Effect of Image Down-sampling on Detection of Adversarial Examples

Anjie Peng[1,2], Chenggang Li[1], Ping Zhu[3(✉)], Zhiyuan Wu[1], Kun Wang[1], Hui Zeng[1(✉)], and Wenxin Yu[1]

[1] Southwest University of Science and Technology, Sichuan, MY 621010, China
zengh5@mail2.sysu.edu.cn
[2] Science and Technology on Communication Security Laboratory, Sichuan, CD 610041, China
[3] Chengdu University of Information Technology, Sichuan, CD 610103, China

Abstract. Detecting adversarial examples and rejecting to input them into a CNN classifier is a crucial defense method to prevent the CNN being fooled by the adversarial examples. Considering that attackers usually utilize down-sampling to match the input size of CNN and the detection methods are commonly evaluated on down-sampled images, we study how the detectability of adversarial examples is affected by the interpolation algorithm if the legitimate image is down-sampled prior to be attacked. Since the down-sampling changes the relationships among neighboring pixels, the steganalysis-based detectors relying on the neighborhood dependencies are probably affected sharply. Experimental results on ImageNet verify that the detection accuracy varies among different interpolation kernels dramatically (the largest difference of accuracy is up to about 9%), and such novel phenomena appear valid universally across the tested CNN models and attack algorithms for the steganalysis-based detection method. Our work is of interest to both attackers and defenders for the purpose of benchmarking the attack algorithm and detection method respectively.

Keywords: Convolution Neural Network · Adversarial Image · Detection · Down-sampling

1 Introduction

Adversarial examples have attracted attentions to the security of convolution neural network (CNN) classifiers. Adversarial attacks, such as FGSM [1], BIM [2], DeepFool [3], BP [4], C&W [5], craft imperceptive perturbations on a legitimate image carefully to generate the adversarial image, and effectively force the CNN to misclassify the original ground truth label. This form of attack throws out some security threats in the CNN-based applications, especially in the security sensitive field, for instance, self-driving cars [6], robots [7]. How to harden CNNs against the adversarial attacks [8–13] is a hot topic.

This work was partially supported by NSFC (No. 41865006), Sichuan Science and Technology Program (No. 2022YFG0321, 2022NSFSC0916).

M. Tanveer et al. (Eds.): ICONIP 2022, CCIS 1791, pp. 550–561, 2023.
https://doi.org/10.1007/978-981-99-1639-9_46

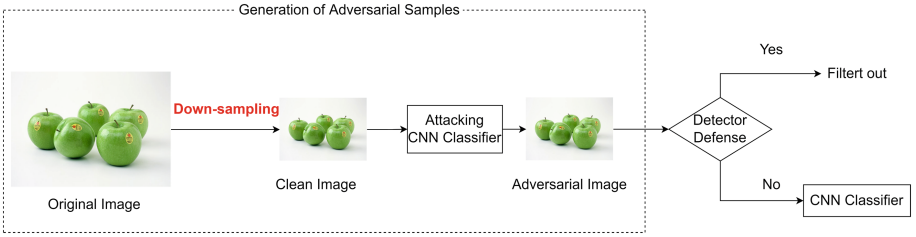

Fig. 1. The flow chart of a detector defense against the adversarial image. The detector defense filters out the adversarial image, and only feeds the clean image into the CNN for classification. In this work, we focus on how the pre-processing down-sampling in the process of generating an adversarial image (dashed box) affects the detectability of the detector defense.

Besides adversarial training [8,9,28], detecting the adversarial images and filtering out them before inputting them into the CNN is another important defense approach as illustrated in Fig. 1. Input transformation and steganalysis-based method are two typical detection algorithms. Since adversarial perturbations are not robust against image transformations, input transformation methods first feed a questioned image and its elaborately manipulated version into the CNN, and then detect the questioned image as adversarial if the CNN outputs are inconsistent before and after the transformation, such as the denoising filter composed by scalar quantization and smoothing spatial filter [14], feature squeezing (FS for short) [15], image quilting [16], image resampling [31]. As indicated by Goodfellow et al. [1] that the adversarial attack can be treated as a sort of accidental steganography, some steganalysis-based methods are proposed [17–21].

In this work, we study how the pre-processing down-sampling affects the detectability of adversarial images. Unlike the post-processing re-sampling used in the input transformation detection [31], the role of the down-sampling in our work is a pre-processing operation. We consider it as an important topic for several reasons. (1) Down-sampling is a commonly used operation when generating adversarial images as shown in Fig. 1. To save computation sources for the deep architecture, the size of input image for CNN is usually small. For example, ResNet [22] models accept RGB inputs of size $224 \times 224 \times 3$. To match the small input size of CNN, the image needs to be down-sampled before attacking. Some adversarial platforms employ different down-sampling algorithms for the attack. For example, Cleverhans[1](bilinear), EvadeML[2](nearest), RealSafe[3](bilinear), Foolbox[4](bicubic), Advertorch[5](bilinear). (2) For the purpose of benchmarking the detection method. Figure 1 shows that the down-sampling

[1] https://github.com/cleverhans-lab/cleverhans.
[2] https://evadeML.org/zoo.
[3] https://github.com/thu-ml/ares.
[4] https://github.com/bethgelab/foolbox.
[5] https://github.com/BorealisAI/advertorch.

possibly be a factor affecting the detectability of detector defense. Many detection methods [1,14–21] are evaluated on the down-sampled adversarial images but without considering the effects of down-sampling.

To our best knowledge, the role of the pre-processing down-sampling and its influence on detection method has not been studied so far. Many works [23–26] have analyzed the impacts of pre-processing and post-processing on steganalysis and forensics. Inspired by these works, we select three typical interpolation algorithms, two typical attacks, BIM [2], and C&W [5], two CNN models, ResNet-50 [22] and Inception-V3 [29], two typical detection methods, ESRM [19] and FS [15], for considerations. Experimental results reveal that the detection accuracies vary quite dramatically among different interpolation kernels and attack parameters for the state-of-the-art steganalysis-based method ESRM [19]. These results may provide some implications to attackers and defenders, and assist them develop their own optimal strategies to evade detection or improve defense ability.

2 Motivation Experiment

Jan Kodovský et.al. [23,24] find that down-sampling remarkably affects the steganalysis results. As steganography versus steganalysis is analogous to adversarial attack versus detection defense [17], we also study how the down-sampling affects the detectability of adversarial images.

To motivate our study, we select 1000 images from the validation dataset of ImageNet-1000(ILSVRC-2012) as source image database. Next, we prepare three kinds of down-sampled database generated on three commonly used interpolation kernels: nearest, bilinear and lanczos using resizing algorithm Resize $(\cdot)$ in PyTorch. All source images are down-sampled so that the smaller side of the image is 224 pixels, finally central-cropped to 224×224 pixels.

Fig. 2 illustrates the results of ESRM [19] detecting untargeted BIM adversarial images which are generated on ResNet-50. For each attack strength budget ϵ, an ESRM detector is constructed by training the ensemble classifier [27] with using ESRM feature. Half of the images are used for training and the other half are for testing, while the performance is evaluated by the detection accuracy (Acc) under equal number of legitimate images and adversarial images.

The results in Fig. 2 show that striking discrepancies of Acc is reflected in detecting different versions of down-sampled database. For example, at the attack strength budget $\epsilon = 1$, the Acc of nearest kernel is 83.1%, being about 10.0% lower than the Acc of the bilinear kernel. These results indicate that the choice of the interpolation kernel significantly affects the detectability, and thus a deeper understanding of this phenomenon is of a great importance for fairly benchmarking the detection method.

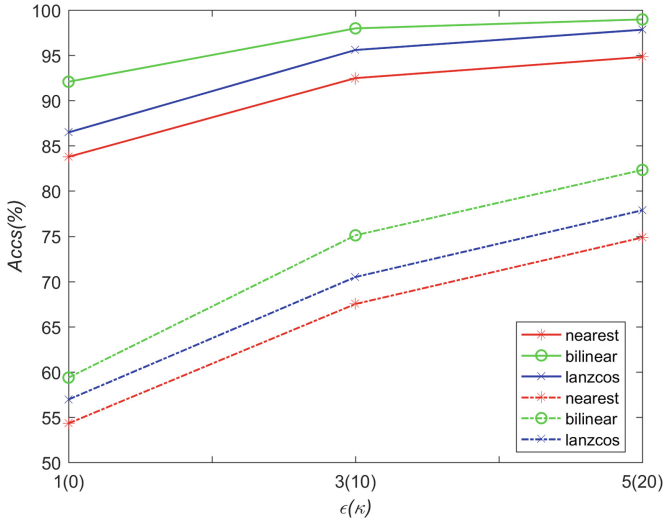

Fig. 2. The accuracies (*Accs, %*) of ESRM detecting BIM down-sampled adversarial images created with three different interpolation kernels against ResNet-50. The x axis is the attack strength budget ϵ.

3 Further Investigation

Inspired by the results shown in Fig. 2, we select two adversarial attacks (BIM [2], C&W [5]) attacking two CNN classifiers (ResNet-50 [22], Inception-V3 [29]), and further investigate how the interpolation kernel affects two detection methods (ESRM [19], FS [15]) on a larger image database.

3.1 Down-Sampling Algorithm

The image down-sampling is executed before attacking CNN classifiers as shown in Fig. 1. The down-sampling process is executed as follows: (1) Determine the position of new pixels based on the scaling factor; (2) Input the distances between the new pixels and neighbor old pixels to the interposition kernel to compute the weights; (3) Sum weights of intensities of neighbor old pixels as the values of new pixels. Obviously, the interpolation kernel and scaling factor are two primary factors. In this work, we focus on the interpolation kernel and employ the PyTorch function Resize $(\cdot)$ with three commonly used interpolation kernels nearest neighbor φ_n, bilinear φ_b, and lanczos φ_l in the experiments. As indicated by the formula (1)-(3), the bilinear and lanczos kernels consider more neighboring pixels than the nearest neighbor kernel. The bilinear and lanczos kernels are expected to cause stronger dependencies of neighboring pixels on the down-sampled images than the nearest neighbor kernel does.

The resolutions of the images in the validation dataset of ImageNet-1000 (ILSVRC-2012) are larger than the input size of our used ResNet-50 [22] and Inception-V3 [29], such as $500 \times 375 \times 3$, $500 \times 333 \times 3$, $375 \times 500 \times 3$, $500 \times 500 \times 3$. To match the input size of our CNNs, we first down-sample the short side to 224 and 299, and then crop the center part to form the resized image of size $224 \times 224 \times 3$ for ResNet-50 and $299 \times 299 \times 3$ for Inception-V3 respectively.

$$\varphi_n(x) = \begin{cases} 1, & -\frac{1}{2} \leq x < \frac{1}{2} \\ 0, & otherwise \end{cases} \tag{1}$$

$$\varphi_b(x) = \begin{cases} 1 - |x|, & |x| \leq 1 \\ 0, & otherwise \end{cases} \tag{2}$$

$$\varphi_l(x) = \begin{cases} 1, & x = 0 \\ \frac{2sin(\pi x)sin(\frac{\pi x}{2})}{\pi^2 x^2}, & 0 < |x| < 2 \\ 0, & otherwise \end{cases} \tag{3}$$

3.2 Adversarial Attacks

After creating the down-sampled images to match the input size of CNN, the adversarial image is generated on Advertorch platform[6]. Two typical attack algorithms BIM [2] and C&W [5] are considered for attacking against commonly used pre-trained CNN models ResNet-50 [22][7] and Inception-V3 [29][8] respectively. BIM [2] is an iterative gradient-based attack. For an image x of label y^{true}, with initializing $x_0^{adv} = x$, it can be formulated as (4), where $f(\cdot)$ is a CNN classifier, $\nabla(J(\cdot))$ is the gradient of the loss function $J(\cdot)$, $clip_{x,\epsilon}(\cdot)$ limits the perturbation is less than ϵ. We set the iteration number to be 10, $\alpha = 1$ and $\epsilon = 1, 3, 5$ to ensure attacking successfully and adding imperceptive perturbations on the legitimate image. C&W [5] is an optimization-based attack. It optimizes the problem (5) to generate adversarial images, where c is a hyperparameter tuning the L_2 distance and the prediction function $F(x^{adv}) = max(Z(x^{adv})_{l^*} - max\{Z(x^{adv})_i, i \neq l^*\}, -\kappa)$ for untargeted attack which makes the true label l^* least-likely. We set the confidence $\kappa = 0, 10, 20$ and the other parameters used default value in the Advertorch platform.

$$x_{i+1}^{adv} = x_i^{adv} + clip_{x,\epsilon}(\alpha sign(\nabla_{x_i^{adv}} J(f(x_i^{adv}), y^{true}))) \tag{4}$$

$$\begin{aligned} &\arg\min_{x^{adv}} ||x^{adv} - x||_2 + cF(x^{adv}) \\ &s.t. x_{ij}^{adv} \in [0, 1] \end{aligned} \tag{5}$$

[6] https://github.com/BorealisAI/advertorch.
[7] https://download.pytorch.org/models/resnet50-0676ba61.pth.
[8] https://download.pytorch.org/models/inception_v3_google-0cc3c7bd.pth.

3.3 Detection Results of ESRM for Different Interpolation Kernels

ESRM [19] is a steganalysis-based detection method. It enhances the steganalysis feature SRM [30] via considering the modification probability of each pixel and allocating large weights to the probably modified pixels when calculating co-occurrences. As ESRM feature is a composition of co-occurrences of multiple high frequency residuals, it yields tremendous dimensions up to 34671 and it is susceptible to neighboring pixels dependencies. The FLD ensemble classifier with default settings is employed [19,27] to construct the binary detector for detecting adversarial images from legitimate images.

To evaluate the detection accuracy of ESRM, we randomly select 5000 images that consists of 5 images per class from the famous validation dataset of ImageNet-1000(ILSVRC-2012) as the source dataset. The ratio of training samples and testing samples is 1:1.

Table 1 denotes the results of ESRM detecting BIM and C&W adversarial images attacking ResNet-50. It is shown that the *Acc* varies sharply when detecting different down-sampled adversarial images. For each attack strength, the *Acc* of detecting bilinear down-scaled images is the highest, while that of detecting nearest neighbor down-scaled images is the lowest. The minimum and maximum difference between them are 4.15% (at BIM, $\epsilon = 5$) and 8.3% (at BIM, $\epsilon = 1$) respectively. The mainly reason for these discrepancies of *Acc* is that different interpolation kernels result in different dependencies of neighboring pixels which finally cause different detection accuracies of ESRM method. For the down-scaling with the nearest interpolation kernel, it skips some original pixels and assigns the new resized pixel value based on the formula (1) via replacing it with the nearest original pixel value. However, for the bilinear and lanczos kernel, the pixel values in the down-scaled image are interpolated as a certain linear combination of the original pixel values. Obviously, for the down-sampling fixing other parameters, the bilinear and lanczos kernel results in stronger neighboring pixel dependencies than the nearest kernel. As indicated in steganalysis [23,24,30], the stronger dependencies will be disturbed more when adding adversarial perturbations onto the legitimate image to generate the adversarial image. This means that attacking on the bilinear and lanczos down-scaled image will alter more on neighboring pixel dependencies than attacking on the nearest neighbor down-scaled image. Since ESRM feature is based on the neighboring pixel dependencies, it is expected that ESRM detector possesses superior detectability on the bilinear and lanczos resized adversarial images than on the nearest resized adversarial images as empirically verified in Table 1.

To further verify our investigation, we repeat experiments of ESRM detecting adversarial images attacking Inception-V3. The results in Table 2 also illustrate that the *Acc* varies when detecting different down-sampled adversarial images. Similarly, the *Acc* of detecting nearest, lanczos, and bilinear resized images are ascending. Because the down-scaling factor used for Inception-V3 is lower than that for ResNet-50 and the changes on neighboring pixel dependencies are weakened. Compared with the results in Table 1, the discrepancies of *Acc* for

Table 1. The *Acc (%)* of **ESRM** detecting different down-sampled adversarial images attacking **ResNet-50**.

BIM	Interpolation kernel	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$
	nearest	83.78	92.48	94.82
	bilinear	92.08	92.08	98.97
	lanczos	86.48	86.48	97.83
C&W	Interpolation kernel	$\kappa = 0$	$\kappa = 10$	$\kappa = 20$
	nearest	54.31	67.51	74.86
	bilinear	59.37	59.37	82.33
	lanczos	82.33	70.49	77.87

detecting three kinds of down-scaled adversarial images become smaller as shown in Table 2.

The results in Tables 1 and 2 indicates that ESRM feature based on the neighboring pixel dependencies is affected by the interpolation kernel, and thus the detectability of ESRM detector is different for detecting adversarial images generated from different down-sampled legitimate images. Besides, we figure out that detecting BIM adversarial images are easier than detecting C&W adversarial images. Because BIM adds more adversarial perturbations onto legitimate image, which disturbs the neighboring pixel dependencies more. Under this reason, detecting the adversarial images generated by stronger attack strength(larger ϵ, κ) also become easier as shown in Tables 1 and 2.

Table 2. The *Acc (%)* of **ESRM** detecting different down-sampled adversarial images attacking **Inception-V3**.

BIM	Interpolation kernel	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$
	nearest	89.28	94.71	95.95
	bilinear	91.86	97.83	98.38
	lanczos	89.84	96.39	97.50
C&W	Interpolation kernel	$\kappa = 0$	$\kappa = 10$	$\kappa = 20$
	nearest	58.44	69.71	79.37
	bilinear	63.54	75.89	85.31
	lanczos	60.61	72.72	82.28

3.4 Detection Results of FS for Different Interpolation Kernels

FS [15] is a typical transformation-based method, which is built on the assumption that the legitimate image is more robust against image transformations than the adversarial image. For a questioned image, the detection process of FS is as follows. (1) Employ bit depth reduction, median filtering, and non-local mean

methods successively to squeeze input space. (2) Input the original image and its squeezed version to the CNN to get the corresponding SoftMax outputs. (3) Calculate the L_1 distance of two SoftMax outputs. (4) Compare the distance with a threshold distance T and predict the one whose distance is larger than T as adversarial, otherwise legitimate. The default best joint detection method of FS is employed in the experiment, where the threshold T is determined via fixing FPR=5% (*i.e.*, at most 5% legitimate images are misclassified as adversarial). We randomly select 11000 images that consists of 11 images per class from ILSVRC-2012 as the source dataset. One half of the legitimate image is used for determining the threshold T. We report the detection accuracy of the other half legitimate images and their corresponding adversarial images.

The experimental results in Tables 3 and 4 show that the *Accs* of detecting nearest, bilinear and lanczos resized image are different at the same attack strength level. The greatest difference among them occurs at detecting C&W on ResNet-50 with $\kappa = 0$, where the *Acc* of detecting nearest, bilinear and lanczos resized image are 61.55%, 72.09% and 73.47% respectively. On the contrary, for a stronger attack or a more robust CNN classifier, the difference in the *Acc* of detecting different down-sampled images has become smaller.

Notice that FS is different from ESRM, it is a transformation-based detector, whose detectability is determined not only by the difference between the legitimate image and adversarial image, but also by the robustness of the CNN classifier. Hence, we find out the reason for the above decreased discrepancy probably be that the robust CNN model and the strong attack smooth out the differences caused by different interpolation kernels. For example, the classification accuracies of pre-trained Inception-V3 [29] on nearest, bilinear and lanczos resized images are 84.53%, 84.63% and 85.07% respectively. These similar results indicate that these CNN models have similar robustness against image squeezing operations, so it is expected that the Accs are nearly same for different down-scaled images as shown in Table 4. The attack with strong attack strength also yields *Acc*s be nearly same as shown in Table 3.

Table 3. The *Acc (%)* of **FS** detecting different down-sampled adversarial images attacking **ResNet-50**.

BIM	Interpolation kernel	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$
	nearest	78.16	58.55	52.80
	bilinear	77.43	56.85	51.32
	lanczos	78.58	57.45	51.61
C&W	Interpolation kernel	$\kappa = 0$	$\kappa = 10$	$\kappa = 20$
	nearest	61.55	82.38	77.99
	bilinear	72.09	83.69	79.99
	lanczos	73.47	84.38	79.77

Table 4. The *Acc (%)* of **FS** detecting different down-sampled adversarial images attacking **Inception-V3**.

BIM	Interpolation kernel	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$
	nearest	81.90	63.28	56.47
	bilinear	81.22	62.86	55.69
	lanczos	84.51	64.00	56.79
C&W	Interpolation kernel	$\kappa = 0$	$\kappa = 10$	$\kappa = 20$
	nearest	81.95	90.43	83.87
	bilinear	82.55	90.03	82.92
	lanczos	83.59	90.74	83.30

Notice that FS is different from ESRM, it is a transformation-based detector, whose detectability is determined not only by the difference between the legitimate image and adversarial image, but also by the robustness of the CNN classifier. Hence, we find out the reason for the above decreased discrepancy probably be that the robust CNN model and the strong attack smooth out the differences caused by different interpolation kernels. For example, the classification accuracies of pre-trained Inception-V3 [29] on nearest, bilinear and lanczos resized images are 84.53%, 84.63% and 85.07% respectively. These similar results indicate that these CNN models have similar robustness against image squeezing operations, so it is expected that the Accs are nearly same for different down-scaled images as shown in Table 4. The attack with strong attack strength also yields Accs be nearly same as shown in Table 3.

3.5 Discussion

The results in Tables 1, 2, 3 and 4 verify that the pre-processing down-sampling affects the detectability of the adversarial examples for the detector defense. As diverse interpolation kernels change the dependencies of neighboring pixels differently, the down-sampling brings different impacts on the detectability of the steganalysis-based detector ESRM which is relied on the neighborhood dependencies. The bilinear and lanczos kernel results in stronger neighboring pixel dependencies than the nearest kernel. Generally, the stronger neighborhood dependencies will be disturbed more by the adversarial perturbation, thus causing the bilinear and lanczos interpolated down-sampled images are easier to be detected the nearest down-sampled images. Simultaneously, the down-sampling also affects the detectability of the image transformation detector FS, but with smaller differences of influences among different interpolation kernels.

The experimental results may give some implications for attackers and defenders to develop their own optimal strategies under the attack and defense confrontation situation. Since ESRM and FS bear relative low detection accuracies on the nearest down-scaled adversarial images, in order to evade the detection defense as much as possible, attackers tend to apply the nearest neighbor

down-sampled images for adversarial attack. For defenders, as indicated by the detection results of ResNet-50 and Inception-V3, develop more accurate and robust CNN model is a choice of improving the defense ability.

4 Conclusion

The down-sampling is usually applied before generating the adversarial images. In this paper, we study how the pre-processing down-sampling affects the detectability of adversarial images. To our best knowledge, this paper is the first work related to the influence of down-sampling on the detectability, and it reveals a surprising sensitivity of steganalysis-based detection to the choice of the interpolation kernel. To get complete empirically investigations, experiments are executed on three interpolation kernels, two qualitatively different attack algorithms, BIM and C&W, and two state-of-the-art detecting methods named FS and ESRM. Since down-sampling alters the strength of dependencies among neighboring image pixels, experimental results verify that the detectability of steganalysis-based feature ESRM is affected heavily by the interpolation kernel used in the down-sampling. Besides, the detectability of FS is also affected by the interpolation kernel, but it is less affected than ESRM does.

The main contribution of this paper is explaining how the detectability of adversarial images varies with the interpolation algorithms and its settings. Our work is probably advantageous for attackers and defenders to benchmark their performance as full as possible under the attack and defense confrontation situation. In the future, we will study how some other pre-processing factors (such as smoothing, noising) affect the detectability of adversarial images.

References

1. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Machine Learning, pp. 1–10 (2015)
2. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations. arXiv:1706.06083 (2018)
3. Moosavi-Dezfooli, S.M., Fawzi, A. and Frossard, P.: Deepfool: a simple and accurate method to fool deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2574–2582 (2016)
4. Zhang, H., Avrithis, Y., Furon, T., Amsaleg, L.: Walking on the edge: fast, low-distortion adversarial examples. IEEE Trans. Inf. Forensics Secur. **16**, 701–713 (2020)
5. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy, pp. 39–57 (2017)
6. Bourzac, K.: Bringing big neural networks to self-driving cars, smartphones, and drones. IEEE Spectrum, 13–29 (2016)
7. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015)
8. Wong, E., Rice, L., Kolter, J., Z.: Fast is better than free: Revisiting adversarial training. arXiv:2001.03994 (2020)

9. Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., Jordan, M.: Theoretically principled trade-off between robustness and accuracy. In: International Conference on Machine Learning, pp. 7472–7482 (2019)
10. Machado, G.R., Silva, E., Goldschmidt, R.R.: Adversarial machine learning in image classification: a survey toward the defender's perspective. ACM Comput. Surv. (CSUR) **55**(1), 1–38 (2021)
11. Grosse, K., Manoharan, P., Papernot, N., Backes, M., McDaniel, P.: On the (statistical) detection of adversarial examples. arXiv:1702.06280 (2017)
12. Lu, J., Issaranon, T., Forsyth, D.: Safetynet: detecting and rejecting adversarial examples robustly. In: Proceedings of the IEEE International Conference On Computer Vision, pp. 446–454 (2017)
13. Li, X. and Li, F.: Adversarial examples detection in deep networks with convolutional filter statistics. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5764–5772 (2017)
14. Liang, B., Li, H., Su, M., Li, X., Shi, W., Wang, X.: Detecting adversarial image examples in deep neural networks with adaptive noise reduction. IEEE Trans. Dependable Secur. Comput. **18**(1), 72–85 (2018)
15. Xu, W., Evans, D. and Qi, Y.: Feature squeezing: Detecting adversarial examples in deep neural networks. In: Network and Distributed System Security Symposium. arXiv:1704.01155 (2017)
16. Guo, C., Rana, M., Cisse, M., Van Der Maaten, L.: Countering adversarial images using input transformations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. arXiv:1711.00117 (2017)
17. Schöttle, P., Schlögl, A., Pasquini, C., Böhme, R.: Detecting adversarial examples-a lesson from multimedia security. In: 2018 26th European Signal Processing Conference (EUSIPCO), pp. 947–951 (2018)
18. Fan, W., Sun, G., Su, Y., Liu, Z., Lu, X.: Integration of statistical detector and Gaussian noise injection detector for adversarial example detection in deep neural networks. Multimed. Tools Appl. **78**(14), 20409–20429 (2019). https://doi.org/10.1007/s11042-019-7353-6
19. Liu, J., et al.: Detection based defense against adversarial examples from the steganalysis point of view. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4825–4834 (2019)
20. Bonnet, B., Furon, T., Bas, P.: Forensics through stega glasses: the case of adversarial images. In: International Conference on Pattern Recognition, pp. 453–469 (2021)
21. Peng, A., Deng, K., Zhang, J., Luo, S., Zeng, H., Yu, W.: Gradient-based adversarial image forensics. In: International Conference on Neural Information Processing, pp. 417–428 (2020)
22. He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
23. Kodovský, J., Fridrich, J.: Steganalysis in resized images. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2857–2861 (2013)
24. Kodovský, J., Fridrich, J.: Effect of image downsampling on steganographic security. IEEE Trans. Inf. Forensics Secur. **9**(5), 752–762 (2014)
25. Stamm, M.C., Wu, M. and Liu, K.R.: Information forensics: An overview of the first decade. IEEE access. **1**, 167–200 (2013). (Kang, X., Stamm, M.C., Peng, A. and Liu, K.R.: Robust median filtering forensics using an autoregressive model. IEEE Trans. Inf. Forensics Secur. **8**(9), pp. 1456–1468 (2013))

26. Kang, X., Stamm, M.C., Peng, A., Liu, K.R.: Robust median filtering forensics using an autoregressive model. IEEE Trans. Inf. Forensics Secur. **8**(9), 1456–1468 (2013)
27. Kodovsky, J., Fridrich, J., Holub, V.: Ensemble classifiers for steganalysis of digital media. IEEE Trans. Inf. Forensics Secur. **7**(2), 432–444 (2011). (Dong, Y., et al.: Benchmarking adversarial robustness on image classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 321–331 (2020))
28. Dong, Y., et al.: Benchmarking adversarial robustness on image classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 321–331 (2020)
29. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
30. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. **7**(3), 868–882 (2012)
31. Mustafa, A., Khan, S.H., Hayat, M., Shen, J., Shao, L.: Image super-resolution as a defense against adversarial attacks. IEEE Trans. Image Process. **29**, 1711–1724 (2020)