Can Shape Structure Features Improve Model Robustness under Diverse Adversarial Settings?

Mingjie Sun [1*] Zichao Li [2*] Chaowei Xiao [3,4,8*]
Haonan Qiu [5] Bhavya Kailkhura [7] Mingyan Liu [8] Bo Li [6]
[1]CMU [2]UCSC [3]NVIDIA [4]ASU [5]NTU [6]UIUC
[7]Lawrence Livermore National Laboratory [8]University of Michigan, Ann Arbor
sunmj15@gmail.com, zil023@ucsd.edu, xiaocw@umich.edu
qhnmoon@gmail.com, kailkhura1@llnl.gov,mingyan@umich.edu, lbo@illinois.edu

Abstract

Recent studies show that convolutional neural networks (CNNs) are vulnerable under various settings, including adversarial attacks, common corruptions, and backdoor attacks. Motivated by the findings that human visual system pays more attention to global structure (e.g., shapes) for recognition while CNNs are biased towards local texture features in images, in this work we aim to analyze whether "edge features" could improve the recognition robustness in these scenarios, and if so, to what extent? To answer these questions and systematically evaluate the global structure features, we focus on shape features and propose two edge-enabled pipelines EdgeNetRob and EdgeGANRob, forcing the CNNs to rely more on edge features. Specifically, EdgeNetRob and EdgeGANRob first explicitly extract shape structure features from a given image via an edge detection algorithm. Then EdgeNetRob trains downstream learning tasks directly on the extracted edge features, while EdgeGANRob reconstructs a new image by refilling the texture information with a trained generative adversarial network (GANs). To reduce the sensitivity of edge detection algorithms to perturbations, we additionally propose a robust edge detection approach Robust Canny *based on vanilla Canny. Based on our evaluation, we find that EdgeNetRob can help boost model robustness under different attack scenarios at the cost of the clean model accuracy. EdgeGANRob, on the other hand, is able to improve the clean model accuracy compared to EdgeNetRob while preserving the robustness. This shows that given such edge features, how to leverage them matters for robustness, and it also depends on data properties. Our systematic studies on edge structure features under different settings will shed light on future robust feature exploration and optimization.*

1. Introduction

Convolutional neural networks (CNNs) have been studied extensively [17], and have achieved state-of-the-art performance in many learning tasks [7, 11, 14, 21, 31, 38, 44, 48, 49, 52, 66, 67, 69]. However, different from the human cognition system, recent works have shown that CNNs are vulnerable to *adversarial attacks* [4, 5, 19, 40, 40–42, 51, 58–62], where imperceptible perturbation can be added to the test data to tamper the predictions. Different from *adversarial examples* where test data is manipulated, *data poisoning* or *backdoor attacks*, where training data is manipulated to reduce model's generalization ability, have also been proposed [9, 33]. In addition, recent studies show that CNNs tend to learn spurious statistical features instead of high level abstraction, making it fail to generalize under common corruptions (e.g. fog and snow) [22]. For each of these settings, different robust algorithms have been proposed to solve them *independently*. For instance, adversarial training based methods [19, 39, 47] are proposed to improve the robustness against adversarial attacks but are inefficient to backdoor attacks; spectral signature [54] is designed for defending against backdoor attacks while remains vulnerable to adversarial attacks and common corruptions. Given existing studies on human visual systems, in this paper we aim to ask: Is it possible to learn semantically meaningful structure features to simultaneously improve the robustness of DNNs under different settings including adversarial attacks, backdoors, and common corruptions?

To improve the general robustness of CNNs under different attacks, recent studies explore the underlying cause of their vulnerability. Ilyas et al. [25] attributes the existence of adversarial examples to the non-robust but highly-predictive features. They suggest to train a classifier only on "robust features" which contain the necessary information for recognition and are insensitive to small perturbations. In addition, Baker et al. [2] and Geirhos et al. [16] have

[*] indicates equal contributions. Our code is available at https://github.com/Eric-mingjie/Shape-Robustness

shown that human recognition relies mainly on global object shapes rather than local patterns (e.g. textures), while CNNs are more biased towards the latter. Geirhos et al. [16] creates a texture-shape cue conflict, such as a cat shape with elephant texture, and feeds it to a CNN model trained with IamgeNet and human respectively. While human can still recognize it as a cat, CNN incorrectly predicts it as an elephant. Landau et al. [32] have also shown that the shape of objects is the most important cue for human object recognition.

Given the above observation, natural questions emerge: *Can we improve the robustness of CNNs under different attacks by making it rely more on global shape structure? What are the conditions that affect such robustness improvement?* In this paper, we aim to answer the above questions by quantitatively evaluating whether the shape structure features could improve model robustness under different attacks settings, and how to leverage such features. In particular, we focus on a specific type of shape representation: edges (image points that have sharp changes in brightness). Edge features come with two benefits: 1) it is an effective way for modelling shape; 2) edges are easy to be captured in images, with many algorithms [3, 36, 65] available.

To evaluate different ways of leveraging such shape features, in this paper we explore two edge feature enabled pipelines EdgeNetRob and EdgeGANRob. The framework is shown in Figure 1. As illustrated, the pipeline of EdgeNetRob (grey lines) is a simple yet efficient approach which extracts the structural (edge) information via an edge detection algorithm and then trains the classifier on the extracted edge features. As a result, EdgeNetRob forces the CNNs to make predictions solely based on shape information rather than texture/color, thus eliminating the texture bias [16]. Comparing with the adversarial training based methods, EdgeNetRob is more general and efficient since it does not need to generate adversarial examples during training. However, one potential problem for EdgeNetRob is that the algorithm may decrease the clean accuracy of CNNs due to the missing texture/color information. *Could we refill the texture/color information based on the extracted edge features to improve the robustness?* To answer this question, we explore the pipeline EdgeGANRob (blue lines in Figure 1), which embeds a generator to refill the texture/colors based on extracted edge information.

To extract the edge information, we first leverage two standard edge detection algorithms: Canny [3] and a network-based detection algorithm, RCF [36]. However, our results show that by simply applying these edge detection algorithms to EdgeNetRob, the models are still vulnerable to sophisticated adaptive attacks. Thus, we propose a robust edge detection algorithm, s *Robust Canny*. We show *Robust Canny* is able to significantly improve the robustness of EdgeNetRob and EdgeGANRob.

We evaluate EdgeNetRob and EdgeGANRob on four datasets with clear edge information (Fashion MNIST, CelebA), and unclear or complicated edge information (CIFAR-10, Tiny-ImageNet) among different attack settings (e.g., adversarial attacks, common corruptions, and backdoor attacks). Our results show that edge features are able to improve the model robustness under these settings. The clean accuracy could be improved by refilling the texture information on the extracted edges via GANs on datasets with clear edge information. However, for datasets with complicated or less clear edge information, the clean accuracy can barely be improved by only refilling the texture information and further studies are required. We believe this work will open new directions for understanding shape features and designing more robust structural features to improve the model robustness against different attacks. Please find more visualization results on the anonymous website: https://sites.google.com/view/edge-robustness.

The main contributions of this paper are as follows: (i) We propose two shape feature enable pipelines EdgeNetRob and EdgeGANRob to evaluate whether the shape-based feature could improve the model robustness under different adversarial scenarios, including adversarial attacks, common corruption, and backdoor attacks. (ii) We propose a robust edge detection algorithm *Robust Canny* to improve the robustness of edge detection against sophisticated adaptive attacks. (iii) We conduct comprehensive experiments under various settings with different datasets. We show that such shape structure features can indeed improve model robustness under different adversarial scenarios, while sometimes at the cost of sacrificing certain clean accuracy depending on data properties.

2. Related work

Adversarial attacks Adversarial examples are the clean images perturbed by carefully designed perturbations to mislead machine learning models. These could be viewed as worst-case analysis of the models robustness. A wide range of methods against adversarial examples have been proposed [34, 50], among which many are shown to be not robust against adaptive attacks [1, 6]. The gradient obfuscation has been identified as a common pitfall for defense methods [1], thus suggested that defense methods should be evaluated against sophisticated adaptive attacks [6]. The current effective defense methods are based on adversarial training [19, 39].

Common corruptions Different from adversarial examples, Common Corruptions proposed by Hendrycks and Dietterich [22] aim to measure the model's generalization by using the unseen corruptions. They consist of 15 types of algorithmically generated corruptions including noise, blur, weather and digital categories. Current DNNs are vulnerable to these corruptions.

Backdoor attacks Backdoor attack [8, 20] is a type of

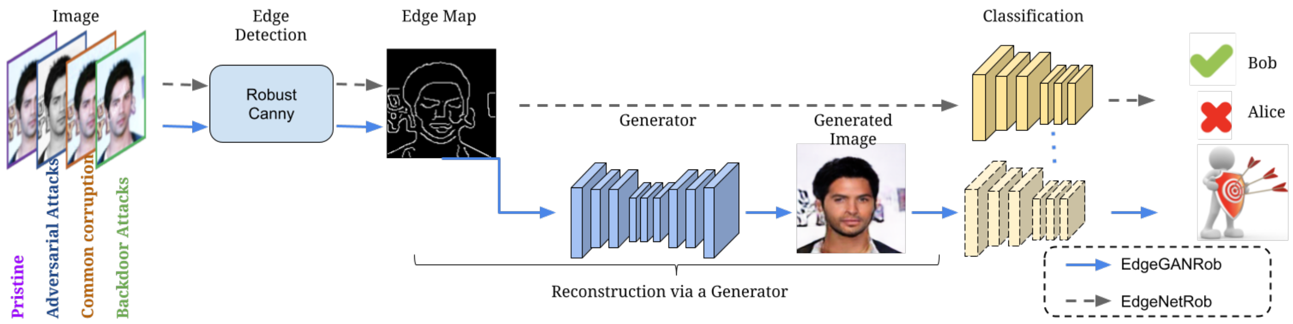

Figure 1: Overview of the edge feature enabled pipelines. EdgeNetRob directly feeds the output of an edge detector to the classifier; while EdgeGANRob refill the edge images with texture information to reconstruct a new instance for prediction.

poisoning attack [45] by injecting a backdoor pattern into training data. As a result, the trained models will predict a test instance with the backdoor pattern as the specific target. Tran et al. [54] has proposed to detect poisoned training data by using tools from robust statistics. Liu et al. [35] proposes an approach to protect models from backdoor attacks via neuron pruning.

Although the above methods achieve the improvement of the robustness individually, none of them could not simultaneously improve the robustness under different attacks including adversarial attacks, common corruptions and backdoor attacks. In this work, we propose a single method to improve the robustness under different attacks including adversarial attacks, common corruptions and backdoor attacks.

Semantically robust features. Recent work has highlighted a connection between recognition robustness and robust features. For image recognition, [2, 16] shows that CNNs rely more on textures than global shape structure, while humans rely more on shape structure than detailed texture. [27] uses visualization methods and finds that adversarially robust models tend to capture global structure of the objects. [25] argues that there exists non-robust features in natural images which are highly predictive but not interpretable by human. These work shows that it is possible to improve the robustness of CNNs by learning from *robust* structural features. However, they did not directly identify which features are robust. In this work, we propose to explicitly use edge as a robust feature proxy to evaluate .

3. Edge Feature Enabled Pipelines

To evaluate the robustness of edge features, we introduce two edge-based pipelines EdgeNetRob and EdgeGANRob. Both pipelines first extract the edge information and then train the classifier based on either the edge images or GAN filled images. In this section, we first introduce a simple but efficient algorithm EdgeNetRob: We use the edge image extracted by standard edge detection algorithms, but find that these edge detection algorithms are not robust against sophisticated adaptive attacks. Thus, we propose a robust edge detection algorithm, Robust Canny. We then introduce EdgeGANRob, which refills the texture information of the extract edge image with a trained GAN. Finally, we describe three settings for robustness evaluation.

3.1. EdgeNetRob

EdgeNetRob consists of two stages: First, we exploit an edge detection method to extract edge maps from an image, and then a standard image classifier $f_\theta(\cdot)$ is trained on the extracted edge maps. Formally, denoting the edge extractor function as $e(\cdot)$, the EdgeNetRob pipeline aims to solve the following problem: $\min_\theta \mathbb{E}_{(x,y)\sim D}\left[\mathcal{L}\left(f_\theta\left(e(x)\right),y\right)\right]$ where D represents the underlying data distribution and $\mathcal{L}$ denotes the loss function (e.g., cross-entropy loss). EdgeNetRob forces the decision of CNN to be solely based on edges, thus making it less sensitive to local textures. Compared with the adversarial training-based methods [39, 68], EdgeNetRob is simple, efficient, and scalable which could be applied to large-scale datasets. However, despite the simplicity and efficiency of EdgeNetRob, it may degrade the performance of CNNs over clean test data given that the texture/color information is missing. We will provide detailed discussion about this tradeoff in Section 4.

3.2. Robust Edge Detection

Based on our analysis, the vanilla edge detection algorithm such as Canny and RCF could be vulnerable against adaptive attacks given that they are both end to end models. For instance, Cosgrove and Yuille [12] finds that neural network based edge detectors such as HED [65] can fail easily when facing adversarial perturbation. Though traditional edge detection methods such as Canny [3] is intrinsically robust since they output binary edge maps, as illustrated in Figure 2 (first row), when we apply a sophisticated adaptive attack based on a differentiable proxy of Canny and generate adversarial perturbation, the output of Canny edge detector can become noisy and incorrect. Thus, in this section we propose a robust edge detection algorithm name

Robust Canny to improve the robustness of vanilla Canny by truncating the noisy pixels in its intermediate stages.

Specifically, there are 6 stages in our proposed Robust Canny: (1) *Noise reduction*: A Gaussian filter is applied to smooth the image. (2) *Gradient computation*: We apply the Sobel operator [29] to compute the gradient magnitude and direction at each pixel from the smoothed images. (3) *Noise masking*: We reduce the noise in the presence of adversarial perturbations by thresholding the gradient magnitudes by a level α. (4) *Non-maximum suppression (NMS)*: An edge thinning step is taken to deblur the output of the Sobel operator. Gradient magnitudes that are not at a maximum along the direction of the gradient are suppressed (set to zero). (5) *Double threshold*: Apply lower and upper thresholds (θ_l, θ_h) for the gradient magnitude after NMS, and pixels are then mapped to 3 levels: strong, weak, and non-edge pixels. (6) *Edge tracking by hysteresis*: Edge pixels are detected by searching for strong pixels, or weak pixels that are connected to other strong pixels.

We could observe that we have modified the vanilla Canny algorithm by adding a noise masking stage after computing the image gradients. Later in Figure 2, we show that the gradient computation stage is sensitive to input perturbations. Thus, we set all gradient magnitudes lower than a threshold α to zero to mitigate the perturbation noise. By adding a truncation operation, it is expected that adversarial noise on the gradient map with small magnitude will be reduced in early stages without affecting the quality of final edge maps. In addition to the masking operation, we find that the parameters of Canny (e.g. standard deviation of gaussian filter σ, thresholds θ_l, θ_h) also affect the robustness level. Specifically, we notice that larger σ and higher thresholds θ_l, θ_h result in higher robustness due to the stronger smoothing and pruning effects. This, however, comes at the cost of clean accuracy drop, as larger σ leads to blurrier images and higher θ_l, θ_h may eliminate useful information in the output edges. To obtain a robust edge detector, we should carefully choose its parameters (e.g., $\sigma, \theta_l, \theta_h$). More details are provided in the experiment section.

3.3. EdgeGANRob

As EdgeNetRob may decrease the clean accuracy due to the loss of texture information, here we propose EdgeGANRob, which embeds a generative model to refill the texture/colors for the edge images generated by EdgeNetRob as shown in Figure 1, and therefore improve clean accuracy. The core component of EdgeGANRob is the refilling network, for which we use an inpainting Generative Adversarial Network (GAN) [18]. Next, we describe how we train the inpainting GAN in EdgeGANRob. Recall that the task of generating color images from edge maps is well defined under the image-to-image translation framework (pix2pix) [26]. We train our inpainting GAN with two steps: first, we follow the common setup of pix2pix [26, 56] to train a conditional GAN using the following objective function: (we use G and D to denote the generator and discriminator networks.)

$$\min_G \max_D \mathcal{L}_{gan} = \min_G \left(\lambda_{adv} \max_D \mathcal{L}_{adv} + \lambda_{FM} \mathcal{L}_{FM} \right) \tag{1}$$

where $\mathcal{L}_{adv}, \mathcal{L}_{FM}$ denote the adversarial loss [18] and feature matching loss [28] with λ_{adv} and λ_{FM} controlling their relative importance. Second, since we hope the classifier to achieve high accuracy over the generated RGB images, we jointly fine-tune the trained GAN obtained from the first stage along with the classifier, using the following objective function: (θ is the parameters of the classifier.)

$$\min_{G,\theta} \left(\max_D \mathcal{L}_{gan} + \lambda_{cls} \mathcal{L}_{cls} \right) \tag{2}$$

where L_{cls} represents the classification loss of generated images by inpainting GAN. Note that in the first step we do not include classification loss to help GAN generate more realistic and diverse images, after which it would be easy to fine-tune the classifier jointly.

3.4. Rationale of Shape Features for Robustness

Here we will discuss the intuition for why shape structured features could help improve model robustness against different types attacks: (i) **adversarial attacks**, (ii) **common corruptions**, (iii) **backdoor attacks**. We provide the **rational** for leveraging robust edges features to improve the robustness as below. For *adversarial attacks*, EdgeGANRob is expected to improve the robustness as edges are invariant to small imperceptible adversarial perturbations. Intuitively, consider a ℓ_∞ threat model, it is very challenging for an attacker to make a specific edge pixel appear/disappear by reversing the magnitude of image gradient with only limited adversarial budget per pixel. When the test data is under *certain corruptions* well-preserved shape structure, leveraging edge features could be helpful to improve the model's generalization ability. EdgeGANRob would work in this case by focusing on shape structure which makes it less sensitive to changes in test data. Recall that in *backdoor attacks*, an attacker aims to poison the training data with a specific pattern such that the trained models can be tricked into predicting a certain class when the pattern is injected at testing time. Thus, extracting edges can be viewed as data sanitization to remove malicious patterns, thus rendering potential backdoor attacks ineffective.

4. Experimental Results

We evaluate the robustness of EdgeNetRob and EdgeGANRob under different attacks in this section. We also compare their performance against state of the art baselines to provide more intuition.

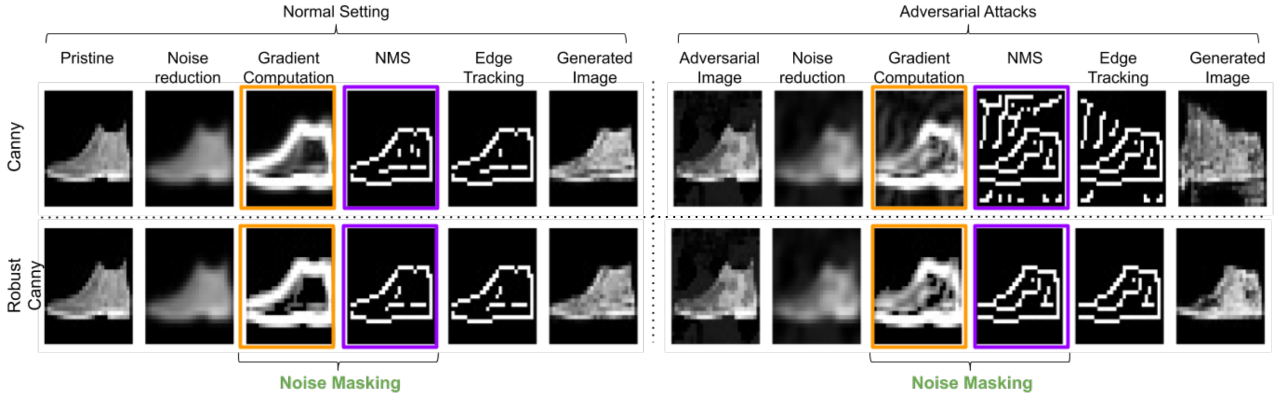

Figure 2: Visualization of intermediate stages of Vanilla Canny (*Upper*) and Robust Canny (*Lower*) on an image randomly sampled from Fashion MNIST. Results for clean images (*Left*) and adversarial images (*Right*) are presented.

4.1. Experimental setup

We conduct comprehensive experiments to evaluate the robustness of EdgeNetRob and EdgeGANRob under three tasks (adversarial attacks common corruptions and backdoor attacks) on two types of datasets: (a) dataset with clear and less ambiguous edge information (Fashion MNIST [63] and CelebA [37]) (Figure 3- left), and (b) dataset with less clear or complicated edge information (CIFAR-10 [30], Tiny-ImageNet and ImageNet [13]) (Figure 3-right). We do not choose the MNIST for our study since it is a toy dataset where strong robustness has been achieved [15, 39]. For ImageNet, we only evaluate the robustness of EdgeNetRob as the Edge to Image for ImageNet is still an active research problem. We leave it as our future work. More details including the network architecture and parameters of the experiments can be found in Appendix A.

4.2. Robustness against Adversarial Attacks

We evaluate our methods using the commonly used ℓ_∞ adversarial perturbation constrained on input range $[0, 1]$ [17, 39, 43, 50, 64]. We use standard perturbation budget on these datasets as in [43, 46, 50, 53, 57]: ℓ_∞ as 25/255 for Fashion MNIST; ℓ_∞ as 8/255 for CIFAR-10, CelebA and 4/255 Tiny ImageNet and ImageNet. We evaluate our methods in both whitebox and blackbox settings. For whitebox, we evaluate the robustness of edge feature against the strongest adaptive attack where the attacker has full knowledge about the defense algorithm. The attacker will attack the whole pipeline of EdgeNetRob (including edge extractor and classifier) and EdgeGANRob (including the edge extractor, generator and classifier). As Canny edge detector contains the non-differentiable operations, we measure its robustness against white-box attacks by using the BPDA attack [1]. This attack requires a differentiable version of Canny. Therefore, we approximate the non-differentiable operations and provide a differentiable version in Appendix C. Additionally, for blackbox attack, we select the gradient free attack, SPSA [55]. We refer the reader to Appendix B for more details on the attack settings.

Table 1: Comparison of different edge extraction methods when used for EdgeNetRob on Fashion MNIST

Method	Clean Accuracy	FGSM	PGD-10	PGD-40
RCF	**90.15**	50.07	3.37	0.18
Canny	88.32	66.98	54.07	39.99
Robust Canny	87.00	**79.03**	**78.53**	**76.75**

4.2.1 Robust Edge Detector

Here we first illustrate why a robust edge detector is needed against adversarial attacks. We compare the robustness of three edge detection methods: 1) RCF [36] which uses a CNN as backbone to generate edge maps; 2) Canny [3] which is a traditional edge detection algorithm; 3) the proposed robust edge detection algorithm, Robust Canny. To attack Canny edge detector, we apply the white-box BPDA attack [1]. We evaluate EdgeNetRob with different edge detectors, and the results on Fashion MNIST are reported in Table 1. First, we can see that using edges generated by RCF is not robust, as under strong adaptive attack the accuracy drops near to 0. This result is in accordance with Cosgrove and Yuille [12], where they show that there exist adversarial examples for neural network based edge detectors. Second, it can be noticed that adaptive attack (PGD-40) can reduce the accuracy of Canny based EdgeNetRob to 39.99% . This also verifies that our adaptive attack customized for Canny is strong and valid. We find that Robust Canny can significantly boost the robustness under strong adaptive attack: from 39.99% to 76.75%. This shows that the truncation of values in Robust Canny is effective in reducing the adversarial vulnerability. Thus, for the experi-

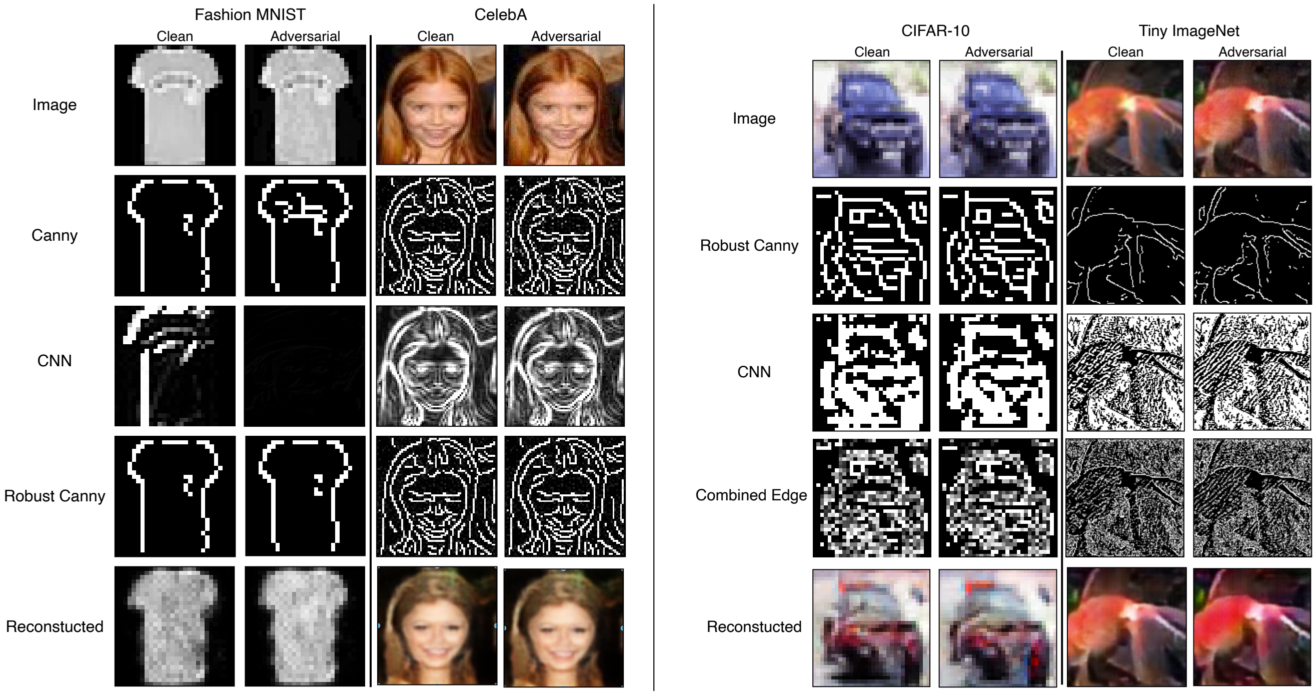

Figure 3: Edges and reconstructed images from four datasets. Row 2-4 represent edges extracted by Canny, CNN and Robust Canny respectively. For CIFAR-10 and Tiny-ImageNet, the second row shows results of robust Canny and the fourth row shows results of Combined edge. The last row presents the reconstructed images. Note that adversarial examples generated against different edge detectors are visually close, and thus we randomly select one of them to show in the first row.

Table 2: Evaluation of adversarial robustness on various datasets. The results of edge feature enabled pipelines are shown in grey.

Dataset	Method	Clean Accuracy	FGSM	PGD-10	PGD-40
Fashion MNIST	Vanilla Net	92.88	27.82	1.76	0.48
	PGD-training	86.99	78.99	74.79	72.62
	EdgeNetRob	87.00	**79.03**	**78.53**	**76.75**
	EdgeGANRob	**87.14**	78.67	76.82	72.69
CelebA	Vanilla Net	98.30	18.67	0.00	0.00
	PGD-training	92.75	84.67	82.55	81.31
	EdgeNetRob	94.51	87.97	84.36	82.81
	EdgeGANRob	**95.88**	**91.06**	**88.12**	**84.60**
CIFAR-10	Vanilla Net	91.89	6.76	0.00	0.00
	PGD-training	76.50	56.55	45.80	**44.15**
	EdgeNetRob	**79.21**	**63.49**	45.26	33.08
	EdgeGANRob	76.25	62.61	**46.72**	37.15
Tiny ImageNet	Vanilla Net	58.55	4.10	0.21	0.00
	PGD-training	48.10	30.11	23.65	**22.31**
	EdgeNetRob	**48.20**	**39.21**	**24.52**	19.53
	EdgeGANRob	44.30	36.22	23.15	13.55
ImageNet	Vanilla Net	76.40	6.50	0.00	0.00
	Fast [57]	55.45	39.62	30.48	**30.20**
	EdgeNetRob	**64.13**	**44.10**	**31.23**	22.73

ments later, we will use Robust Canny as the default edge extractor for EdgeNetRob and EdgeGANRob.

4.2.2 Comparison with Baselines

We compare EdgeNetRob and EdgeGANRob with the state-of-the-art baseline: PGD adversarial training proposed in [39] for Fashion MNIST, CIFAR, CelebA and Tiny ImageNet. For Imagenet, we select efficient PGD adversarial training variant : Fast [57] as the baseline. Note that, the main goal of this paper is to analyze how useful the edges features are in terms of improving robustness under different attacks, rather than demonstrating they are the most robust among all individual attack settings.

As shown in Figure 3, we randomly select images from these datasets and visualize the benign, adversarial images and their corresponding edge maps based on EdgeNetRob. We observe that the edge images drawn from Fashion-MNIST and CelebA have clear edge information for recognition while the edges drawn from CIFAR-10 and Tiny ImageNet are less clear and complicated which are hard to be recognized. Therefore, in the following part, we analyze the result by dividing the datasets into two categories: (1) dataset with clear edge information (Fashion-MNIST and CelebA) and dataset with less clear or complicated edge informations (CIFAR-10, Tiny-ImageNet and Imagenet)

Next, we show the qualitative and quantitative results on those two categories in Table 2. For Fashion MNIST and CelebA, we notice that EdgeNetRob and EdgeGAN-

Rob lead to a small drop in clean accuracy compared to the vanilla baseline model due the missing texture information. However, when compared with adversarial training , both EdgeNetRob and EdgeGANRob surprisingly achieve a higher clean accuracy. Moreover, we also observe that the clean accuracy of EdgeGANRob is higher than EdgeNetRob on Fashion MNIST and CelebA dataset given more texture information. It validates the necessity of adding GANs to close the accuracy gap resulted from directly training on binary edge images on dataset with *the clear edge information*. In terms of adversarial robustness, we observe that under strong adaptive attacks, EdgeNetRob and EdgeGANRob still remain comparable robustness compared with PGD adversarial training. Additionally, we also use the gradient free attack (SPSA) to evaluate the adversarial robustness in blackbox setting. The results are shown in the Table E of the appendix. We could observe that SPSA could successfully attack the vanilla model and both EdgeNetRob and EdgeGANRob could improve the robustness and are even higher than PGD adversarial training. It further illustrates that our careful designed adaptive attacks are stronger enough and shows the edge feature indeed achieves non-trivial robustness.

Table 3: Clean and Robust accuracy of different edge detection methods on CIFAR-10. The robust accuracy is calculated using PGD-40.

Edge Detector	Clean Accuracy	Robust Accuracy
Robust Canny ($e_{\text{robust canny}}$)	67.85	36.31
CNN (e_{cnn})	87.11	0.82
Combined Edge (e_{combined})	79.21	33.08

Figure 3 (left) show the edges of clean (benign) and adversarial examples among different edge detector (vanilla canny, cnn-based, robust canny) on Fashion MNIST and CelebA. We could observe that the edges between benign and adversarial images are different for the Canny and CNN-based edge detection algorithms. However, for the proposed robust canny algorithm, the edges are almost similar between benign and adversarial images. These visualization results also indicated the vulnerability of vanilla and CNN-based edge detectors.

All of these results validate that edge information is a type of promising feature which could improve the robustness on dataset with clear edge information.

For datasets with unclear or complicated edge information such as CIFAR-10, Tiny-ImageNet and ImageNet, we first see the qualitative results in Figure 3 (right). Note that as shown in previous paragraph that the edge images of vanilla Canny are vulnerable and similar to the robust Canny on clean image, here we do not visualize the edge of vanilla Canny. Compared with edge extracted by CNN, we could observe that there is less information for the edges extracted by Robust Canny. In Table 3, we evaluate EdgeNetRob by using the edge maps extracted by Robust Canny ($e_{\text{robust canny}}$). It achieves 67.85% clean accuracy and 36.31% robust accuracy under adaptive attack with PGD-40. This result shows that EdgeNetRob is able to achieve non-trivial robustness compared to vanilla network (0%) and further indicates the effectiveness of edge feature to adversarial robustness. We attribute the relatively low clean accuracy (67.85%) to the quality of edges extracted by Canny. Additionally, CNN-based edge features (e_{cnn}) could achieve 87.11% accuracy on clean image yet less robust (0.82% robust accuracy). The edge features extracted from robust canny fail to contain enough edge information while CNN-based edge features have higher quality yet less robust. It motivates us to combined them together. Therefore, we propose to use more fine-grained edge features, Combined Edge, to improve edge quality on the datasets with less clear or complicated edge information.

Specifically, we use a three-layer CNN to extract edges and concatenate them with the edges extracted by Robust Canny (More details of combined edge can be found in Appendix D). In this setting, we apply EdgeNetRob on the Combine Edge feature and observe significant improvement on the clean accuracy as 79.21% while still preserve the non-trivial robust accuracy as 33.08%. We also apply this Combined Edge feature to EdgeGANRob and other dataset with less clear or complicated edge informations (Tiny-ImageNet and ImageNet). The final results are shown in Table 2. We could observe that even on the dataset with less clear or complicated edge information, edge features could still be used to achieve nontrivial robustness against adversarial attacks. The result of SPSA is shown in the appendix. It also indicates that edge features are able to achieve non-trivial robustness.

4.3. Robustness under Common Corruptions

We evaluate EdgeNetRob and EdgeGANRob under common corruptions [22]. For common corruptions, we test the models under 15 types of algorithmically generated corruptions used in [22]. Note that as the main goal of this paper is to analyze how useful the edges features to simultaneously improve robustness under different attacks, rather than they are the most robust among all individual attack settings, we do not compare our results with the STOA augmentation-based method (e.g Augmix [23] and DeepAugment [24]). The results are shown in Table 5. We can see that EdgeNetRob and EdgeGANRob are able to increase the robustness against common corruptions compared with the vanilla model on most datasets except Tiny-ImageNet. We address the potential reason for lower corruption robustness is due to the low clean accuracy of Tiny-ImageNet. Overall, the results suggest that robust edge feature is able to help boost robustness against common corruptions compared to vanilla network.

Table 4: Results of EdgeNetRob and EdgeGANRob against backdoor attacks on four datasets. (ASR: attack success rate – lower value indicates higher robustness) The results of edge feature enabled pipelines are shown in grey

Dataset	Ratio	Backdoor Pattern	Method							
			Vanilla Net		*Spectral Signature*		EdgeNetRob		EdgeGANRob	
			Clean Acc	Pois ASR	Clean Acc	Pois ASR	Clean Acc	Pois ASR	Clean Acc	Pois ASR
Fashion MNIST	20%	Pixel	87.43	94.30	86.23	45.62	83.48	0.12	**88.89**	**0.07**
		Pattern	87.12	95.22	85.93	52.31	82.21	2.74	**88.62**	**1.28**
CelebA	5%	Pixel	98.3	97.2	**98.45**	64.78	92.8	10.9	94.42	**4.59**
		Pattern	98.0	97.4	**97.98**	54.23	93.1	12.5	95.49	**3.01**
CIFAR-10	5%	Pixel	91.87	99.50	**91.69**	9.83	78.27	9.42	76.12	**3.55**
		Pattern	91.85	99.15	**91.60**	9.56	77.45	10.31	75.43	**3.13**
Tiny-ImageNet	10%	Pixel	55.25	75.91	**55.74**	28.90	43.48	27.26	41.59	**25.88**
		Pattern	55.05	95.93	**56.25**	67.55	42.72	55.54	41.03	**43.72**

Table 5: Performance of EdgeNetRob and EdgeGANRob against common corruptions. The results of edge feature enabled pipelines are shown in grey

Dataset	Method	Clean Acc.	Common Corruptions Acc.
Fashion MNIST	Vanilla Net	92.88	61.62
	EdgeNetRob	87.00	61.00
	EdgeGANRob	87.14	**63.14**
CelebA	Vanilla Net	98.30	64.96
	EdgeNetRob	94.51	68.27
	EdgeGANRob	95.88	**69.27**
CIFAR-10	Vanilla Net	91.89	67.15
	EdgeNetRob	79.21	**71.73**
	EdgeGANRob	76.25	68.64
Tiny ImageNet	Vanilla Net	58.52	**27.43**
	EdgeNetRob	48.20	22.60
	EdgeGANRob	44.30	21.09
ImageNet	Vanilla Net	76.40	23.30
	EdgeNetRob	64.13	**25.10**

4.4. Robustness against backdoor attacks

Here we evaluate the robustness of edge feature against backdoor attacks. We use the same backdoor patterns: Pixel and Pattern as in Tran et al. [54]. Sampled backdoored images and their edges are shown in Figure A. We randomly choose two source and target class pairs and report their average performance. Similar to Tran et al. [54], we select poisoning ratio as 20% for Fashion MNIST, 5% for CelebA and CIFAR-10 and 10% for Tiny ImageNet . We compare our method with the vanilla network and the defense method *Spectral Signature* [54].

The results are presented in Table 4, where we show the test accuracy on standard test data ('Clean Acc') and the attack success rate on poisoned data ('Pois ASR' – lower indicates higher robsutness). We observe that our embedding pattern can successfully attack the vanilla Net with high poisoning attack success rate on all datasets. It can be seen that *Spectral Signature* can not always achieve high performance on Fashion MNIST and CelebA. However, both EdgeNetRob and EdgeGANRob consistently obtain low poisoning attack success rate among different settings, and EdgeGANRob achieves the lowest poisoning attack success rate. Figure A in the Appendix shows the qualitative results of the backdoored images after edge detection and the reconstructed images. We can observe that the effect of backdoor patterns can be partially removed by the edge detector.

5. Conclusion

We introduced edge feature enable pipelines together with a proposed robust feature extractor to evaluate the model robustness improvement under different attacks. It shows that the edge enabled pipelines can boost the robustness under different settings including adversarial attacks, common corruptions and backdoor attacks simultaneously with minor clean accuracy decrease. Our results highlight the importance of using shape structural information in improving model robustness and we believe it will inspire promising directions for future work.

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and LLNL LDRD Program Project No. 20-ER-014 (LLNL-CONF-825477). This work was partially supported by the NSF grant No.1910100, NSF CNS 20-46726 CAR, NSF CNS-2012001, IIS-2040800, the ARO under contract W911NF1810208, Amazon Research Award and grant from Open Philanthropy and Good Ventures Foundation. The authors thank the anonymous reviewers and ACo'x for helpful comments and thank Xinchen Yan, Jun-Yan Zhu, Zhiding Yu, De-An Huang and Zhaoheng Zheng for helpful discussions.

References

[1] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *ICML*, 2018. 2, 5, 11

[2] N. Baker, H. Lu, G. Erlikhman, and P. J. Kellman. Deep convolutional networks do not classify based on global object shape. *PLoS computational biology*, 14(12):e1006613, 2018. 1, 3

[3] J. Canny. A computational approach to edge detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986. 2, 3, 5

[4] Y. Cao, C. Xiao, B. Cyr, Y. Zhou, W. Park, S. Rampazzi, Q. A. Chen, K. Fu, and Z. M. Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 2267–2281, 2019. 1

[5] N. Carlini and D. A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 39–57, 2017. doi: 10.1109/SP.2017.49. URL https://doi.org/10.1109/SP.2017.49. 1

[6] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, A. Madry, and A. Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019. 2

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 1

[8] X. Chen, C. Liu, B. Li, K. Lu, and D. Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017. 2

[9] X. Chen, C. Liu, B. Li, K. Lu, and D. Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017. 1

[10] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi. Describing textures in the wild. *CVPR*, 2013. 12

[11] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008. 1

[12] C. Cosgrove and A. L. Yuille. Adversarial examples for edge detection: They exist, and they transfer. *arXiv preprint arXiv:1906.00335*, 2019. 3, 5

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[14] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. L. Seltzer, G. Zweig, X. He, J. D. Williams, et al. Recent advances in deep learning for speech research at microsoft. In *ICASSP*, volume 26, page 64, 2013. 1

[15] G. W. Ding, K. Y. C. Lui, X. Jin, L. Wang, and R. Huang. On the sensitivity of adversarial robustness to input data distributions. In *ICLR*, 2019. 5

[16] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *ICLR*, 2019. 1, 2, 3

[17] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org. 1, 5

[18] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 4

[19] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2, 11

[20] T. Gu, B. Dolan-Gavitt, and S. Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017. 2

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 11

[22] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019. 1, 2, 7

[23] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019. 7

[24] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020. 7

[25] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In *NIPS*, 2019. 1, 3

[26] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 4

[27] T. Itazuri, Y. Fukuhara, H. Kataoka, and S. Morishima. What do adversarially robust models look at? *arXiv preprint arXiv:1905.07666*, 2019. 3

[28] J. Johnson, A. Alahi, and F.-F. Li. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 4

[29] N. Kanopoulos, N. Vasanthavada, and R. L. Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2):358–367, 1988. 4

[30] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. 2009. 5

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[32] B. Landau, L. B. Smith, and S. S. Jones. The importance of shape in early lexical learning. *Cognitive development*, 3(3): 299–321, 1988. 2

[33] B. Li, Y. Wang, A. Singh, and Y. Vorobeychik. Data poisoning attacks on factorization-based collaborative filtering. In *NIPS*, 2016. 1

[34] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *CVPR*, 2018. 2

[35] K. Liu, B. Dolan-Gavitt, and S. Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks.

arXiv preprint arXiv:1805.12185, 2018. 3

[36] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai. Richer convolutional features for edge detection. In *CVPR*, 2017. 2, 5

[37] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 5

[38] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[39] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1, 2, 3, 5, 6, 11

[40] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2574–2582, 2016. 1

[41] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. The limitations of deep learning in adversarial settings. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 372–387. IEEE, 2016.

[42] H. Qiu, C. Xiao, L. Yang, X. Yan, H. Lee, and B. Li. Semanticadv: Generating adversarial examples via attribute-conditioned image editing. In *European Conference on Computer Vision*, pages 19–37. Springer, 2020. 1

[43] P. Samangouei, M. Kabkab, and R. Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. In *IClR*, 2018. 5

[44] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 1

[45] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and G. Tom. Poison frogs! targeted clean-label poisoning attacks on neural networks. *arXiv preprint arXiv:1804.00792*, 2018. 3

[46] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! *arXiv preprint arXiv:1904.12843*, 2019. 5

[47] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. P. Dickerson, C. Studer, L. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! In *NeurIPS*, 2019. 1

[48] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529 (7587):484, 2016. 1

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[50] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018. 2, 5

[51] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[52] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, et al. Going deeper with convolutions. In *CVPR*, 2015. 1

[53] R. Theagarajan, M. Chen, B. Bhanu, and J. Zhang. Shieldnets: Defending against adversarial attacks using probabilistic adversarial robustness. In *CVPR*, 2019. 5

[54] B. Tran, J. Li, and A. Madry. Spectral signatures in backdoor attacks. In *NIPS*, 2018. 1, 3, 8

[55] J. Uesato, D. B. O', A. van den Oord, and P. Kohli. Adversarial risk and the dangers of evaluating against weak attacks. In *ICML*, 2018. 5, 11

[56] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 4

[57] E. Wong, L. Rice, and J. Z. Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020. 5, 6

[58] C. Xiao, R. Deng, B. Li, F. Yu, D. Song, et al. Characterizing adversarial examples based on spatial consistency information for semantic segmentation. In *Proceedings of the (ECCV)*, pages 217–234, 2018. 1

[59] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song. Generating adversarial examples with adversarial networks. In *IJCAI*, 2018.

[60] C. Xiao, J.-Y. Zhu, B. Li, W. He, M. Liu, and D. Song. Spatially transformed adversarial examples. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=HyydRMZC-`.

[61] C. Xiao, R. Deng, B. Li, T. Lee, B. Edwards, J. Yi, D. Song, M. Liu, and I. Molloy. Advit: Adversarial frames identifier based on temporal consistency in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3968–3977, 2019.

[62] C. Xiao, D. Yang, B. Li, J. Deng, and M. Liu. Meshadv: Adversarial meshes for visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6898–6907, 2019. 1

[63] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 5

[64] C. Xie, Y. Wu, L. v. d. Maaten, A. Yuille, and K. He. Feature denoising for improving adversarial robustness. In *CVPR*, 2019. 5

[65] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, 2015. 2, 3

[66] L. Yang, Q. Huang, H. Huang, L. Xu, and D. Lin. Learn to propagate reliably on noisy affinity graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1

[67] F. Yu, V. Koltun, and T. Funkhouser. Dilated residual networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[68] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan. Theoretically principled trade-off between robustness and accuracy. *arXiv preprint arXiv:1901.08573*, 2019. 3

[69] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei. Deep feature flow for video recognition. In *CVPR*, volume 1, page 3, 2017. 1