Engineering 6 (2020) 346–360

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng

Research
Artificial Intelligence—Feature Article

Adversarial Attacks and Defenses in Deep Learning

Kui Ren [a,b,*], Tianhang Zheng [c], Zhan Qin [a,b], Xue Liu [d]

[a] Institute of Cyberspace Research, Zhejiang University, Hangzhou 310027, China
[b] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China
[c] Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 2E8, Canada
[d] School of Computer Science, McGill University, Montreal, QC H3A 0E9, Canada

ARTICLE INFO

Article history:
Received 3 May 2019
Revised 6 September 2019
Accepted 26 December 2019
Available online 3 January 2020

Keywords:
Machine learning
Deep neural network
Adversarial example
Adversarial attack
Adversarial defense

ABSTRACT

With the rapid developments of artificial intelligence (AI) and deep learning (DL) techniques, it is critical to ensure the security and robustness of the deployed algorithms. Recently, the security vulnerability of DL algorithms to adversarial samples has been widely recognized. The fabricated samples can lead to various misbehaviors of the DL models while being perceived as benign by humans. Successful implementations of adversarial attacks in real physical-world scenarios further demonstrate their practicality. Hence, adversarial attack and defense techniques have attracted increasing attention from both machine learning and security communities and have become a hot research topic in recent years. In this paper, we first introduce the theoretical foundations, algorithms, and applications of adversarial attack techniques. We then describe a few research efforts on the defense techniques, which cover the broad frontier in the field. Several open problems and challenges are subsequently discussed, which we hope will provoke further research efforts in this critical area.

1. Introduction

A trillion-fold increase in computation power has popularized the usage of deep learning (DL) for handling a variety of machine learning (ML) tasks, such as image classification [1], natural language processing [2], and game theory [3]. However, a severe security threat to the existing DL algorithms has been discovered by the research community: Adversaries can easily fool DL models by perturbing benign samples without being discovered by humans [4]. Perturbations that are imperceptible to human vision/hearing are sufficient to prompt the model to make a wrong prediction with high confidence. This phenomenon, named the adversarial sample, is considered to be a significant obstacle to the mass deployment of DL models in production. Substantial research efforts have been made to study this open problem.

According to the threat model, existing adversarial attacks can be categorized into white-box, gray-box, and black-box attacks. The difference between the three models lies in the knowledge of the adversaries. In the threat model of white-box attacks, the adversaries are assumed to have full knowledge of their target model, including model architecture and parameters. Hence, they can directly craft adversarial samples on the target model by any means. In the gray-box threat model, the knowledge of the adversaries is limited to the structure of the target model. In the black-box threat model, the adversaries can only resort to the query access to generate adversarial samples. In the frameworks of these threat models, a number of attack algorithms for adversarial sample generation have been proposed, such as limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [4], the fast gradient sign method (FGSM) [5], the basic iterative method (BIM)/projected gradient descent (PGD) [6], distributionally adversarial attack [7], Carlini and Wagner (C&W) attacks [8], Jacobian-based saliency map attack (JSMA) [9], and DeepFool [10]. These attack algorithms are designed in the white-box threat model. However, they are also effective in many gray-box and black-box settings due to the transferability of the adversarial samples among models [11,12].

Meanwhile, various defensive techniques for adversarial sample detection/classification have been proposed recently, including heuristic and certificated defenses. Heuristic defense refers to a defense mechanism that performs well in defending specific

* Corresponding author.
E-mail address: kuiren@zju.edu.cn (K. Ren).

https://doi.org/10.1016/j.eng.2019.12.012

attacks without theoretical accuracy guarantees. Currently, the most successful heuristic defense is adversarial training, which attempts to improve the DL model's robustness by incorporating adversarial samples into the training stage. In terms of empirical results, PGD adversarial training achieves state-of-the-art accuracy against a wide range of L_∞ attacks on several DL model benchmarks such as the modified National Institute of Standards and Technology (MNIST) database, the Canadian Institute for Advanced Research-10 (CIFAR-10) dataset, and ImageNet [13,14]. Other heuristic defenses mainly rely on input/feature transformations and denoising to alleviate the adversarial effects in the data/feature domains. In contrast, certified defenses can always provide certifications for their lowest accuracy under a well-defined class of adversarial attacks. A recently popular network certification approach is to formulate an adversarial polytope and define its upper bound using convex relaxations. The relaxed upper bound is a certification for trained DL models, which guarantees that no attack with specific limitations can surpass the certificated attack success rate, as approximated by the upper bound. However, the actual performance of these certificated defenses is still much worse than that of the adversarial training.

In this paper, we investigate and summarize the adversarial attacks and defenses that represent the state-of-the-art efforts in this area. After that, we provide comments and discussions on the effectiveness of the presented attack and defense techniques. The remainder of the paper is organized as follows: In Section 2, we first sketch out the background. In Section 3, we detail several classic adversarial attack methods. In Section 4, we present a few applications of adversarial attacks in real-world industrial scenarios. In Section 5, we introduce a few recently proposed defense methods. In Section 6, we provide some observations and insights on several related open problems. In Section 7, we conclude this survey.

2. Preliminaries

2.1. Definitions and notations

In this section, we first clarify the definitions and notations used in this paper. Specifically, a dataset is defined as $\{x_i, y_i\}_{i=1}^N$, where x_i is a data sample with label y_i, and N is the size of the dataset. A neural network is denoted as $f(\cdot)$ with input x and prediction $f(x)$. The corresponding optimization loss (also called adversarial loss) is denoted by $J(\theta, x, y)$, where θ represents the model weights. For a classification task, the cross-entropy between $f(x)$ and the label (one-hot) y is always applied as the optimization loss, which is denoted by $J(f(x); y)$. A data sample x' is considered to be an adversarial sample of x when x' is close to x under a specific distance metric while $f(x') \neq y$. Formally, an adversarial sample of x is defined as follows:

$$x' : D(x, x') < \eta,\ f(x') \neq y \tag{1}$$

where $D(\cdot, \cdot)$ is the distance metric and η is a predefined distance constraint, which is also known as the allowed perturbation. Empirically, a small η is adopted to guarantee the similarity between x and x' such that x' is indistinguishable from x.

2.2. Distance metrics

By definition, an adversarial sample x' should be close to a benign sample x under a specific distance metric. The most commonly used distance metric is the L_p distance metric [8]. The L_p distance between x and x' is denoted as $||x - x'||_p$, where $||\cdot||_p$ is defined as follows:

$$||\boldsymbol{v}||_p = \left(|\boldsymbol{v}_1|^p + |\boldsymbol{v}_2|^p + \cdots + |\boldsymbol{v}_d|^p\right)^{1/p} \tag{2}$$

where p is a real number; d is the dimension of the distance vector $\boldsymbol{v}$.

Specifically, the L_0 distance corresponds to the number of the elements in the benign sample x modified by the adversarial attack. The L_2 distance measures the standard Euclidean distance between x and x'. The most popular distance metric—that is, the L_∞ distance—measures the maximum element-wise difference between benign and adversarial samples. There are also several adversarial attacks for discrete data that apply to other distance metrics, such as the number of dropped points [15] and the semantic similarity [16].

2.3. Threat models

There are three mainstream threat models for adversarial attacks and defenses: the black-box, gray-box, and white-box models. These models are defined according to the knowledge of adversaries. In the black-box model, an adversary does not know the structure of the target network or the parameters, but can interact with the DL algorithm to query the predictions for specific inputs. The adversaries always craft adversarial samples on a surrogate classifier trained by the acquired data-and-prediction pairs and other benign/adversarial samples. Owing to the transferability of adversarial samples, black-box attacks can always compromise a naturally trained non-defensive model. In the gray-box model, an adversary is assumed to know the architecture of the target model, but to have no access to the weights in the network. The adversary can also interact with the DL algorithm. In this threat model, the adversary is expected to craft adversarial samples on a surrogate classifier of the same architecture. Due to the additional structure information, a gray-box adversary always shows better attack performance compared with a black-box adversary. The strongest adversary—that is, the white-box adversary—has full access to the target model including all the parameters, which means that the adversary can adapt the attacks and directly craft adversarial samples on the target model. Currently, many defense methods that have been demonstrated to be effective against black-box/gray-box attacks are vulnerable to an adaptive white-box attack. For example, seven out of nine heuristic defenses in the 2018 International Conference on Learning Representations (ICLR2018) were compromised by the adaptive white-box attacks proposed in Ref. [17].

3. Adversarial attacks

In this section, we introduce a few representative adversarial attack algorithms and methods. These methods target to attack image classification DL models, but can also be applied to other DL models. We detail the specific adversarial attacks on the other DL models in Section 4.

3.1. L-BFGS algorithm

The vulnerability of deep neural networks (DNNs) to adversarial samples is first reported in Ref. [4]; that is, hardly perceptible adversarial perturbations are introduced to an image to mislead the DNN classification result. A method called L-BFGS is proposed to find the adversarial perturbations with the minimum L_p norm, which is formulated as follows:

$$\min_x ||x - x'||_p \text{ subject to } f(x') \neq y' \tag{3}$$

where $||x - x'||_p$ is the L_p norm of the adversarial perturbations and y' is the adversarial target label ($y' \neq y$). However, this optimization problem is intractable. The authors propose minimizing a hybrid

loss, that is, $c||x-x'||_p + J(\theta, x', y')$, where c is a hyper parameter, as an approximation of the solution to the optimization problem, where an optimal value of c could be found by line search/grid search.

3.2. Fast gradient sign method

Goodfellow et al. [5] first propose an efficient untargeted attack, called the FGSM, to generate adversarial samples in the L_∞ neighbor of the benign samples, as shown in Fig. 1. FGSM is a typical one-step attack algorithm, which performs the one-step update along the direction (i.e., the sign) of the gradient of the adversarial loss $J(\theta, x, y)$, to increase the loss in the steepest direction. Formally, the FGSM-generated adversarial sample is formulated as follows:

$$x' = x + \epsilon \cdot \text{sign}[\nabla_x J(\theta, x, y)] \tag{4}$$

where ϵ is the magnitude of the perturbation. FGSM can be easily extended to a targeted attack algorithm (targeted FGSM) by descending the gradient of $J(\theta, x, y')$, in which y' is the target label. This update procedure can decrease the cross-entropy between the predicted probability vector and the target probability vector if cross-entropy is applied as the adversarial loss. The update rule for targeted FGSM can be formulated as follows:

$$x' = x - \epsilon \cdot \text{sign}[\nabla_x J(\theta, x, y')] \tag{5}$$

Moreover, it has been discovered that random perturbing before executing FGSM on benign samples can enhance the performance and the diversity of the FGSM adversarial samples.

3.3. BIM and PGD

Kurakin et al. [6] present BIM to improve the performance of FGSM by running a finer iterative optimizer for multiple iterations. The BIM performs FGSM with a smaller step size and clips the updated adversarial sample into a valid range for T iterations; that is, in the tth iteration, the update rule is the following:

$$x'_{t+1} = \text{Clip}\{x'_t + \alpha \cdot \text{sign}[\nabla_x J(\theta, x'_t, y)]\} \tag{6}$$

where $\alpha T = \epsilon$ and α is the magnitude of the perturbation in each iteration. The PGD can be considered as a generalized version of BIM without the constraint $\alpha T = \epsilon$. In order to constrain the adversarial perturbations, the PGD projects the adversarial samples learned from each ite ration into the ϵ–L_∞ neighbor of the benign samples. Hence, the adversarial perturbation size is smaller than ϵ. Formally, the update procedure follows

$$x'_{t+1} = \text{Proj}\{x'_t + \alpha \cdot \text{sign}[\nabla_x J(\theta, x'_t, y)]\} \tag{7}$$

where Proj projects the updated adversarial sample into the ϵ–L_∞ neighbor and a valid range.

3.4. Momentum iterative attack

Inspired by the momentum optimizer, Dong et al. [18] propose the integration of the momentum memory into the iterative process of BIM and derived a new iterative algorithm called momentum iterative FGSM (MI-FGSM). Specifically, MI-FGSM updates the adversarial sample iteratively following

$$x'_{t+1} = \text{Clip}\{x'_t + \alpha \cdot \text{sign}(g_{t+1})\} \tag{8}$$

where the gradient g is updated by $g_{t+1} = \xi \cdot g_t + \nabla_x J(\theta, x'_t, y) / ||\nabla_x J(\theta, x'_t, y)||_1$, ξ is a decay factor.

The authors further proposed a scheme that aims to build an ensemble of models to attack a model in the black-box/gray-box settings. The basic idea is to consider the gradients of multiple models with respect to the input and identify a gradient direction that is more likely to transfer to other models. The combination of MI-FGSM and the ensemble attack scheme won the first places in the non-targeted adversarial attack and targeted adversarial attack competitions (black-box setting) at the 2017 Neural Information Processing Systems (NIPS) conference.

3.5. Distributionally adversarial attack

Zheng et al. [7] propose a new adversarial attack that performs on the space of probability measures, called the distributionally adversarial attack (DAA). Unlike PGD, where adversarial samples are generated independently for each benign sample, DAA performs optimization over the potential adversarial distributions. Moreover, the proposed objective first includes the Kraft–McMillan (KL) divergence between the adversarial and benign data distribution in the calculation of the adversarial loss to increase the adversarial generalization risk during the optimization. This distribution optimization problem is formulated as follows:

$$\max_{\mu} \int_{\mu} J(\theta, x', y) \mathrm{d}\mu + \text{KL}[\mu(x') || \pi(x)] \tag{9}$$

where μ denotes the adversarial data distribution and $\pi(x)$ denotes the benign data distribution.

Since direct optimization over the distribution is intractable, the authors exploit two particle-optimization methods for approximation. Compared with PGD, DAA explores new adversarial patterns, as shown in Fig. 2 [7]. It ranks second on the Massachusetts Institute of Technology (MIT) MadryLab's white-box leaderboards [13], and is considered to be one of the most effective L_∞ attacks on multiple defensive models.

3.6. Carlini and Wagner attack

Carlini and Wagner [8] propose a set of optimization-based adversarial attacks (C&W attacks) that can generate L_0, L_2, and L_∞ norm measured adversarial samples, namely CW_0, CW_2, and

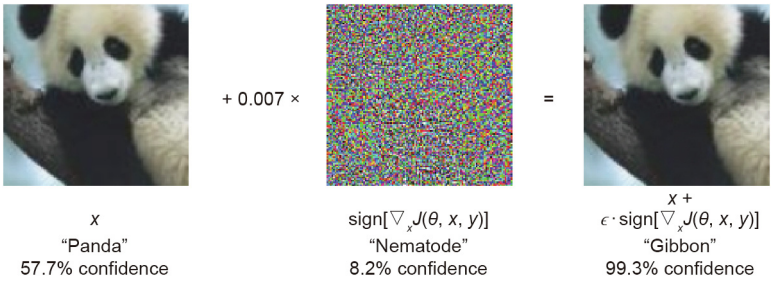


Fig. 1. A demonstration of an adversarial sample generated by applying FGSM to GoogleNet [5]. The imperceptible perturbation crafted by FGSM fools GoogleNet into recognizing the image as a gibbon.

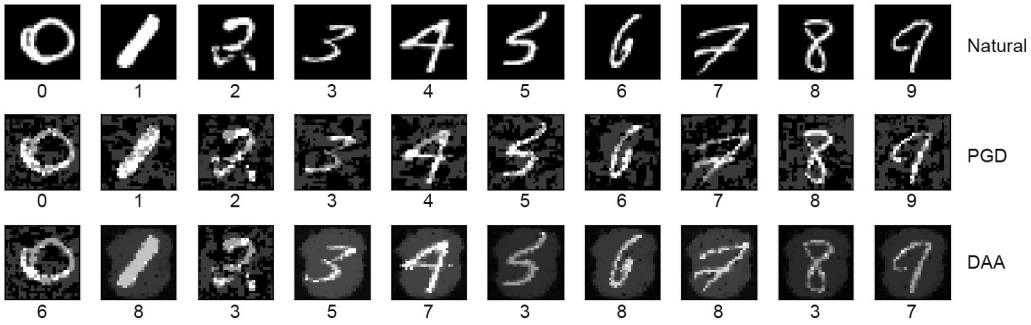


Fig. 2. Comparison between PGD and DAA. DAA tends to generate more structured perturbations [7].

CW_∞. Similar to L-BFGS, it formulates the optimization objective as follows:

$$\min_{\delta} D(x, x+\delta) + c \cdot f(x+\delta) \text{ subject to } x+\delta \in [0,1] \tag{10}$$

where δ denotes the adversarial perturbation, $D(\cdot,\cdot)$ denotes the L_0, L_2, or L_∞ distance metric, and $f(x+\delta)$ denotes a customized adversarial loss that satisfies $f(x+\delta) \leq 0$ if and only if the DNN's prediction is the attack target. To ensure $(x+\delta)$ yields a valid image (i.e., $x+\delta \in [0,1]$), it introduces a new variable κ to substitute δ as follows:

$$\delta = \frac{1}{2}[\tanh(\kappa)+1] - x \tag{11}$$

such that $x+\delta = \frac{1}{2}[\tanh(\kappa)+1]$, which always resides in the range of $[0,1]$ in the optimization process.

C&W attacks achieve 100% attack success rate on naturally trained DNNs for MNIST, CIFAR-10, and ImageNet. They also compromise defensive distilled models, on which L-BFGS and DeepFool fail to find the adversarial samples.

3.7. Jacobian-based saliency map approach

Papernot et al. [9] propose an efficient target attack called the JSMA, which can fool DNNs with small L_0 perturbations. The method first computes the Jacobian matrix of the logit outputs $l(x)$ before the softmax layer:

$$\nabla l(x) = \frac{\partial l(x)}{\partial x} = \left[\frac{\partial l_j(x)}{\partial x_\gamma}\right]_{\gamma \in 1,\ldots,M_{\text{in}}, j \in 1,\ldots,M_{\text{out}}} \tag{12}$$

where M_{in} is the number of neurons on the input layer; M_{out} is the number of neurons on the output layer; γ is the index for input x component; j is the index for output l component.

The Jacobian matrix identifies how the elements of input x affect the logit outputs of different classes. According to the Jacobian matrix, an adversarial saliency map $S(x,y')$ is defined to select the features/pixels that should be perturbed in order to obtain the desired changes in logit outputs. Specifically, the proposed algorithm perturbs the element $x[\gamma]$ with the highest value of $S(x,y')[\gamma]$ to increase/decrease the logit outputs of the target/other class significantly. Hence, perturbations on a small proportion of elements can already affect the $l(x)$ and fool the neural network.

3.8. DeepFool

Moosavi-Dezfooli et al. [10] propose a new algorithm named DeepFool to find the minimum L_2 adversarial perturbations on both an affine binary classifier and a general binary differentiable classifier. For an affine classifier $f(x) = w^{\mathrm{T}}x + b$, where w is the weight of the addine classifier and b is the bias of the affine classifier, the minimum perturbation to change the class of example x_0 is the distance from x_0 to the decision boundary hyperplane $\mathcal{F} = \{x : w^{\mathrm{T}}x + b = 0\}$, that is, $-\frac{f(x_0)}{||w||_2^2}w$. For a general differentiable classifier, DeepFool assumes that f is linear around x_t' and iteratively calculates the perturbation δ_t:

$$\underset{\delta_t}{\operatorname{argmin}} ||\delta_t||_2 \text{ subject to } f(x_t') + \nabla f(x_t')^{\mathrm{T}}\delta_t = 0 \tag{13}$$

This process runs until $f(x_t') \neq f(x)$, and the minimum perturbation is eventually approximated by the sum of δ_t. This method can also be extended to attack the general multi-class classifiers, where the problem is transformed into calculating the distance from x_0 to the surface of a convex polyhedron formed by the decision boundaries between all the classes, as illustrated in Fig. 3 [10]. Experiments show that the perturbation introduced in DeepFool is smaller than FGSM on several benchmark datasets.

3.9. Elastic-net attack to DNNs

Chen et al. [19] propose an adversarial attack that considers the process of generating adversarial examples as an elastic-net regularized optimization problem—namely, the elastic-net attack to DNNs (EAD). In general, EAD attempts to find adversarial examples that can fool the neural network while minimizing the perturbation in terms of L_1 and L_2 distance metrics. Hence, the optimization problem is formulated as follows:

$$\min_{x'} cJ(\theta, x', y') + \beta||x'-x||_1 + ||x'-x||_2^2 \text{ subject to } x' \in [0,1]^d \tag{14}$$

where c and β are hyper parameters, $J(\theta, x', y')$ is the targeted adversarial loss, and $\beta||x'-x||_1 + ||x'-x||_2^2$ is used to penalize the L_1 and L_2 distances between the adversarial samples x' and the benign samples x. EAD first introduces an L_1 norm constraint into adversarial attacks, leading to a different set of adversarial examples with comparable performance to other state-of-the-art methods.

3.10. Universal adversarial attack

In all the attacks mentioned above, the crafted adversarial perturbations are specific to benign samples. In other words, the adversarial perturbations do not transfer across benign samples. Hence, there is a straightforward question: Is there a universal perturbation that can fool the network on most benign samples? Ref. [20] first tries to discover such a perturbation vector by iteratively updating the perturbation using all the target benign samples. In each iteration, for the benign samples that the current perturbation cannot fool, an optimization problem, which is similar to L-BFGS [4], and which aims to discover the minimum additional perturbation required to compromise the samples, is

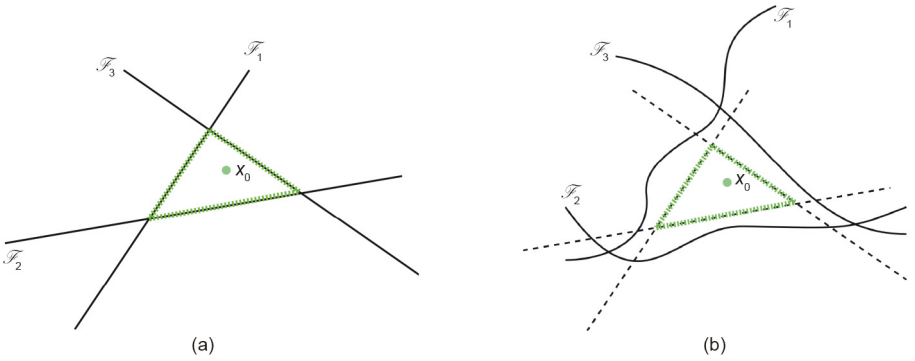


Fig. 3. Convex polyhedron formed by the decision boundaries between all the classes. (a) Linear model; (b) nonlinear model [10].

solved. The additional perturbation is then added to the current perturbation. Eventually, the perturbation enables most of the benign samples to fool the network. Experiments show that this simple iterative algorithm is effective to attack deep nets such as CaffeNet [21], GoogleNet [22], VGG [23], and ResNet [24]. Surprisingly, this cross-sample transferability also maintains across models; for example, the universal perturbations crafted on a VGG can also achieve a fooling ratio above 53% on the other models.

3.11. Adversarial patch

All of the elements of the benign samples (e.g., all the pixels in the benign images) are perturbed in the aforementioned attack algorithms. Recent studies show that perturbations in a restricted region/segment of the benign samples can also fool DL models. These perturbations are called adversarial patches. Sharif et al. [25] proposed the crafting of adversarial perturbations only on an eyeglasses frame attached to the facial images, as shown in Fig. 4. By optimization over a commonly used adversarial loss, such as cross-entropy, the locally crafted perturbation can easily fool VGG-Face convolutional neural network (CNN) [26]. The authors implement this attack in the physical world by three-demensional (3D) printing pairs of eyeglasses with the generated perturbations. This work also presents video demos in which people wearing the adversarial eyeglasses are recognized as the attack targets by a real VGG-Face CNN system. Brown et al. [27] propose a method to generate universal robust adversarial patches. In Ref. [27], the adversarial loss that aims to optimize the patch is defined based on the benign images, patch transformations, and patch locations. Universality is achieved by optimizing the patch over all the benign images. Robustness to noise and transformations is achieved by using the expectation over transformation (EoT) method [28] to compute noise/transformation-insensitive gradients for the optimization. Liu et al. [29] propose adding a Trojan patch on benign samples to generate adversarial samples. The proposed attack first selects a few neurons that can significantly influence the network outputs. Then pixel values in the region of the adversarial patch are initialized to make the selected neurons achieve their maximums. Finally, the model is retrained with benign images and the images with the Trojan patch to adjust the weights related to those selected neurons. Despite performing similarly to the original model on benign images, the retrained model shows malicious behaviors on the images stamped with the adversarial patch.

Fig. 4. Eyeglasses with adversarial perturbations deceive a facial recognition system to recognize the faces in the first row as those in the second row [25].

3.12. GAN-based attacks

Xiao et al. [30] were the first to propose the generation of adversarial samples with generative adversarial networks (GANs). Specifically, a generator is trained to learn the adversarial distribution by maximizing the target adversarial loss $J(\theta, x', y')$ and the GAN loss. A soft hinge loss is incorporated as a penalty to constrain the L_p distance between the generated adversarial samples x' and benign samples x. Notably, under a dynamic distillation setting, where the surrogate classifier (distillation model) is also trained together with the generator by the target classifier's outputs on the generated adversarial samples, the attack reduces the accuracy of the MadryLab's MNIST secret model to 92.74% [13], which is currently the best black-box attack result. Song et al. [31] train an auxiliary classifier GAN (AC-GAN) [32] to model the data distribution for each class. The proposed attack is executed by optimizing over a well-defined objective to find the latent codes of a particular class, which can generate samples that are classified by the target classifier as another class. Since the generated adversarial samples are not close to any existing benign samples, they are referred to as unrestricted adversarial samples. Since this attack does not follow the conventional constraints defined for adversarial samples, it is more effective in attacking models adversarially trained by the attacks that satisfy those constraints.

3.13. Practical attacks

Extending adversarial attack algorithms such as PGD and C&W to the physical world still needs to overcome two major challenges, even though these algorithms are very effective in the digital domain. The first challenge is that the environmental noise and natural transformations will destruct the adversarial perturbations calculated in the digital space. For example, the destruction rate of blur, noise, and joint photographic experts group (JPEG) encoding is reported to be above 80% [6]. The second challenge is specific to the ML tasks using images/videos, in which only the pixels corresponding to certain objects can be perturbed in the physical

world. In other words, adversaries cannot perturb the backgrounds. Athalye et al. [28] propose the EoT method to address the first issue. Instead of using the gradients calculated in the ideal digital domain, EoT adds/applies a set of random noise/natural transformations on the input and then takes an average over all the gradients with respect to those noisy/transformed inputs. This averaged gradient is adopted in gradient-based attack algorithms such as FGSM and PGD to improve the robustness of the generated adversarial samples. In fact, utilization of an adversarial patch can simply solve the second problem—that is, the spatial constraint. Moreover, Eykholt et al. [33] propose a mask/patch transformation to separate the background and the object such that the adversarial perturbations can be restricted in the objects' region. In addition, the authors consider the fabrication errors caused by the difference between the printable and the perturbed RGB values in Ref. [33], as shown in Fig. 5 [33]. The difference is an additional penalty term called the non-printable score, which is included in the optimization loss. Eventually, the work in Ref. [33] successfully generates printable adversarial perturbations on real-world traffic signs and achieved more than 80% attack success rate overall.

3.14. Obfuscated-gradient circumvention attacks

Athalye et al. [17] identify a common problem shared by most heuristic defenses including eight out of nine defenses published in ICLR2018. The problem is that the gradients of those defensive models are either nonexistent or nondeterministic due to add-ons/operations such as quantization and randomization. For these defenses, this work proposes three methods that can circumvent the add-ons/operations to reveal valid gradients for crafting adversarial samples. For defenses relying on non-differentiable add-ons such as quantization, it circumvents the add-ons by using differentiable functions to approximate them. For defenses armed with nondeterministic operations such as random transformations, it simply uses EoT [28] to identify a general gradient direction under the possible transformations and updates the adversarial samples along this general direction. For the defenses that yield exploding or vanishing gradients caused by optimization loops, it proposes to make a change of variable such that the optimization loop will be transformed into a differentiable function. Using the gradients approximated by those three methods, it successfully breaks seven out of nine heuristic defenses in ICLR2018.

4. Adversarial attacks on pervasive applications of industrial interest

In the last section of this article, we will mainly introduce the typical attack algorithms and methods. Most were initially designed for image classification tasks. However, these methods can also be applied to other domains such as image/video segmentation [34,35], 3D recognition [36,37], audio recognition [38], and reinforcement learning [39], which attract growing attention from both academia and industry. Besides, specific data and applications could lead to unique adversarial attacks. Hence, in this section, we sketch out these unique adversarial attacks on the other pervasive applications.

4.1. Adversarial attacks on semantic segmentation models

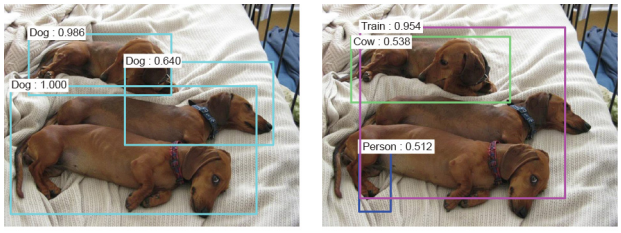
Xie et al. [40] were the first to propose a systematic algorithm—dense adversarial generation (DAG)—to generate adversarial samples for object-detection and segmentation tasks, as shown in Fig. 6. The basic idea of DAG is to consider all the targets in the detection/segmentation task simultaneously and optimize the overall loss. Moreover, in order to tackle the larger number of the proposals in the pixel-level object-detection task (i.e., scaling in $O\left(K^2\right)$, where K is the number of pixels), DAG preserves an increased but reasonable number of proposals by changing the intersection-over-union rate in the optimization process. In Ref. [41], the authors observe that for the segmentation task, the relationship between the widely used adversarial losses and the accuracy is not as well-established as in the classification task. Therefore, they propose a new surrogate loss called Houdini to approximate the real adversarial loss, which is the product of a stochastic margin and a task loss. The stochastic margin characterizes the difference between the predicted probability of the ground truth and that of the predicted target. The task loss is independent of the model, which corresponds to the maximization objective. Also, it further derives an approximation for the gradient of the new surrogate loss with respect to the input to enable the gradient-based optimization over the input. Experiments show that Houdini achieves state-of-the-art attack performance on semantic segmentation and makes the adversarial perturbations more imperceptible to human vision.

4.2. Adversarial attacks on 3D recognition

Point-cloud is an important 3D data representation for 3D object recognition. PointNet [37], PointNet++ [42], and dynamic graph CNN (DGCNN) [43] are the three most popular DL models for point-cloud-based classification/segmentation. However, these three models were also recently found to be vulnerable to adversarial attacks [15,44,45]. In Ref. [44], the authors first extend the C&W attack to the 3D point-cloud DL models. The point locations correspond to the pixel values, and the C&W loss is optimized by shifting the points (i.e., perturbing the point locations). Similarly, the work proposed in Ref. [45] applies BIM/PGD to point-cloud classification and also achieves high attack success rates. In Ref. [15], the authors propose a new attack by dropping the existing points in the point clouds. They approximate the contribution of each point to the classification result by point-shifting to the centroid of the point-cloud and dropping the points with large positive contributions. With a certain number of points dropped, the

(a) (b)

Fig. 5. (a) shows the original image identified by an Inception v3 model as a microwave, and (b) shows its physical adversarial example, identified as a phone [33].



(a) (b)

Fig. 6. In (a), faster CNN correctly detects three dogs and identifies their regions, while in (b) generated by DAG, the segmentation results are completely wrong [40].

classification accuracy of PointNet, PointNet++, and DGCNN are significantly reduced. Besides, works in Ref. [46] propose to add adversarial perturbations on 3D meshes such that the 2D projections of the 3D meshes can mislead 2D-image classification models. This 3D attack is implemented by the optimization on a hybrid loss with the adversarial loss to attack the target 2D model and a penalty loss to keep the 3D adversarial meshes perceptually realistic.

4.3. Adversarial attacks on audio and text recognition

Carlini and Wagner [47] successfully constructed high-quality audio adversarial samples through the optimization over the C&W loss. For an audio signal, up to 50 words in the text translation can be modified by only adversarial perturbing 1% of the audio signal on DeepSpeech [48]. They also found that the constructed adversarial audio signals are robust to pointwise noise and MP3 compression. However, due to the nonlinear effects of microphones and recorders, the perturbed audio signals do not remain adversarial after being played over the air. The authors in Ref. [49] propose simulating the nonlinear effects and the noise while taking them into account in the attack process. Specifically, the authors model the received signal as a function of the transmitted signal, which consists of the transformations for modeling the effects of band-pass filter, impulse response, and white Gaussian noise. The adversarial loss is defined in the received signals instead of the transmitted signals. The proposed attack successfully generates adversarial audio samples in the physical world, which can attack the audio-recognition models even after being played in the air. For text recognition, Liang et al. [50] propose three word-level perturbation strategies on text data, including insertion, modification, and removal. The attack first identifies the important text items for classification, and then exploits one of the perturbation approaches on those text items. Experiments show that this attack can successfully fool some state-of-the-art DNN-based text classifiers. Moreover, TextBugger adopts five types of perturbation operations on text data, including insertion, deletion, swap, character substitution, and word substitution, as shown in Fig. 7 [16]. In the white-box setting, those five operations are also conducted on the important words identified by the Jacobian matrix [9]. However, in the black-box threat model, the Jacobian matrix is unavailable on sentences and documents. The adversary is assumed to have access to the confidence values of the prediction. In this context, the importance of each sentence is defined as its confidence value regarding the predicted class. The importance of each word in the most salient sentence is defined by the difference between the confidence values of the sentence with and without the word.

4.4. Adversarial attacks on deep reinforcement learning

Huang et al. [51] show that existing attack methods can also be used to degrade the performance of the trained policy in deep reinforcement learning by adding adversarial perturbations on the raw inputs of the policy. In particular, the authors construct a surrogate loss $J(\theta, x, y)$ with the parameters θ, the input of the policy x, and a weighted score over all possible actions y. FGSM [5] is used to attack feed-forward policies trained with three algorithms, respectively, including Deep Q-networks [52], asynchronous advantage actor-critic [53], and trust region policy optimization [54]. In most cases, the proposed attack can reduce the accuracy of the agent by 50% under the white-box setting. In the black-box setting, this attack is also effective. The adversarial effects can transfer across those three algorithms, although the attack performance may degrade. Ref. [55] proposes perturbing the input states s_t in the Q-function $Q(s_{t+1}, a, \theta_t)$, such that the learning process will produce an adversarial action a'. FGSM and JSMA are nominated as the adversarial-perturbation-crafting method. Lin et al. [56] propose two attack tactics for deep reinforcement learning, namely the strategically timed attack and the enchanting attack. In the strategically timed attack, the reward is minimized by only perturbing the image inputs for a few specific time-steps. This attack is simply conducted by optimizing the perturbations over the reward. The enchanting attack adversarially perturbs the image frames to lure the agent to the target state. This attack requires a generative model to predict the future states and actions in order to formulate a misleading sequence of actions as guidance for generating perturbations on the image frames.

5. Adversarial defenses

In this section, we summarize the representative defenses developed in recent years, mainly including adversarial training, randomization-based schemes, denoising methods, provable defenses, and some other new defenses. We also present a brief discussion on their effectiveness against different attacks under different settings.

5.1. Adversarial training

Adversarial training is an intuitive defense method against adversarial samples, which attempts to improve the robustness of a neural network by training it with adversarial samples. Formally, it is a min–max game that can be formulated as follows:

$$\min_{\theta} \max_{D(x,x')<\eta} J(\theta, x', y) \tag{15}$$

where $J(\theta, x', y)$ is the adversarial loss, with network weights θ, adversarial input x', and ground-truth label y. $D(x, x')$ represents a certain distance metric between x and x'. The inner maximization problem is to find the most effective adversarial samples, which is solved by a well-designed adversarial attack, such as FGSM [5] and PGD [6]. The outer minimization is the standard training procedure to minimize the loss. The resulting network is supposed to be resistant against the adversarial attack used for the adversarial sample generation in the training stage. Recent studies in Refs. [13,14,57,58] show that adversarial training is one of the most effective defenses against adversarial attacks. In particular, it achieves state-of-the-art accuracy on several benchmarks.

Task: sentiment analysis. **Classifier:** CNN. **Original label:** 99.8% negative. **Adversarial label:** 81.0% positive.

Text: I love these ~~awful~~ **awf ul** 80's summer camp movies. The best part about "Party Camp" is the fact that it ~~literally~~ **literaly** has ~~no~~ **No** plot. The ~~cliches~~ **clichs** here are limitless: the nerds vs. the jocks, the secret camera in the girls locker room, the hikers happening upon a nudist colony, the contest at the conclusion, the secretly horny camp administrators, and the ~~embarrassingly~~ **embarrasing1y** ~~foolish~~ **fo0lish** sexual innuendo littered throughout. This movie will make you laugh, but never intentionally. I repeat, never.

Fig. 7. Adversarial text generated by TextBugger [16]: A negative comment is misclassified as a positive comment.

Therefore, in this section, we elaborate on the best-performing adversarial training techniques in the past few years.

5.1.1. FGSM adversarial training

Goodfellow et al. [5] first propose enhancing the robustness of a neural network by training it with both benign and FGSM-generated adversarial samples. Formally, the proposed adversarial objective can be formulated as follows:

$$\tilde{J}(\theta,x,y)=cJ(\theta,x,y)+(1-c)J(\theta,x+\epsilon\cdot\text{sign}[\nabla_x J(\theta,x,y)],y) \quad (16)$$

where $x+\epsilon\cdot\text{sign}[\nabla_x J(\theta,x,y)]$ is the FGSM-generated adversarial sample for the benign sample x, and c is used to balance the accuracy on benign and adversarial samples as a hyper parameter. Experiments in Ref. [5] show that the network becomes somewhat robust to FGSM-generated adversarial samples. Specifically, with adversarial training, the error rate on adversarial samples dramatically fell from 89.4% to 17.9%. However, the trained model is still vulnerable to iterative/optimization-based adversarial attacks despite its effectiveness when defending FGSM-generated adversarial samples. Therefore, a number of studies further dig into adversarial training with stronger adversarial attacks, such as BIM/PGD attacks.

5.1.2. PGD adversarial training

Extensive evaluations demonstrate that a PGD attack is probably a universal first-order L_∞ attack [13]. If so, model robustness against PGD implies resistance against a wide range of first-order L_∞ attacks. Based on this conjecture, Madry et al. [13] propose using PGD to train a robust network adversarially. Surprisingly, PGD adversarial training indeed improves the robustness of CNNs and ResNets [24] against several typical first-order L_∞ attacks, such as FGSM, PGD, and CW_∞ attacks under both black-box and white-box settings. Even the currently strongest L_∞ attack, that is, DAA, can only reduce the accuracy of the PGD adversarially trained MNIST model to 88.56% and the accuracy of the CIFAR-10 model to 44.71%. In the recent Competition on Adversarial Attacks and Defenses (CAADs), the first-ranking defense against ImageNet adversarial samples relied on PGD adversarial training [14]. With PGD adversarial training, the baseline ResNet [23] already achieves over 50% accuracy under 20-step PGD, while the denoising architecture proposed in Ref. [14] only increases the accuracy by 3%. All the above studies and results indicate that PGD adversarial training is overall the most effective countermeasure against L_∞ attacks. However, due to the large computational cost required for PGD adversarial sample generation, PGD adversarial training is not an efficient method. For example, PGD adversarial training on a simplified ResNet for CIFAR-10 requires approximately three days on a TITAN V graphics processing unit (GPU), and the first ranking model in CAAD costs 52 h on 128 Nvidia V100 GPUs. Besides, a PGD adversarially trained model is only robust to L_∞ attacks and is vulnerable to other L_p-norm adversaries, such as EAD [19,59] and CW_2 [8].

5.1.3. Ensemble adversarial training

To avoid the large computational cost brought by PGD adversarial training, Ref. [60] proposes to adversarially train a robust ImageNet model by FGSM with random starts. However, the adversarially trained model is even vulnerable to black-box attacks. To tackle this problem, the authors propose a training methodology that incorporates adversarial samples transferred from multiple pre-trained models; namely, ensemble adversarial training (EAT) [61]. Intuitively, EAT increases the diversity of adversarial samples used for adversarial training, and thus enhances network robustness against adversarial samples transferred from other models. Experiments show that EAT models exhibit robustness against adversarial samples generated by various single-step and multi-step attacks on the other models. In some circumstances, the performance of EAT against black-box and gray-box attacks is even better than that of PGD adversarial training.

5.1.4. Adversarial logit pairing

Kannan et al. [62] propose a new adversarial training approach called adversarial logit pairing (ALP). Similar to the stability training strategy proposed in Ref. [63], ALP encourages the similarity between pairs of examples in the learned logit space by including the cross-entropy between the logits of benign samples x and the corresponding perturbed samples x' in the training loss. The only difference is that the x' used in Ref. [62] are PGD adversarial samples. The training loss is formally defined as follows:

$$\tilde{J}(\theta,x,x',y)=J(\theta,x,y)+cJ(\theta,x,x') \quad (17)$$

where $J(\theta,x,y)$ is the original loss, $J(\theta,x,x')$ is the cross-entropy between the logits of x and x', and c is a hyper parameter.

Experiments in Ref. [62] show that this pairing loss helps improve the performance of PGD adversarial training on several benchmarks, such as SVHN, CIFAR-10, and tiny ImageNet. Concretely, it is claimed in Ref. [62] that ALP increases the accuracy of the Inception V3 model under the white-box PGD attack from 1.5% to 27.9%. Its performance is almost as good as that of EAT against black-box attacks. However, the work in Ref. [64] evaluates the robustness of an ALP-trained ResNet and discovers that the ResNet only achieves 0.6% correct classification rate under the targeted attack considered in Ref. [62]. The authors also point out that ALP is less amenable to gradient descent, since ALP sometimes induces a "bumpier," that is, depressed loss landscape tightly around the input points. Therefore, ALP might not be as robust as expected in Ref. [62].

5.1.5. Generative adversarial training

All of the above adversarial training strategies employ deterministic attack algorithms to generate training samples. Lee et al. [65] first propose to exploit a nondeterministic generator to generate adversarial samples in the process of adversarial training. Specifically, the proposed work sets up a generator, which takes the gradients of the trained classifier with respect to benign samples as inputs and generates FGSM-like adversarial perturbations. By training the classifier on both benign and generated samples, it also obtains a more robust classifier to FGSM compared with the FGSM adversarially trained model. Liu and Hsieh [66] first propose the use of an AC-GAN architecture [32] for data augmentation to improve the generality of PGD adversarial training. The AC-GAN learns to generate fake samples similar to the PGD adversarial samples through feeding the PGD adversarial samples into the discriminator as real samples. The PGD-like fake samples are exploited to train the auxiliary classifier along with the pre-trained discriminator. According to Ref. [66], such a combination of a generator, discriminator, auxiliary classifier, and PGD attack in a single network not only results in a more robust classifier, but also leads to a better generator.

5.2. Randomization

Many recent defenses resort to randomization schemes for mitigating the effects of adversarial perturbations in the input/feature domain. The intuition behind this type of defense is that DNNs are always robust to random perturbations. A randomization-based defense attempts to randomize the adversarial effects into random effects, which are not a concern for most DNNs. Randomization-based defenses have achieved comparable

performance under black-box and gray-box settings, but in the white-box setting, the EoT method [28] can compromise most of them by considering the randomization process in the attack process. In this section, we present details of several typical randomization-based defenses and introduce their performance against various attacks in different settings.

5.2.1. Random input transformation

Xie et al. [67] utilize two random transformations—random resizing and padding—to mitigate the adversarial effects at the inference time. Random resizing refers to resizing the input images to a random size before feeding them into DNNs. Random padding refers to padding zeros around the input images in a random manner. The pipeline of this quick and sharp mechanism is shown in Fig. 8 [67]. The mechanism achieves a remarkable performance under black-box adversarial settings, and ranked second place in the NIPS 2017 adversarial examples defense challenge. However, under the white-box setting, this mechanism was compromised by the EoT method [28]. Specifically, by approximating the gradient using an ensemble of 30 randomly resized and padded images, EoT can reduce the accuracy to 0 with 8/255 L_∞ perturbations. In addition, Guo et al. [68] apply image transformations with randomness such as bit-depth reduction, JPEG compression, total variance minimization, and image quilting before feeding the image to a CNN. This defense method resists 60% of strong gray-box and 90% of strong black-box adversarial samples generated by a variety of major attack methods. However, it is also compromised by the EoT method [28].

5.2.2. Random noising

Liu et al. [69] propose to defend adversarial perturbations by a random noising mechanism called random self-ensemble (RSE). RSE adds a noise layer before each convolution layer in both training and testing phases, and ensembles the prediction results over the random noises to stabilize the DNN's outputs, as shown in Fig. 9 [69]. Lecuyer et al. [70] view the random noising defensive mechanism from the angle of differential privacy (DP) [71] and propose a DP-based defense called PixelDP. PixelDP incorporates a DP noising layer inside DNN to enforce DP bounds on the variation of the distribution over its predictions of the inputs with small, norm-based perturbations. PixelDP can be used to defend L_1/L_2 attacks using Laplacian/Gaussian DP mechanisms. Inspired by PixelDP, the authors in Ref. [72] further propose to directly add random noise to pixels of adversarial examples before classification, in order to eliminate the effects of adversarial perturbations. Following the theory of Rényi divergence, it proves that this simple method can upper-bound the size of the adversarial perturbation it is robust to, which depends on the first- and second-largest probabilities of the output probability distribution (vector).

5.2.3. Random feature pruning

Dhillon et al. [73] present a method called stochastic activation pruning (SAP) to protect pre-trained networks against adversarial samples by stochastically pruning a subset of the activations in each layer and preferentially retaining activations with larger magnitudes. After activation pruning, SAP scales up the surviving activations to normalize the inputs of each layer. However, on CIFAR-10, EoT [28] can also reduce the accuracy of SAP to 0 with 8/255 L_∞ adversarial perturbations. Luo et al. [74] introduce a new CNN structure by randomly masking the feature maps output from the convolutional layers. By randomly masking the output features, each filter only extracts the features from partial positions. The authors claim that this assists the filters in learning features distributing consistently with respect to the mask pattern; hence, the CNN can capture more information on the spatial structures of local features.

5.3. Denoising

Denoising is a very straightforward method in terms of mitigating adversarial perturbations/effects. Previous works point out two directions to design such a defense: input denoising and feature denoising. The first direction attempts to partially or fully remove the adversarial perturbations from the inputs, and the second direction attempts to alleviate the effects of adversarial perturbations on high-level features learned by DNNs. In this section, we elaborate on several well-known defenses in both directions.

5.3.1. Conventional input rectification

In order to mitigate the adversarial effects, Xu et al. [75] first utilize two squeezing (denoising) methods—bit-reduction and image-blurring—to reduce the degrees of freedom and remove the adversarial perturbations, as shown in Fig. 10. Adversarial sample detection is realized by comparing the model predictions on the original and squeezed images. If the original and squeezed inputs produce substantially different outputs from the model, the original input is likely to be an adversarial sample. Xu et al. [76] further show that the feature-squeezing methods proposed in Ref. [75] can mitigate the C&W attack. However, He et al. [77] demonstrate that feature squeezing is still vulnerable to an adaptive knowledgeable adversary. It adopts the CW_2 loss as the adversarial loss. After each step of the optimization procedure, an intermediate image is available from the optimizer. The reduced-color-depth version of this intermediate image is checked by the detection system proposed by Xu et al. [75]. Such an optimization

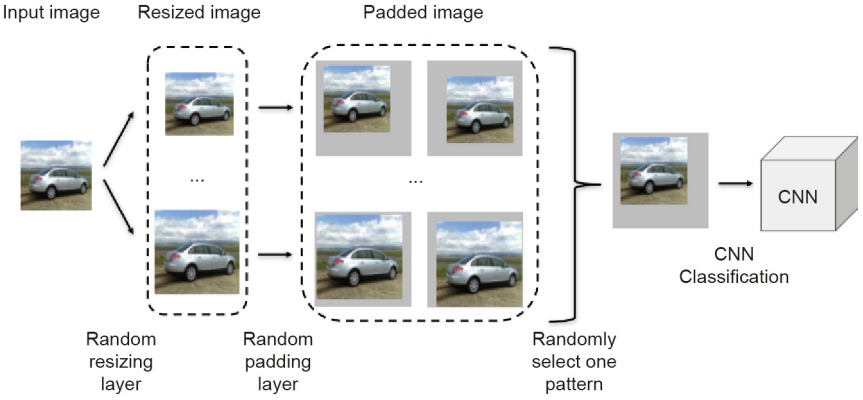


Fig. 8. The pipeline of the randomization-based defense mechanism proposed by Xie et al. [67]: The input image is first randomly resized and then randomly padded.

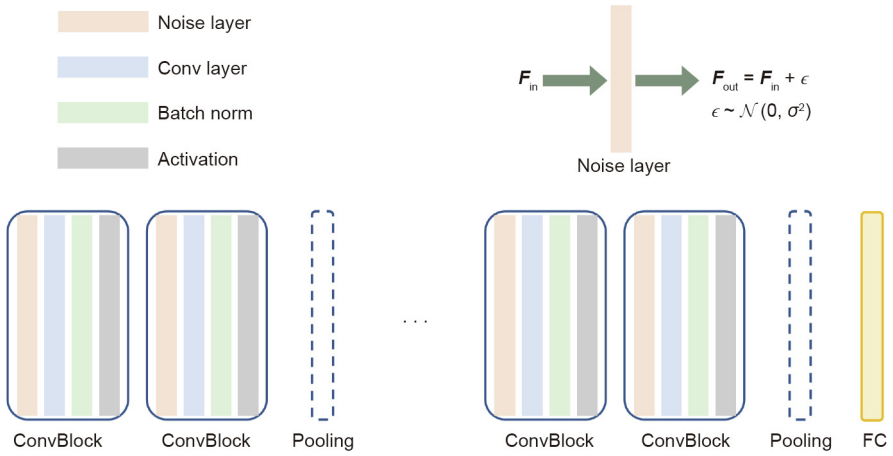


Fig. 9. The architecture of RSE [69]. FC: fully connected layer; $\boldsymbol{F}_{in}$: the input vector of the noise layer; $\boldsymbol{F}_{out}$: the output vector of the noise layer; ϵ: the perturbation which follow the Gaussian distribution $\mathcal{N}(0, \sigma^2)$; conv: convolution.

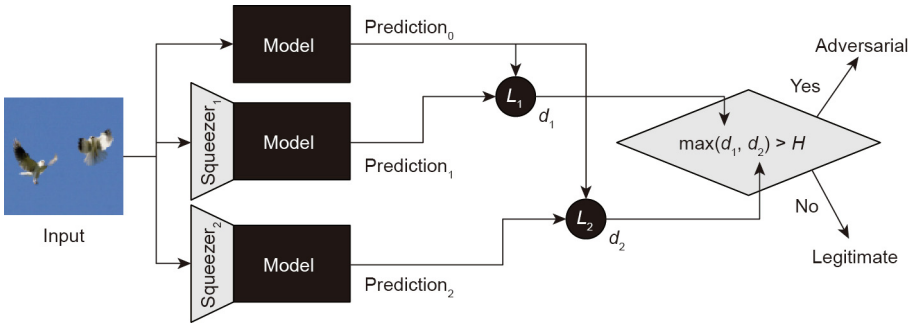


Fig. 10. The feature-squeezing framework proposed by Xu et al. [75]. d_1 and d_2: the difference between the model's prediction on a squeezed input and its prediction on the original input; H: the threshold which is used to detect adversarial examples.

procedure runs multiple times, and all the intermediate adversarial samples that can bypass Xu's system are aggregated. This whole adaptive attack can break the input squeezing system with perturbations much smaller than those claimed in Ref. [75]. Moreover, Sharma and Chen [78] also show that EAD and CW_2 can bypass the input squeezing system with increasing adversary strength.

5.3.2. GAN-based input cleansing

A GAN is a powerful tool to learn a generative model for data distributions. Thus, plenty of works intend to exploit GANs to learn benign data distribution in order to generate a benign projection for an adversarial input. Defense-GAN and adversarial perturbation elimination GAN (APE-GAN) are two typical algorithms among all these works. Defense-GAN [79] trains a generator to model the distribution of benign images, as shown in Fig. 11 [79]. In the testing stage, Defense-GAN cleanses an adversarial input by searching for an image close to the adversarial input in its learned distribution, and feed this benign image into the classifier. This strategy can be used to defend against various adversarial attacks. Currently, the most effective attack scheme against Defense-GAN is based on backward pass differentiable approximation [17], which can reduce its accuracy to 55% with 0.005 L_2 adversarial perturbations. APE-GAN [80] directly learns a generator to cleanse an adversarial sample by using it as input, and outputs a benign counterpart. Although APE-GAN achieves a good performance in the testbed of Ref. [80], the adaptive white-box CW_2 attack proposed in Ref. [81] can easily defeat APE-GAN.

5.3.3. Auto encoder-based input denoising

In Ref. [82], the authors introduce a defensive system called MagNet, which includes a detector and a reformer. In MagNet, an auto-encoder is used to learn the manifold of benign samples. The detector distinguishes the benign and adversarial samples based on the relationships between those samples and the learned

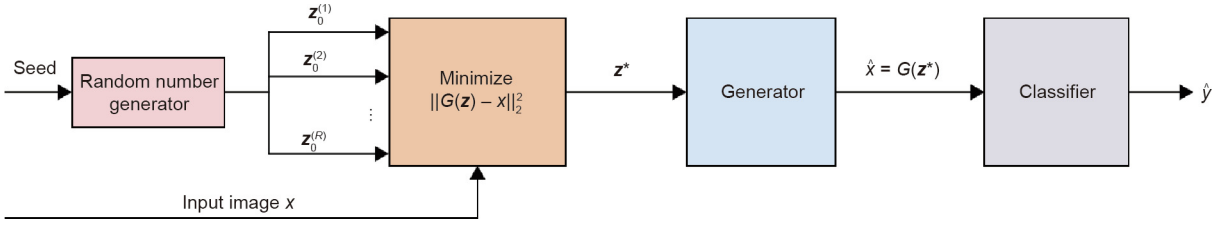


Fig. 11. The pipeline of Defense-GAN [79]. G: the generative model which can generate a high-dimensional input sample from a low dimensional vector $\boldsymbol{z}$; R: the number of random vectors generated by the random number generator.

manifold. The reformer is designed to rectify the adversarial samples into benign samples. The authors show the effectiveness of MagNet against a variety of adversarial attacks under gray-box and black-box settings, such as FGSM, BIM, and C&W. However, Carlini and Wagner [81] demonstrate that MagNet is vulnerable to the transferable adversarial samples generated by the CW_2 attack.

5.3.4. Feature denoising

Liao and Wagner [83] propose a high-level representation guided denoiser (HGD) to polish the features affected by the adversarial perturbations. Instead of denoising on the pixel level, HGD trains a denoising u-net [34] using a feature-level loss function to minimize the feature-level difference between benign and adversarial samples. In the NIPS 2017 competition, HGD won first place in the defense track (black-box setting). Despite the effectiveness under black-box settings, HGD is compromised by a PGD adversary under a white-box setting in Ref. [84]. Experiments in Ref. [84] indicate that the PGD attack with 4/255 L_∞ perturbations can already reduce the accuracy of HGD to 0. Xie et al. [14] design a block for learning several denoising operations to rectify the features learned by intermediate layers in DNNs. The modified PGD adversarially trained network ranked first place in the adversarial defense track of CAAD 2018. Despite the remarkable success of Ref. [14], the contribution of the feature-denoising block to network robustness is not compared with PGD adversarial training, since the PGD adversarially trained baseline can also achieve nearly 50% accuracy under white-box PGD attacks, and the denoising block only improves the accuracy of this baseline by 3%.

5.4. Provable defenses

All of the above defenses are heuristic defenses, which means that the effectiveness of these defenses is only experimentally validated, rather than being theoretically proved. Without a theoretical error-rate guarantee, those heuristic defenses might be broken by a new attack in the future. Therefore, many researchers have put efforts into developing provable defensive methods, which can always maintain a certain accuracy under a well-defined class of attacks. In this section, we introduce several typical certificated defenses.

5.4.1. Semidefinite programming-based certificated defense

Raghunathan and Kolter [85] first propose a certifiable defense method against adversarial examples on two-layer networks. The authors derive a semidefinite relaxation to upper-bound the adversarial loss and incorporate the relaxation into the training loss as a regularizer. This training method produces a network with a certificate that no attack with at most 0.1/1.0 L_∞ perturbations can cause more than 35% test error on MNIST. In Ref. [86], Raghunathan et al. further propose a new semidefinite relaxation for certifying arbitrary ReLU networks. The newly proposed relaxation is tighter than the previous one and can produce meaningful robustness guarantees on three different networks.

5.4.2. Dual approach-based provable defense

Along with Ref. [85], Wong and Kolter [87] formulate a dual problem to upper-bound the adversarial polytope. They show that the dual problem can be solved by optimization over another deep neural network. Unlike Ref. [85], which only applies to two-layer fully connected networks, this approach can be applied to deep networks with arbitrary linear operator layers, such as convolution layers. The authors further extend the technique in Ref. [87] to much more general networks with skip connections and arbitrary nonlinear activations in Ref. [88]. They also present a nonlinear random projection technique to estimate the bound in a manner that only scales linearly in the size of the hidden units, making the approach applicable to larger networks. On both MNIST and CIFAR datasets, the proposed work trains classifiers using the proposed techniques that substantially improve the previous provable robust adversarial error guarantees: from 5.8% to 3.1% on MNIST with L_∞ perturbations of $\epsilon = 0.1$, and from 80% to 36.4% on CIFAR with L_∞ perturbations of $\epsilon = 2/255$.

5.4.3. Distributional robustness certification

From the perspective of distribution optimization, Sinha et al. [89] formulate an optimization problem over adversarial distributions as follows:

$$\min_{\theta} \sup_{\varphi \in \phi} E_{\varphi}[J(\theta, x, y)] \tag{18}$$

where ϕ is a candidate set for all the distributions around the benign data, which can be constructed by f-divergence balls [90] or Wasserstein balls [91], φ is sampled from the candidate set ϕ.

Optimization over this distributional objective is equivalent to minimizing the empirical risk over all the samples in the neighbor of the benign data—that is, all the candidates for the adversarial samples. Since ϕ affects the computability, and direct optimization over an arbitrary ϕ is intractable, the work in Ref. [80] derives tractable sets ϕ using the Wasserstein distance metric with computationally efficient relaxations that are computable even when $J(\theta, x, y)$ is non-convex. In fact, the work in Ref. [89] also provides an adversarial training procedure with provable guarantees on its computational and statistical performance. In the proposed training procedure, it incorporates a penalty to characterize the adversarial robustness region. Since optimization over this penalty is intractable, the authors propose a Lagrangian relaxation for the penalty and achieve robust optimization over the proposed distributional loss. In addition, the authors derive guarantees for the empirical minimizer of the robust saddle-point problem and give specialized bounds for domain adaptation problems, which also shed light on the distributional robustness certification.

5.5. Weight-sparse DNNs

Guo et al. [92] are the first to demonstrate the intrinsic relationship between weight sparsity and network robustness against FGSM-generated and DeepFool-generated adversarial samples. For linear models, Ref. [92] demonstrates that optimization over adversarial samples could give rise to a sparse solution of the network weights. For nonlinear neural networks, it applies the robustness guarantees from Refs. [93,94] and demonstrates that the network Lipchitz constant is prone to be smaller when the weight matrices are sparser. Since it is observed that minimizing the Lipchitz constant can help improve network robustness [93], the conclusion is that weight sparsity can also lead to a more robust neural network. In Ref. [95], it is also shown that weight sparsity is beneficial to network robustness verification. The authors demonstrate that weight sparsity could turn the computationally intractable verification problems into tractable ones. The authors improve the weight sparsity on neural networks by training them with L_1 regularization, and discover that weight sparsity significantly speeds up the linear programming solvers [96] for network robustness verification.

5.6. KNN-based defenses

Wang et al. [97] first develop a framework for analyzing the adversarial robustness of the k-nearest neighbor (KNN) algorithm. This framework identifies two distinct regimes of k with different robustness properties. Specifically, KNN with constant k has no robustness under the large sample limit condition in the regions

where the conditional probability $P(y=1|x)$ lies in $(0,1)$. For $k=\Omega[\sqrt{dn\log(n)}]$, where d is the data dimension and n is the sample size, the robustness region of KNN-based classification approaches that of the Bayes optimal classifier under the large sample limit condition. Since $k=\Omega[\sqrt{dn\log(n)}]$ is too large for real datasets with a high data dimension and numerous samples, the authors propose an efficient 1-nearest neighbor algorithm. Based on the observation that the 1-nearest neighbor is robust when oppositely labeled points are far apart, the proposed algorithm removes the nearby oppositely labeled points and keeps the points whose neighbors share the same label. On MNIST, for small adversarial perturbations (low attack radii), this algorithm followed by 1-nearest neighbor-based classification performs slightly worse than the other defenses, such as an adversarially trained classifier, while it outperforms those defenses in the case of large attack radii. Papernot and McDaniel [98] propose a KNN-based defensive mechanism called DkNN by executing the KNN algorithm on the representations of the data learned by each layer of the DNN. The KNN algorithm is mainly used to estimate the abnormality of a prediction on the test input. The prediction is considered abnormal when the intermediate representations learned by the DNN are not close to the representations of those training samples that share the same label with the prediction. Experiments show that DkNN significantly improves the accuracy of DNN under multiple adversarial attacks, especially under the C&W attack.

5.7. Bayesian model-based defenses

Liu et al. [99] combine the Bayesian neural network (BNN) [100] with adversarial training to learn the optimal model-weight distribution under adversarial attacks. Specifically, the authors assume that all the weights in the network are stochastic, and train the network with the techniques commonly used in the BNN theory [100]. Through adversarially training such a stochastic BNN, the BNN with adversarial training shows a significant improvement of adversarial robustness compared with RSE [69] and common adversarial training on CIFAR-10, STL-10, and ImageNet143. Schott et al. [101] suggest modeling the class-conditional distributions for the input data based on the Bayesian model, and classify a new sample as the class under which the corresponding class-conditional model yields the highest likelihood. The authors name the model the "analysis by synthesis" (ABS) model. ABS is considered to be the first robust model for the MNIST dataset against L_0, L_2, and L_∞ attacks. Specifically, it achieves state-of-the-art performance against L_0 and L_2 attacks, but performs slightly worse than PGD adversarially trained model under L_∞ attacks.

5.8. Consistency-based defenses

For ML tasks such as audio recognition and image segmentation, consistency information can be applied to distinguish between benign and adversarial samples. Xiao et al. [102] find that for the semantic segmentation task, adversarially perturbing one pixel also affects the predictions of its surrounding pixels. Therefore, perturbing on a single patch can also break the spatial consistency between its nearby batches. Such consistency information makes the benign and adversarially perturbed images distinguishable. This consistency-based methodology is evaluated against adaptive attacks and demonstrates better performance than other anomaly-detection systems. For the audio-recognition task, Yang et al. [103] explore the temporal consistency of audio signals and discover that adversarial perturbation destroys the temporal consistency. Specifically, for an adversarial audio signal, the translation of a portion of the signal is not consistent with the translation of the whole signal. It shows that the detection based on the consistency test can achieve more than 90% detection rate on adversarial audio signals.

6. Discussions

6.1. White-box and black-box attacks

From the perspective of adversaries, the main difference between white-box and black-box settings is the level of their access to the target model. Under white-box settings, the model structure and the weights are accessible to the adversaries, so they can compute the true model gradients or approximate the gradients by the methods in Ref. [17]. Besides, the adversaries can adapt their attack method with the defense method and parameters. In this context, most of the heuristic defenses introduced before are ineffective against such strong adaptive adversaries. However, under black-box settings, the model structure and the weights are secrets to the adversaries. In this context, to apply the above gradient-based attack algorithms, the adversaries have to infer the model gradients from limited information. Without any model-specific information, unbiased estimation of the model gradient is the expectation of the gradients of the pre-trained models' ensemble with different random seeds. A momentum gradient-based method with this gradient estimation achieved first place in the NIPS 2017 Challenge (under a black-box setting) [18]. Chen et al. [104] investigate another black-box setting, where additional query access is granted to the adversaries. Therefore, the adversaries can infer the gradients from the output of the target model given well-designed inputs. In this setting, the proposed design can apply a zero-order method to give a much better estimation of the model gradients. However, a drawback of this method is its requirement for a large number of queries, which is proportional to the data dimension.

6.2. Differences between the trends of adversarial attacks and defenses

The trend of research on adversarial attacks mainly includes two directions. The first direction is to design more efficient and stronger attacks in order to evaluate various emerging defensive systems. The importance of this direction is intuitive, since we expect to understand all the risks ahead of the potential adversaries. The second direction is to realize the adversarial attacks in the physical world. Previously, the main doubt about this research topic was whether those adversarial attacks were real threats in the physical world. Some researchers suspected that adversarial attacks initially designed in digital spaces would not be effective in the physical world due to the influence of certain environmental factors. Kurakin et al. [6] first address this challenge by using the expectation of the model gradients with respect to the inputs plus the random noise caused by environmental factors. Eykholt et al. [33] further consider the masks and fabrication errors, and implemented adversarial perturbations on traffic signs. Recently, Cao et al. [105] successfully generat adversarial objectives to deceive the LiDAR-based detection system, thus validating the existence of physical adversarial samples again. In terms of defenses, the community is starting to focus on certificated defenses, since most heuristic defenses fail to defend against adaptive white-box attacks, and a certificated defense is supposed to guarantee the defensive performance under certain situations regardless of the attack method used by the adversaries. However, until now, scalability has been a common problem shared by most certificated defenses. For example, interval bound analysis is a recently popular direction to certify DNNs, but it is not scalable to very deep neural networks and large datasets. Clearly, compared with attacks, the development of defenses faces more challenges. This is mainly

because an attack can only target one category of defenses, but defenses are required to be certificated—that is, effective against all possible attack methods under certain situations.

6.3. Recent advances in model robustness analysis

Since it is difficult to theoretically analyze DNNs due to their complicated non-convex properties, the community starts by analyzing the robustness of some simple ML models such as KNN and linear classifiers. Wang et al. [97] point out that the robustness of KNN highly depends on the parameter k, data dimension d, and data size n. k has to be very large to guarantee that KNN is asymptotically robust, like a Bayes optimal classifier. Fawzi et al. [106] introduce a theoretical framework for analyzing the robustness of linear and quadratic classifiers, where model robustness is defined by the average of the norms of the small perturbations that switch the classes of data samples. The upper bounds of the model robustness are proved under an assumption applicable to a large number of classifiers including linear and quadratic classifiers, which shows that the adversarial robustness scales in $O(\sqrt{1/d})$ compared with the robustness to uniform random noise. Recently, the robustness of MLP, CNN, and ResNet have been widely studied by means of interval-bound analysis using abstractions, which attempts to bound the output layer by layer with the perturbation size. We do not detail the analysis in this survey, and refer interested readers to Refs. [107–109]

6.4. Main unresolved challenges

Causality behind adversarial samples: Although numerous adversarial attacks have been proposed, the causation of adversarial samples is still not well-understood. Early studies on this issue owe the pervasiveness of adversarial samples to the model structures and learning methods, which assume that appropriate strategies and network architecture significantly improve adversarial robustness. However, efforts in this direction—especially those yielding obfuscated gradients—actually give a false sense of security [2]. In contrast, very recent works have found that adversarial vulnerability is more likely to be a consequence of high-dimensional data geometry [110–112] and insufficient training data [113]. Specifically, the works proposed in Refs. [110–112] prove that adversarial perturbation scales in $O(\sqrt{1/d})$ on several proof-of-concept datasets, such as $\{0,1\}^n$ and concentric n-dimensional spheres, where d is the data dimension. Schmidt et al. [113] show that adversarially robust generalization requires more data than common ML tasks, and the required data size scales in $O(\sqrt{1/d})$.

Existence of a general robust decision boundary. Since there are numerous adversarial attacks defined under different metrics, a natural question is: Is there a general robust decision boundary that can be learned by a certain kind of DNNs with a particular training strategy? At present, the answer to this question is "no." Although PGD adversarial training demonstrates remarkable resistance against a wide range of L_∞ attacks, Sharma and Chen [59] show that it is still vulnerable to adversarial attacks measured by other L_p norms, such as EAD and CW_2. Recently, Khoury and Hadfield-Menell [111] prove that the optimal L_2 and L_∞ decision boundaries are different for a two-concentric-sphere dataset, and their disparity grows with the codimension of the dataset—that is, the difference between the dimensions of the data manifold and the whole data space.

Effective and efficient defense against white-box attacks. To the best of our knowledge, no defense that can achieve a balance between effectiveness and efficiency has been proposed. In terms of effectiveness, adversarial training demonstrates the best performance but at a substantial computational cost. In terms of efficiency, the configuration of many randomization-based and denoising-based defenses/detection systems only takes a few seconds. However, many recent works [17,84,114,115] show that those schemes are not as effective as they claim to be. Certificated defenses indicate a way to reach theoretically guaranteed security, but both their accuracy and their efficiency are far from meeting the practical requirements.

7. Conclusions

In this paper, we have presented a general overview of the recent representative adversarial attack and defense techniques. We have investigated the ideas and methodologies of the proposed methods and algorithms. We have also discussed the effectiveness of these adversarial defenses based on the most recent advances. New adversarial attacks and defenses developed in the past two years have been elaborated. Some fundamental problems, such as the causation of adversarial samples and the existence of a general robust boundary, have also been investigated. We have observed that there is still no existing defense mechanism that achieves both efficiency and effectiveness against adversarial samples. The most effective defense mechanism, which is adversarial training, is too computationally expensive for practical deployment, while many efficient heuristic defenses have been demonstrated to be vulnerable to adaptive white-box adversaries. We have also discussed several open problems and challenges in this critical area to provide a useful research guideline to boost the development of this critical area.

Acknowledgements

This work has been supported by Ant Financial, Zhejiang University Financial Technology Research Center.

Compliance with ethics guidelines

Kui Ren, Tianhang Zheng, Zhan Qin, and Xue Liu declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 26th Conference on Neural Information Processing Systems; 2012 Dec 3–6; Lake Tahoe, NV, USA; 2012. p. 1097–105.
[2] Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. 2014. arXiv:1406.1078.
[3] Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den Driessche G, et al. Mastering the game of Go with deep neural networks and tree search. Nature 2016;529(7587):484–9.
[4] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, et al. Intriguing properties of neural networks. 2013. arXiv:1312.6199.
[5] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. 2014. arXiv:1412.6572.
[6] Kurakin A, Goodfellow I, Bengio S. Adversarial examples in the physical world. 2016. arXiv:1607.02533.
[7] Zheng T, Chen C, Ren K. Distributionally adversarial attack. 2018. arXiv:1808.05537.
[8] Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: Proceedings of the 2017 IEEE Symposium on Security and Privacy; 2017 May 22–26; San Jose, CA, USA; 2017. p. 39–57.
[9] Papernot N, McDaniel P, Jha S, Fredrikson M, Celik ZB, Swami A. The limitations of deep learning in adversarial settings. In: Proceedings of the 2016 IEEE European Symposium on Security and Privacy; 2016 Mar 21–24; Saarbrucken, Germany; 2016. p. 372–87.
[10] Moosavi-Dezfooli SM, Fawzi A, Frossard P. DeepFool: a simple and accurate method to fool deep neural networks. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA; 2016. p. 2574–82.

[11] Papernot N, McDaniel P, Goodfellow I. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. 2016. arXiv:1605.07277.
[12] Liu Y, Chen X, Liu C, Song D. Delving into transferable adversarial examples and black-box attacks. 2016. arXiv:1611.02770.
[13] Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. 2017. arXiv: 1706.06083.
[14] Xie C, Wu Y, van der Maaten L, Yuille A, He K. Feature denoising for improving adversarial robustness. 2018. arXiv:1812.03411.
[15] Zheng T, Chen C, Yuan J, Li B, Ren K. PointCloud saliency maps. 2018. arXiv:1812.01687.
[16] Li J, Ji S, Du T, Li B, Wang T. TextBugger: generating adversarial text against real-world applications. 2018. arXiv:1812.05271.
[17] Athalye A, Carlini N, Wagner D. Obfuscated gradients give a false sense of security: circumventing defenses to adversarial examples. 2018. arXiv:1802.00420.
[18] Dong Y, Liao F, Pang T, Su H, Zhu J, Hu X, et al. Boosting adversarial attacks with momentum. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA; 2018. p. 9185–193.
[19] Chen PY, Sharma Y, Zhang H, Yi J, Hsieh CJ. EAD: elastic-net attacks to deep neural networks via adversarial examples. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence; 2018 Feb 2–7; New Orleans, LA, USA; 2018.
[20] Moosavi-Dezfooli SM, Fawzi A, Fawzi O, Frossard P. Universal adversarial perturbations. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017 Jul 21–26; Honolulu, HI, USA; 2017. p. 1765–73.
[21] Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, et al. Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia; 2014 Nov 3–7; Orlando, FL, USA; 2014. p. 675–8.
[22] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition; 2015 Jun 7–12; Boston, MA, USA; 2015. p. 1–9.
[23] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014. arXiv:1409.1556.
[24] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA; 2016. p. 770–8.
[25] Sharif M, Bhagavatula S, Bauer L, Reiter MK. Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security; 2016 Oct 24–28; Vienna, Austria; 2016. p. 1528–40.
[26] Parkhi OM, Vedaldi A, Zisserman A. Deep face recognition. In: Proceedings of British Machine Vision Conference; 2017 Sep 7–10; Swansea, UK; 2015.
[27] Brown TB, Mané D, Roy A, Abadi M, Gilmer J. Adversarial patch. 2017. arXiv:1712.09665.
[28] Athalye A, Engstrom L, Ilya A, Kwok K. Synthesizing robust adversarial examples. 2017. arXiv:1707.07397.
[29] Liu Y, Ma S, Aafer Y, Lee WC, Zhai J, Wang W, et al. Trojaning attack on neural networks. In: Proceedings of Network and Distributed Systems Security Symposium; 2018 Feb 18–21; San Diego, CA, USA; 2018.
[30] Xiao C, Li B, Zhu JY, He W, Liu M, Song D. Generating adversarial examples with adversarial networks. 2018. arXiv:1801.02610.
[31] Song Y, Shu R, Kushman N, Ermon S. Constructing unrestricted adversarial examples with generative models. In: Proceedings of the 32nd Conference on Neural Information Processing Systems; 2018 Dec 3–8; Montréal, QC, Canada; 2018. p. 8312–23.
[32] Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier GANs. In: Proceedings of the 34th International Conference on Machine Learning; 2017 Aug 6–11; Sydney, NSW, Austrila; 2017. p. 2642–51.
[33] Eykholt K, Evtimov I, Fernandes E, Li B, Rahmati A, Xiao C, et al. Robust physical-world attacks on deep learning visual classification. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA; 2018. p. 1625–34.
[34] Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention; 2015 Oct 5–9; Munich, Germany; 2015. p. 234–41.
[35] Grundmann M, Kwatra V, Han M, Essa I. Efficient hierarchical graph-based video segmentation. In: Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2010 Jun 13–18; San Francisco, CA, USA; 2010. p. 2141–8.
[36] Su H, Maji S, Kalogerakis E, Learned-Miller E. Multi-view convolutional neural networks for 3D shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision; 2015 Dec 7–13; Santiago, Chile; 2015. p. 945–53.
[37] Qi CR, Su H, Mo K, Guibas LJ. PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017 Jul 21–26; Honolulu, HI, USA; 2017. p. 652–60.
[38] Lee H, Pham P, Largman Y, Ng AY. Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Proceedings of the 23rd Conference on Neural Information Processing Systems; 2009 Dec 7–10; Vancouver, BC, Canada; 2009. p. 1096–104.
[39] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, et al. Human-level control through deep reinforcement learning. Nature 2015;518 (7540):529–33.
[40] Xie C, Wang J, Zhang Z, Zhou Y, Xie L, Yuille A. Adversarial examples for semantic segmentation and object detection. In: Proceedings of the 2017 IEEE International Conference on Computer Vision; 2017 Oct 22–29; Venice, Italy; 2017. p. 1369–78.
[41] Cisse M, Adi Y, Neverova N, Keshet J. Houdini: fooling deep structured prediction models. 2017. arXiv:1707.05373.
[42] Qi CR, Yi L, Su H, Guibas LJ. PointNet+: deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st Conference on Neural Information Processing Systems; 2017 Dec 4–9; Long Beach, CA, USA; 2017. p. 5099–108.
[43] Wang Y, Sun Y, Liu Z, Sarma SE, Bronstein MM, Solomon JM. Dynamic graph CNN for learning on point clouds. 2018. arXiv:1801.07829.
[44] Xiang C, Qi CR, Li B. Generating 3D adversarial point clouds. 2018. arXiv:1809.07016.
[45] Liu D, Yu R, Su H. Extending adversarial attacks and defenses to deep 3D point cloud classifiers. 2019. arXiv:1901.03006.
[46] Xiao C, Yang D, Li B, Deng J, Liu M. MeshAdv: adversarial meshes for visual recognition. 2018. arXiv:1810.05206v2.
[47] Carlini N, Wagner D. Audio adversarial examples: targeted attacks on speech-to-text. In: Proceedings of 2018 IEEE Security and Privacy Workshops; 2018 May 24; San Francisco, CA, USA; 2018. p. 1–7.
[48] Hannun A, Case C, Casper J, Catanzaro B, Diamos G, Elsen E, et al. Deep speech: scaling up end-to-end speech recognition. 2014. arXiv:1412.5567.
[49] Yakura H, Sakuma J. Robust audio adversarial example for a physical attack. 2018. arXiv:1810.11793.
[50] Liang B, Li H, Su M, Bian P, Li X, Shi W. Deep text classification can be fooled. 2017. arXiv:1704.08006.
[51] Huang S, Papernot N, Goodfellow I, Duan Y, Abbeel P. Adversarial attacks on neural network policies. 2017. arXiv:1702.02284.
[52] Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, et al. Playing Atari with deep reinforcement learning. 2013. arXiv:1312.5602.
[53] Mnih V, Badia AP, Mirza M, Graves A, Harley T, Lillicrap TP, et al. Asynchronous methods for deep reinforcement learning. In: Proceedings of the 33rd International Conference on Machine Learning; 2016 Jun 19–24; New York, NY, USA; 2016. p. 1928–37.
[54] Schulman J, Levine S, Moritz P, Jordan M, Abbeel P. Trust region policy optimization. In: Proceedings of the 32nd International Conference on Machine Learning; 2015 Jul 6–11; Lille, France; 2015. p. 1889–97.
[55] Behzadan V, Munir A. Vulnerability of deep reinforcement learning to policy induction attacks. In: Proceedings of the International Conference on Machine Learning and Data Mining in Pattern Recognition; 2017 Jul 15–20; New York, NY, USA; 2017. p. 262–75.
[56] Lin YC, Hong ZW, Liao YH, Shih ML, Liu MY, Sun M. Tactics of adversarial attack on deep reinforcement learning agents. 2017. arXiv:1703.06748.
[57] Carlini N, Katz G, Barrett C, Dill DL. Ground-truth adversarial examples. In: ICLR 2018 Conference; 2018 Apr 30; Vancouver, BC, Canada; 2018.
[58] Papernot N, Faghri F, Carlini N, Goodfellow I, Feinman R, Kurakin A, et al. Technical report on the CleverHans v2.1.0 adversarial examples library. 2016. arXiv:1610.00768v6.
[59] Sharma Y, Chen PY. Attacking the Madry defense model with L1-based adversarial examples. 2017. arXiv:1710.10733v4.
[60] Kurakin A, Goodfellow I, Bengio S. Adversarial machine learning at scale. 2016. arXiv: 1611.01236.
[61] Tramèr F, Kurakin A, Papernot N, Goodfellow I, Boneh D, McDaniel P. Ensemble adversarial training: attacks and defenses. 2017. arXiv:1705.07204.
[62] Kannan H, Kurakin A, Goodfellow I. Adversarial logit pairing. 2018. arXiv:1803.06373.
[63] Zheng S, Song Y, Leung T, Goodfellow I. Improving the robustness of deep neural networks via stability training. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA; 2016. p. 4480–8.
[64] Engstrom L, Ilyas A, Athalye A. Evaluating and understanding the robustness of adversarial logit pairing. 2018. arXiv: 1807.10272.
[65] Lee H, Han S, Lee J. Generative adversarial trainer: defense to adversarial perturbations with GAN. 2017. arXiv: 1705.03387.
[66] Liu X, Hsieh CJ. Rob-GAN: generator, discriminator, and adversarial attacker. 2018. arXiv:1807.10454v3.
[67] Xie C, Wang J, Zhang Z, Ren Z, Yuille A. Mitigating adversarial effects through randomization. 2017. arXiv: 1711.01991.
[68] Guo C, Rana M, Cisse M, van der Maaten L. Countering adversarial images using input transformations. 2017. arXiv: 1711.00117.
[69] Liu X, Cheng M, Zhang H, Hsieh CJ. Towards robust neural networks via random self-ensemble. In: Proceedings of the 2018 European Conference on Computer Vision; 2018 Sep 8–14; Munich, Germany; 2018. p. 369–85.
[70] Lecuyer M, Atlidakis V, Geambasu R, Hsu D, Jana S. Certified robustness to adversarial examples with differential privacy. 2018. arXiv:1802.03471v4.
[71] Dwork C, Lei J. Differential privacy and robust statistics. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing; 2009 May 31– Jun 2; Bethesda, MD, USA; 2009. p. 371–80.

[72] Li B, Chen C, Wang W, Carin L. Certified adversarial robustness with additive noise. 2018. arXiv: 1809.03113v6.
[73] Dhillon GS, Azizzadenesheli K, Lipton ZC, Bernstein J, Kossaifi J, Khanna A, et al. Stochastic activation pruning for robust adversarial defense. 2018. arXiv: 1803.01442.
[74] Luo T, Cai T, Zhang M, Chen S, Wang L. Random mask: towards robust convolutional neural networks. In: ICLR 2019 Conference; 2019 Apr 30; New Orleans, LA, USA; 2019.
[75] Xu W, Evans D, Qi Y. Feature squeezing: detecting adversarial examples in deep neural networks. 2017. arXiv: 1704.01155.
[76] Xu W, Evans D, Qi Y. Feature squeezing mitigates and detects Carlini/Wagner adversarial examples. 2017. arXiv: 1705.10686.
[77] He W, Wei J, Chen X, Carlini N, Song D. Adversarial example defenses: ensembles of weak defenses are not strong. 2017. arXiv: 1706.04701.
[78] Sharma Y, Chen PY. Bypassing feature squeezing by increasing adversary strength. 2018. arXiv:1803.09868.
[79] Samangouei P, Kabkab M, Chellappa R. Defense-GAN: protecting classifiers against adversarial attacks using generative models. 2018. arXiv:1805.06605.
[80] Shen S, Jin G, Gao K, Zhang Y. APE-GAN: adversarial perturbation elimination with GAN. 2017. arXiv: 1707.05474.
[81] Carlini N, Wagner D. MagNet and "efficient defenses against adversarial attacks" are not robust to adversarial examples. 2017. arXiv:1711.08478.
[82] Meng D, Chen H. MagNet: a two-pronged defense against adversarial examples. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security; 2017 Oct 30–Nov 3; New York, NY, USA; 2017. p. 135–47.
[83] Liao F, Liang M, Dong Y, Pang T, Hu X, Zhu J. Defense against adversarial attacks using high-level representation guided denoiser. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA; 2018. p. 1778–87.
[84] Athalye A, Carlini N. On the robustness of the CVPR 2018 white-box adversarial example defenses. 2018. arXiv:1804.03286.
[85] Raghunathan A, Steinhardt J, Liang P. Certified defenses against adversarial examples. 2018. arXiv:1801.09344.
[86] Raghunathan A, Steinhardt J, Liang P. Semidefinite relaxations for certifying robustness to adversarial examples. In: Proceedings of the 32nd Conference on Neural Information Processing Systems; 2018 Dec 3–8; Montréal, QC, Canada; 2018. p. 10877–87.
[87] Wong E, Kolter JZ. Provable defenses against adversarial examples via the convex outer adversarial polytope. In: Proceedings of the 31st Conference on Neural Information Processing Systems; 2017 Dec 4–9; Long Beach, CA, USA; 2017.
[88] Wong E, Schmidt FR, Metzen JH, Kolter JZ. Scaling provable adversarial defenses. 2018. arXiv:1805.12514.
[89] Sinha A, Namkoong H, Duchi J. Certifying some distributional robustness with principled adversarial training. 2017. arXiv:1710.10571.
[90] Namkoong H, Duchi JC. Stochastic gradient methods for distributionally robust optimization with *f*-divergences. In: Proceedings of the 30th Conference on Neural Information Processing Systems; 2016 Dec 5–10; Barcelona, Spain; 2016. p. 2208–16.
[91] Gao R, Kleywegt AJ. Distributionally robust stochastic optimization with Wasserstein distance. 2016. arXiv:1604.02199.
[92] Guo Y, Zhang C, Zhang C, Chen Y. Sparse DNNs with improved adversarial robustness. In: Proceedings of the 32nd Conference on Neural Information Processing Systems; 2018 Dec 3–8; Montréal, QC, Canada; 2018. p. 242–51.
[93] Hein M, Andriushchenko M. Formal guarantees on the robustness of a classifier against adversarial manipulation. In: Proceedings of the 31st Conference on Neural Information Processing Systems; 2017 Dec 4–9; Long Beach, CA, USA; 2017. p. 2266–76.
[94] Weng TW, Zhang H, Chen PY, Yi J, Su D, Gao Y, et al. Evaluating the robustness of neural networks: an extreme value theory approach. 2018. arXiv:1801.10578.
[95] Xiao KY, Tjeng V, Shafiullah NM, Madry A. Training for faster adversarial robustness verification via inducing ReLU stability. 2018. arXiv:1809.03008.
[96] Katz G, Barrett C, Dill DL, Julian K, Kochenderfer MJ. Reluplex: an efficient SMT solver for verifying deep neural networks. In: Proceedings of the International Conference on Computer Aided Verification; 2017 Jul 24–28; Heidelberg, Germany; 2017. p. 97–117.
[97] Wang Y, Jha S, Chaudhuri K. Analyzing the robustness of nearest neighbors to adversarial examples. 2017. arXiv: 1706.03922.
[98] Papernot N, McDaniel P. Deep *k*-nearest neighbors: towards confident, interpretable and robust deep learning. 2018. arXiv:1803.04765.
[99] Liu X, Li Y, Wu C, Hsieh C. Adv-BNN: improved adversarial defense through robust Bayesian neural network. 2018. arXiv:1810.01279.
[100] Neal RM. Bayesian learning for neural networks. New York: Springer Science & Business Media; 2012.
[101] Schott L, Rauber J, Bethge M, Brendel W. Towards the first adversarially robust neural network model on MNIST. 2018. arXiv:1805.09190.
[102] Xiao C, Deng R, Li B, Yu F, Liu M, Song D. Characterizing adversarial examples based on spatial consistency information for semantic segmentation. In: Proceedings of the European Conference on Computer Vision; 2018 Sep 8–14; Munich, Germany; 2018. p. 217–34.
[103] Yang Z, Li B, Chen PY, Song D. Characterizing audio adversarial examples using temporal dependency. 2018. arXiv:1809.10875.
[104] Chen PY, Zhang H, Sharma Y, Yi J, Hsieh CJ. Zoo: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security; 2017 Nov 3; Dalas, TX, USA; 2017. p. 15–26.
[105] Cao Y, Xiao C, Yang D, Fang J, Yang R, Liu M, et al. Adversarial objects against LiDAR-based autonomous driving systems. 2019. arXiv:1907.05418.
[106] Fawzi A, Fawzi O, Frossard P. Analysis of classifiers' robustness to adversarial perturbations. Mach Learn 2018;107(3):481–508.
[107] Mirman M, Gehr T, Vechev M. Differentiable abstract interpretation for provably robust neural networks. In: Proceedings of the 35th International Conference on Machine Learning; 2018 Jul 10–15; Stockholm, Sweden; 2018. p. 3578–86.
[108] Singh G, Gehr T, Mirman M, Puschel M, Vechev M. Fast and effective robustness certification. In: Proceedings of the 32nd Conference on Neural Information Processing Systems; 2018 Dec 3–8; Montréal, QC, Canada; 2018. p. 10802–13.
[109] Gowal S, Dvijotham K, Stanforth R, Bunel R, Qin C, Uesato J, et al. On the effectiveness of interval bound propagation for training verifiably robust models. 2018. arXiv:1810.12715.
[110] Dube S. High dimensional spaces, deep learning and adversarial examples. 2018. arXiv:1801.00634.
[111] Khoury M, Hadfield-Menell D. On the geometry of adversarial examples. 2018. arXiv:1811.00525.
[112] Gilmer J, Metz L, Faghri F, Schoenholz SS, Raghu M, Watterberg M, et al. Adversarial spheres. 2018. arXiv:1801.02774.
[113] Schmidt L, Santurkar S, Tsipras D, Talwar K, Madry A. Adversarially robust generalization requires more data. 2018. arXiv:1804.11285.
[114] Carlini N, Wagner D. Adversarial examples are not easily detected: bypassing ten detection methods. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security; 2017 Nov 3; Dalas, TX, USA; 2017. p. 3–14.
[115] Carlini N. Is AmI (attacks meet interpretability) robust to adversarial examples? 2019. arXiv:1902.02322v1.