE Trans. Image Process.* **29**, 4804–4815 (2020)
68. Miyato, T., Maeda, S., Koyama, M., Ishii, S.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(8), 1979–1993 (2019)
69. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: *Proceedings of ACM Asia Conference on Computer Communication and Security*, pp. 506–519 (2017)
70. Yuan, X., He, P., Zhu, Q., Li, X.: Adversarial examples: attacks and defenses for deep learning. *IEEE Trans. Neural Netw. Learn. Syst.* **30**(9), 2805–2824 (2019)
71. Zhang, W.: Generating adversarial examples in one shot with image-to-image translation GAN. *IEEE Access* **7**, 151103–151119 (2019)
72. Alarifari, R., Alberti, G.S., Gauksson, T.: ADEF: an iterative algorithm to construct adversarial deformations. In: *Proceedings of the International Conference on Learning Representations (ICLR)* (2019)
73. Su, J., Vargas, D.V., Sakurai, K.: One pixel attack for fooling deep neural networks. *IEEE Trans. Evol. Comput.* **23**(5), 828–841 (2019)
74. Wu, X., Zhang, S., Zhou, Q., Yang, Z., Zhao, C., Latecki, L.J.: Entropy minimization versus diversity maximization for domain adaptation. *IEEE Trans. Neural Netw. Learn. Syst.* (2021). <https://doi.org/10.1109/TNNLS.2021.3110109>
75. Karimi, M.P., Amirkhani, A., Shokouhi, S.B.: Robust object detection against adversarial perturbations with gabor filter. In: *2021 29th Iranian Conference on Electrical Engineering (ICEE)*, pp. 187–192 (2021)
76. Wang, L., Yoon, K.-J.: PSAT-GAN: efficient adversarial attacks against holistic scene understanding. *IEEE Trans. Image Process.* **30**, 7541–7553 (2021)
77. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519 (2017)
78. Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: Ensemble adversarial training: attacks and defenses (2017) [online]. Available: <https://arxiv.org/abs/1705.07204>
79. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014)
80. Szegedy, C., et al.: Intriguing properties of neural networks (2013), [online]. Available: *arXiv preprint arXiv:1312.6199*
81. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: *2017 IEEE Symposium on Security and Privacy (sp)*, IEEE, pp. 39–57 (2017)
82. Liu, Y., Chen, X., Liu, C., Song, D.: Delving into transferable adversarial examples and black-box attacks (2016) [online]. Available: <https://arxiv.org/abs/1611.02770>
83. Poursaeed, O., Katsman, I., Gao, B., Belongie, S.: Generative adversarial perturbations. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4422–4431 (2018)
84. Xiao, C., Li, B., Zhu, J.-Y., He, W., Liu, M., Song, D.: Generating adversarial examples with adversarial networks," 2018, [online]. Available: <https://arxiv.org/abs/1801.02.2018.610>
85. Poudel, B., Li, W.: Black-box adversarial attacks on network-wide multi-step traffic state prediction models. *IEEE International Intelligent Transportation Systems Conference (ITSC)* **2021**, 3652–3658 (2021)
86. Aung, A.M., Fadila, Y., Gondokaryono, R., Gonzalez, L.: Building robust deep neural networks for road sign detection, " 2017, [online]. Available: <https://arxiv.org/abs/1712.09327>
87. Sitawarin, C., Bhagoji, A.N., Mosenia, A., Mittal, P., Chiang, M.: Rogue signs: Deceiving traffic sign recognition with malicious ads and logos," 2018, [online]. Available: <https://arxiv.org/abs/1801.02780>
88. Zhang, Y., Foroosh, H., David, P., Gong, B.: "CAMOU: learning physical vehicle camouflages to adversarially attack detectors in the wild. In: *International Conference on Learning Representations* (2018)
89. He, K., Gkioxari, G., Dollár, P., Girshick, R.: "Mask r-cnn,". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969 (2017)
90. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. *arXiv preprint arXiv:1804.02767* (2018)
91. Zhou, H., et al.: Deepbillboard: Systematic physical-world testing of autonomous driving systems. In: *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*. IEEE, pp. 347–358 (2020)
92. Bolor, A., He, X., Gill, C., Vorobeychik, Y., Zhang, X.: Simple physical adversarial examples against end-to-end autonomous driving models. In: *2019 IEEE International Conference on Embedded Software and Systems (ICCESS)*, IEEE, pp. 1–7 (2019)
93. Yang, J., Bolor, A., Chakrabarti, A., Zhang, X., Vorobeychik, Y.: Finding Physical Adversarial Examples for Autonomous Driving with Fast and Differentiable Image Compositing (2020) [online]. Available: <https://arxiv.org/abs/2010.08844>
94. Zheng, Z., Hong, P.: Robust detection of adversarial attacks by modeling the intrinsic properties of deep neural networks. In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 7924–7933 (2018)
95. Deng, Y., Zhang, T., Lou, G., Zheng, X., Jin, J., Han, Q.-L.: Deep Learning-based autonomous driving systems: a survey of attacks and defenses. *IEEE Trans. Industr. Inf.* **17**(12), 7897–7912 (2021)
96. Kyrkou, C., et al.: Towards artificial-intelligence-based cybersecurity for robustifying automated driving systems against camera sensor attacks. *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)* **2020**, 476–481 (2020)
97. Zheng, X., Julien, C., Podorozhny, R., Cassez, F., Rakotoarivelo, T.: Efficient and scalable runtime monitoring for cyber-physical system. *IEEE Syst. J.* **12**(2), 1667–1678 (2016)
98. Mahmood, F., et al.: Deep adversarial training for multi-organ nuclei segmentation in histopathology images. *IEEE Trans. Med. Imaging* **39**(11), 3257–3267 (2020)
99. Yan, Z., Guo, Y., Zhang, C.: Deep defense: training dnns with improved adversarial robustness. *Advances in Neural Information Processing Systems* (2018)
100. Kumar, A., Kaur, A., Kumar, M.: Face detection techniques: a review. *Artif. Intell. Rev.* **52**(2), 927–948 (2019)
101. Bose, A.J., Aarabi, P.: Adversarial attacks on face detectors using neural net based constrained optimization. In: *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSp)*, pp. 1–6 (2018). <https://doi.org/10.1109/MMSP.2018.8547128>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



Abdollah Amirkhani received the M.Sc. and Ph.D. degrees (with honors) in electrical engineering from Iran University of Science and Technology (IUST), Tehran, in 2012 and 2017, respectively. He earned the Outstanding Student Award (2015) from the First Vice President of Iran. In 2016, he was conferred award by the Ministry of Science, Research and Technology. He is an Assistant Professor in the school of automotive engineering at IUST.

He is the Associate Editor of the "Engineering Science and Technology, an International Journal." He has been actively involved in several National R&D projects, related to the development of new methodologies and learning algorithms based on AI techniques. He is a senior member of the IEEE and his research interests are in machine vision, fuzzy cognitive maps, data mining, and machine learning.



Mohammad Parsa Karimi received the B.Sc. degree in Electrical and Electronic engineering from Shahid Beheshti University, and he is currently a master's degree student in digital electronics at Iran University of Science and Technology. Parsa's research interest is in the field of computer vision and robustifying deep neural networks against adversarial attacks. He is currently researching in the field of robust object detectors, especially in self-driving cars application.



Amin Banitalebi-Dehkordi received his B.A.Sc. and M.A.Sc. degrees in electrical and computer engineering from the University of Tehran, Iran, in 2008 and 2011, and his Ph.D. in the Digital Media Lab at the University of British Columbia (UBC), Canada, in 2014. His academic career has resulted in many publications in the fields of computer vision and pattern recognition, visual attention modeling, video quality assessment, and high dynamic range video.

His industrial experience expands to areas in machine learning, deep learning, computer vision, NLP, and signal/image/video processing. Amin is currently a principal researcher in machine learning and technical lead at Vancouver Research Centre, Huawei Technologies Canada Co., Ltd.