Robust Superpixel-Guided Attentional Adversarial Attack

Xiaoyi Dong[1], Jiangfan Han[2], Dongdong Chen[3]*, Jiayang Liu[1] , Huanyu Bian[1], Zehua Ma[1],
Hongsheng Li[2], Xiaogang Wang[2], Weiming Zhang[1], Nenghai Yu[1],
[1]University of Science and Technology of China
[2]The Chinese University of Hong Kong, [3]Microsoft Cloud AI
{dlight@, ljyljy@, hybian@, mzh045@, }mail.ustc.edu.cn
{jiangfanhan@link., hsli@ee., xgwang@ee.}cuhk.edu.hk
{zhangwm@, ynh@}ustc.edu.cn, cddlyf@gmail.com

Abstract

Deep Neural Networks are vulnerable to adversarial samples, which can fool classifiers by adding small perturbations onto the original image. Since the pioneering optimization-based adversarial attack method, many following methods have been proposed in the past several years. However most of these methods add perturbations in a "pixel-wise" and "global" way. Firstly, because of the contradiction between the local smoothness of natural images and the noisy property of these adversarial perturbations, this "pixel-wise" way makes these methods not robust to image processing based defense methods and steganalysis based detection methods. Secondly, we find adding perturbations to the background is less useful than to the salient object, thus the "global" way is also not optimal. Based on these two considerations, we propose the first robust superpixel-guided attentional adversarial attack method. Specifically, the adversarial perturbations are only added to the salient regions and guaranteed to be the same within each superpixel. Through extensive experiments, we demonstrate our method can preserve the attack ability even in this highly constrained modification space. More importantly, compared to existing methods, it is significantly more robust to image processing based defense and steganalysis based detection.

1. Introduction

Deep neural networks(DNNs) have achieved great achievements on many artificial intelligence tasks such as image recognition [15, 19], object detection [33] and natural language processing [40]. But recent works [37] found that DNNs are vulnerable to adversarial samples. Adversarial samples are carefully crafted images by making small and invisible perturbations on the original images. Although they are indistinguishable to the original images by human being, they can make DNNs make totally wrong predictions. Adversarial samples reveal the defect and sensitivity of DNNs, but can also help us to get better understanding of DNNs and train more robust models.

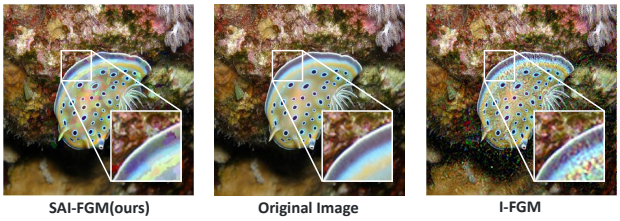

Figure 1. Original image and corresponding adversarial samples generated by SAI-FGM(proposed method) and I-FGM. Meanwhile, we enlarge part of the images to show the difference of smoothness between the three images.

First proposed by [37], methods to generate adversarial sample are various. Among all of them, gradient-based methods [9, 17] which calculate the modification pattern by taking the gradient with respect to the input sample are simple and effective. Fast Gradient Sign Method(FGSM) proposed in [9] using the gradient of classification loss with respect to the input image as the adversarial noise to fool the recognition models. It provides a quick solution to get adversarial samples. Then [17] proposes iterative version of FGSM(I-FGSM) to get better attack performance. [5] further adds a momentum term to improve the transferability of adversarial sample to unseen models. Other works [41, 6] provide various methods to get more robust adversarial sample to unknown even defensive models.

Notwithstanding their success, all of these methods add adversarial perturbations in a *"pixel-wise"* and *"global"* way. Here *"pixel-wise"* means these adversarial perturbations are added onto each pixel independently, thus very noisy in most cases. By contrast, from the statistic perspec-

*Dongdong Chen is the corresponding author.

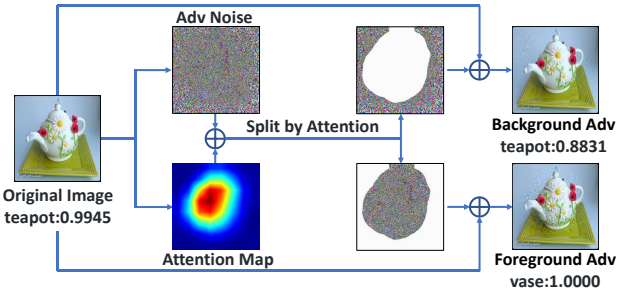

Figure 2. Pipeline of generating background and foreground adversarial samples. Background and foreground noise are generated by splitting the adversarial noise to two equal size parts with the attention map.

tive, natural images often have the local smoothness property. As illustrated in Figure 1, comparing with original image, adversarial samples generated by I-FGSM are much noisy and rough. Such contradiction makes these methods not robust to existing image processing based defense methods and steganalysis based detection methods. For image processing based defense methods, they often process the adversarial samples in a local way (e.g., smoothing, resizing) and destroy the original noisy adversarial pattern. Similarly, the reason why steganalysis based detection methods can detect adversarial samples is also because steganalysis features can detect the small adversarial perturbations that do not follow the local smoothness property.

For *"global"*, it means that most existing methods treat all the pixels in one image equally and add perturbations to all pixels. In this paper, we argue this *"global"* way is not optimal and has two obvious drawbacks. Firstly, as shown in Figure 2, we split the adversarial noise to two equal size parts as the foreground object part and the background part and we find that adding adversarial noise to foreground object is more useful than to the background. Because the target classifier only focuses on the foreground part, which is demonstrated by its activation map. Secondly, foreground objects often contain more textures than background regions (e.g., sky, lake) statistically, so perturbations in the background regions are also much easier to be detected.

Based on the above two considerations, we propose the first robust superpixel-guided attentional adversarial attack method. Specifically, given an input image, we first leverage traditional superpixel generation method [32] to get the over-segmented superpixel map, where pixels in each superpixel have similar colors and follow the local smoothness property. Then we generate the adversarial perturbations in a superpixel manner, i.e., the perturbation within each superpixel must be the same. To further ensure the local smoothness, we improve the original superpixel method [32] by adaptively merging similar superpixels. For the above *"global"* problem, we replace it with an "attentional" way by using the auxiliary information from the class activation map [43]. In other words, we constrain that the perturbations are only added onto the foreground object (i.e., the class activated regions) rather than the whole image.

With the above locally smooth and attentional constraint, though the adversarial perturbation space of our method is much smaller than previous "pixel-wise and global" methods, extensive experiments demonstrate our method can still preserve the attack ability with dedicated designs. More importantly, we show that the proposed superpixel-guided attentional adversarial attack is significantly more robust to image processing based defense and steganalysis based detection methods by a large margin.

To summarize, our contributions are threefold as below.

- We clearly analyze the limitations of existing "pixel-wise and global" adversarial attack methods, and clarify the underlying reason of why they are not robust to image processing and steganalysis.
- Based on the analysis, we propose the first superpixel-guided attentional adversarial attack method, which not only guarantees the local smoothness but also modifies the image in a more effective way.
- Extensive experiments demonstrate the proposed method can preserve the original attack ability and achieve superior robustness simultaneously.

2. Related Work

Adversarial Attack. Current adversarial sample methods can be categorized into three types: optimization-based [37, 2], gradient-based [9, 17], and generation-based [1, 31, 14]. Optimization-based methods model the generation of adversarial samples as an optimization problem and use optimizers like box-constrained L-BGFS or Adam to solve it, which are powerful but quite slow. Goodfellow /etal [9] proposed the first gradients-based method Fast Gradient Sign Method (FGSM) and following work [17] performed small step size iteratively to get better attack performance (I-FGSM). Generation-based methods aim to train a model to generate adversarial samples with a single forward path, which is very fast but requires extra training time. Different from their "pixel-wise and global" way, the proposed method is the first super-pixel level and foreground only adversarial sample generation method.

Adversarial Detection. There are roughly four different types of adversarial detection methods: model-based [11, 8], PCA-based [16, 20], preprocessing-based [22, 42] and steganalysis-based [24]. Model-based methods view adversarial samples as an additional category and retrain the network to classify adversarial samples to this new category. PCA-based methods leverage the fact that adversarial samples place a higher weight on the larger principal components compared with clean images. Preprocessing-based

methods apply some image transformation on the input images and check the prediction variance. Recently, motivated by steganalysis, [24] proposes to use steganalysis features to detect adversarial samples, because these features are sensitive to the change of image statistics and small perturbations such as C&W [2]. Our idea is partially inspired by this method, and this method is also used as one important metric to evaluate the robustness of adversarial samples.

Adversarial Defense. Compared with adversarial detection, adversarial defense is another active way to protect the target model against adversarial attack. Adversarial training proposed by [9] is one of the most popular way to train a robust network by adding the adversarial sample into the training set. In [12, 28, 3, 23, 44], they add one preprocessing step to remove the adversarial noise of the input images before feeding into the target model. For example, method [25] proposes DNN-Oriented JPEG compression with the idea of feature distillation to remove adversarial noise. Another work [13] proposes to use JPEG compression, total variance minimization(TVM) and image quilting to defence adversarial attack. Another type of methods like [38, 29, 30] use some regularizers or smooth labels to make the target model more robust to the perturbation on input images. Among the aforementioned methods, image preprocessing is the most lightweight method as it does not need to change model architecture or parameters. In this paper, we also use this type of methods to evaluate the final adversarial robustness.

Superpixels. In the era before deep learning, superpixel segmentation [32, 27] aims to oversegment the image by grouping pixels that share similar properties. It captures the redundancy of image and is regarded as representative primitive for computing image features. They are widely used in many computer vision algorithms, such as image compression [34], depth estimation [26], and stereo matching [10]. Motivated by the smooth property within each superpixel, we are the first who leverages superpixel as the guidance to add adversarial perturbations. This helps us to generate local smooth and robust adversarial samples.

3. Method

3.1. Problem Definition.

We denote $\mathbf{x}$ as the source image and y as the corresponding ground-truth label. Let $\mathcal{H}$ be the target model with parameters θ. Then $\mathcal{H}(\mathbf{x};\theta)$ is the probability prediction for each class. For an ideal model we should get $\arg\max_c \mathcal{H}(\mathbf{x};\theta)_c = y$. For a real model, the equation is also satisfied for most samples. An adversarial sample $\mathbf{x}^{adv} = \mathbf{x} + \mathbf{r}$ is generated by adding noise $\mathbf{r}$ to the original image $\mathbf{x}$ and satisfies $\arg\max_c \mathcal{H}(\mathbf{x}^{adv};\theta)_c \neq y$. In the mean time, the noise $\mathbf{r}$ should be small enough to guarantee the adversarial sample is similar to the original one. In most cases $\mathbf{r}$ is measured by l_p norm and the constraint $\|\mathbf{r}\|_p \leqslant \epsilon$ is proposed to fulfill the similarity requirement. Here ϵ is a predefined threshold constant.

White-Box *vs*. Gray-Box *vs*. Black-Box. If we have the full knowledge about the target model to attack, including the model architecture and parameters, such an attack is called as white-box attack. In this case, we can generate adversarial samples with back-propagated gradients directly. If we have the full access to the model, but there are some unknown input transformations before feeding images into the model, we call such an attack as gray-box attack. For black-box attack, we have no knowledge about the target model, so adversarial samples are often generated by other white-box models and fool the target model with transferability.

3.2. Motivation

As briefly described in the introduction part, our method is motivated from the below two observations:

1). The contradiction between the local smoothness of natural images and the noisy property of adversarial perturbations. Although previous attack methods can generate strong adversarial samples, most of them add adversarial perturbations in a "pixel-wise" way and do not consider the original neighborhood information. This causes the generated adversarial perturbations to be noisy as shown in the Figure 1(Right). However, pixels in natural images often have the local smooth property, which implies neighboring pixels often have similar pixel values. This is also the theoretical building block for many image processing tasks like image compression. Therefore, adding pixel-wise perturbations will destroy the original local statistics such as first-order and second-order Markov process statistics. This also motivates the method [24] to use steganalysis features to detect adversarial samples as steganalysis features considers such types of local statistics. Besides, this contradiction also makes these adversarial attack methods not robust to image processing based defense methods. This is because processing techniques like resizing or smoothing are based on the local smooth property and will destroy the original noisy pattern of adversarial perturbations. Take I-FGSM as an example, if we use the local averge smoothing with 3×3 kernel as the processing step before feeding the white-box model, the attack success rate will decrease from 99.40% to 73.67%. Hence, the local smooth property should be considered when adding the adversarial perturbations.

2). It is more effective to add adversarial perturbations to salient object the classifier focuses on. Another observation is that most existing adversarial sample methods treat all the pixels equally and add perturbations to the image globally, but we argue this global way is not optimal. To verify it, we conduct one simple experiment that only keeps the perturbations of background regions and the foreground

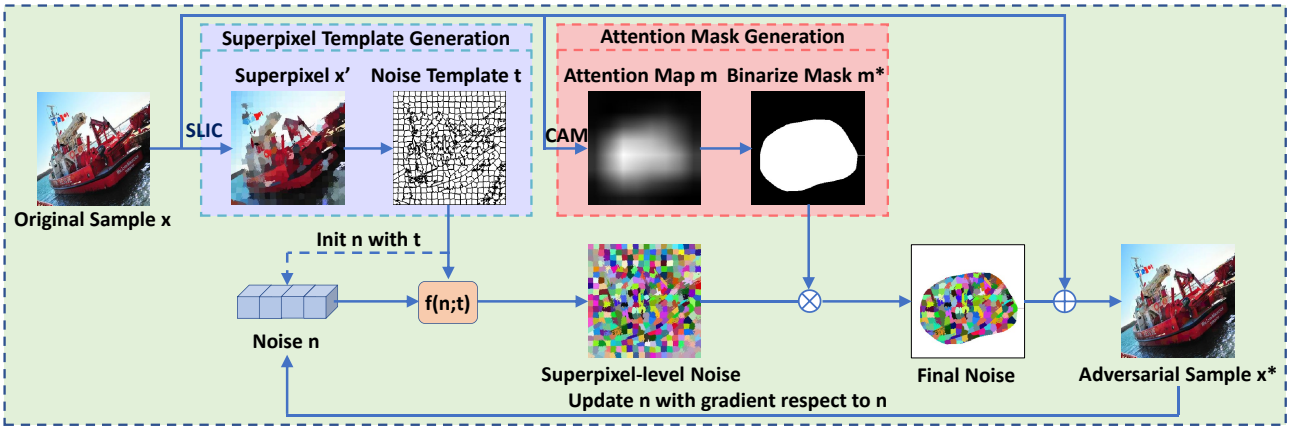

Figure 3. Pipeline of Generating Superpixel Adversarial Samples. We first use SLIC [32] to generate superpixel image $\mathbf{x}'$ from $\mathbf{x}$ as the noise template $\mathbf{t}$. Then we calculate the noise vector $\mathbf{n}$ and use a mapping function $f(\mathbf{n};\mathbf{t})$ to map it to superpixel-level noise with the template $\mathbf{t}$. Then we crop it with the binarized attention mask $\mathbf{m}^*$ and scale it to the threshold ϵ to get the final adversarial noise.

regions respectively. For I-FGM($\delta = 2$), the attack success rates of foreground/background regions adversarial noise are 94.03% and 46.35% respectively. This means adding perturbations to the foreground objects is more effective and can get better attack performance under the global distance measurement. In fact, this phenomenon is also consistent with the attention map the classifier focuses.

Another intuition is that foreground regions often have more textures than the background regions (e.g., sky, ocean) statistically, which implies hiding perturbations in the foreground regions is much easier. To support this hypothesis, we randomly select 2000 images from the ImageNet [4] dataset and use the smoothness metric HILL [18] to measure the texture richness. The average HILL of foreground regions and background regions are 5269.94 and 18853.42 respectively, here larger HILL indicates less textures.

3.3. Superpixel-Guided Attentional Adv Attack

Inspired by the above two observations, we propose the first superpixel-guided attentional adversarial attack method. Rather than adding pixel-wise adversarial perturbations, we use the oversegemented superpixels as guidance and constrain the perturbations within each superpixel should be same. In this way, the generated perturbations are also locally smooth in the superpixel level. Another guidance we use is the foreground attention map, which ensures the perturbations are only added to the foreground objects.

Figure 3 is the overall pipeline of our method, which consists of three steps: 1) Generate the superpixel-guided noise template $\mathbf{t}$ and initialize the adversarial noise $\mathbf{n}$ whose length equals to the superpixel number. 2) Crop and enhance the adversarial perturbations based on the attention map. 3) Use the generated adversarial sample to update $\mathbf{n}$ iteratively.

Template Generation. Instead of generating the template by sampling pixels with grid steps, we use superpixel algorithms to get the segmentation map as the modification template. It can help to ensure local smoothness and reduce the statistical difference between the source image and the adversarial sample. In this paper, the superpixel algorithm SLIC [32] is used by default. Specifically, both the perceptual color distance and spatial distance are considered as the distance measure:

$$\begin{aligned} dis &= d_c + \frac{p}{S} d_s \\ d_c &= \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \\ d_s &= \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \end{aligned} \tag{1}$$

where dis is the distance between the i_{th} pixel and the k_{th} superpixel cluster. Denote N, K as the total pixel number and the pre-defined superpixel number, then each superpixel cluster C_k is represented as the tuple $(l_k, a_k, b_k, x_k, y_k)$. Here l_k, a_k, b_k is the pixel values of C_k in the CIELAB color space and x_k, y_k is the spatial coordinate. $S = \sqrt{N/K}$ is the grid interval, and p is the parameter to control the compactness of a superpixel. By sampling pixels at regular grid steps S as initialization, SILC updates the clustering result by using a linear iterative clustering algorithm to cluster every pixel to it's proximate super-pixel center.

One weakness of SLIC is that the clustering number K is fixed, which means a large smooth region will be clustered into many different small superpixels. If we use it as the template of adversarial noise, different perturbations will be added to each small superpixel. This will influence both the visual quality and statistics of the adversarial samples. To overcome this shortcoming, we propose an adaptive combination strategy to combine neighboring superpixels with similar pixel values. If the color difference between two neighboring superpixels is smaller than β, we combine them as a new superpixel.

Attention Mask. In this paper, we use Class Activation Mapping (CAM) [43] method to generate the attention map of the input image. A class activation map for a particular category indicates the discriminative image regions used by CNN to identify that category. CAM calculates this map by simply projecting back the output layer on to the convolutional feature maps. However, all pixel values in the attention map generated by CAM [43] are between 0 to 1, so it cannot be used to guide the noise cropping directly. We use a binarization factor ϕ to transform the attention map to a binarized map. The binarization can be expressed as

$$m^*_{i,j} = \begin{cases} 0 & m_{i,j} < \phi \\ 1 & m_{i,j} \geqslant \phi \end{cases} \tag{2}$$

where $\mathbf{m}$ is the attention map and $\mathbf{m}^*$ is the binarized attention map, $m_{i,j}$ and $m^*_{i,j}$ are the value of them at the position i, j. With the binarized attention map $\mathbf{m}^*$ as the mask, we crop the adversarial noise to realize attentional attack.

Adversarial Perturbation Generation. To describe how to generate the target adversarial perturbation, we first use I-FGM [17] as the baseline and propose the Superpixel-guided attentional version I-FGM called **SAI-FGM** by using the superpixel and attention map guidance.

For the baseline I-FGM, adversarial noises are generated by calculating the gradient of the loss function with the input image. However, we find calculating gradients for all superpixel is non-trivial. If we simply use the average or max gradient of all pixels within a superpixel as its gradient, it is neither efficient nor effective, which will be verified in the ablation part. Inspired by optimization based methods, we propose a substitute method to generate adversarial noise. We initialize a noise vector $\mathbf{n}$ whose length equals to the superpixel number, and calculate the gradient with respect to $\mathbf{n}$ directly. During each iteration, one mapping function f is first used to fill $\mathbf{n}$ into the superpixel level noise template $\mathbf{t}$ to get filled noise $f(\mathbf{n}; \mathbf{t})$. Then $f(\mathbf{n}; \mathbf{t})$ is cropped and scaled to the threshold ϵ before adding to the original sample $\mathbf{x}$. At each iteration step $i+1$, $\mathbf{n}$ will be updated from $\mathbf{n_i}$ to $\mathbf{n_{i+1}}$ with gradient asscent. Formally, our SAI-FGM can be expressed as

$$\begin{aligned} \mathbf{x_0^{adv}} &= \mathbf{x} + f(\mathbf{n_0}; \mathbf{t}) \\ \mathbf{x_i^{adv}} &= \mathbf{x} + Scale_\epsilon\{Crop_{\mathbf{m}}(f(\mathbf{n_i}; \mathbf{t}))\} \end{aligned} \tag{3}$$

$$\mathbf{n_{i+1}} = \mathbf{n_i} + \alpha \cdot \frac{\nabla_{\mathbf{n}} L(\mathbf{x_i^{adv}}, y, \theta)}{\|\nabla_{\mathbf{n}} L(\mathbf{x_i^{adv}}, y, \theta)\|_2} \tag{4}$$

where $L(\mathbf{x_i^{adv}}, y, \theta)$ is the loss function and $\nabla_{\mathbf{n}} L(\mathbf{x_i^{adv}}, y, \theta)$ is the gradient of $L(\mathbf{x_i^{adv}}, y, \theta)$ with respect to $\mathbf{n_i}$. $Crop$ and $Scale$ represent the crop operation based on the attention map $\mathbf{m}$ and scale operation based on the perturbation scale factor ϵ respectively.

Attack	Inc-v3*	Inc-v4	IncRes-v2	Inc-v3$_{adv}$
FGM	**84.00**	54.75	**56.75**	56.05
SA-FGM(**Ours**)	73.80	**56.20**	51.55	**56.75**
I-FGM	99.80	40.65	38.00	30.70
SAI-FGM(**Ours**)	**100.00**	**68.35**	**64.95**	**65.05**
MI-FGM	99.90	66.45	**67.95**	62.40
M-SAI-FGM(**Ours**)	**99.95**	**68.50**	66.10	**67.45**

Table 1. The attack success rate (%) of adversarial attack on the ImageNet [4] dataset. * indicates the white-box attacks.

M-SAI-FGM. Our method is general to most existing gradient-based methods. So we can extend our method with other attack methods to get more powerful attack ability. For example, when combining our method with MI-FGSM [5] by integrating the momentum term, we get Momentum Superpixel-guided Attentional Version I-FGM called **M-SAI-FGM**. Comparing with SAI-FGM, we update $\mathbf{n_i}$ by replacing Eq.4 with:

$$\begin{aligned} \mathbf{g_{i+1}} &= \mu \cdot \mathbf{g_i} + \frac{\nabla_{\mathbf{n}} L(\mathbf{x_i^{adv}}, y, \theta)}{\|\nabla_{\mathbf{n}} \mathbf{L}(\mathbf{x_i^{adv}}, y, \theta)\|_1} \\ \mathbf{n_{i+1}} &= \mathbf{n_i} + \alpha \cdot \frac{\mathbf{g_{i+1}}}{\|\mathbf{g_{i+1}}\|_2} \end{aligned} \tag{5}$$

4. Experiments

As introduced above, the adversarial perturbations of our method have two key constraints: superpixel level and only added to attentional regions, so our perturbation space is much smaller. Considering such a constrained space, the first question to answer is that *"whether the original attack ability can be preserved or not"*. After answering this question, another key question would be *"whether the proposed method can boost the adversarial robustness"*.

Therefore in this section, the proposed method is evaluated from two aspects: ***attack ability*** and ***attack robustness***. In the following experiments, we use L_2 norm as the distance measurement, threshold of the L_2 norm ϵ is calculated by $\delta\sqrt{N}$, where $N = C \times H \times W$ is the dimension of input images $\mathbf{x}$. Inception-V3 [36] is adopted as the default target attack model. Unless specified, $\phi = 4/12, \alpha = \delta\sqrt{N}/T$ and the total iteration number $T = 10$.

4.1. Attack Ability Comparison

To evaluate the attack ability, we compare the basic attack performance on the ImageNet [4] dataset, including white-box attack success rate and black-box attack success rate. As shown in Table 1, we compare three baselines FGM [9], I-FGM [17], MI-FGM [5] with our corresponding superpixel-guided attentional version SA-FGM, SAI-FGM, M-SAI-FGM. The column "Inc-v3" is white-box attack results while another three "Inc-v4", "IncRes-v2", "Inc-v3$_{adv}$" represent black-box attack results on model Inception-v4 [35], Inception Resnet-v2 [35], Inception-v3$_{adv}$ respectively. Here Inc-v3$_{adv}$ is Inception-v3 [36]

Generation method	$\delta = 2$	$\delta = 4$	$\delta = 6$	$\delta = 8$
FGM	94.32	95.59	96.28	97.09
I-FGM	94.11	94.74	95.45	96.01
SAI-FGM($\beta = 0$)	75.50	82.10	85.00	87.20
SAI-FGM($\beta = 2$)	72.40	78.20	82.00	84.20
SAI-FGM($\beta = 4$)	**63.30**	**72.10**	**76.40**	**80.90**

Table 2. Detect rate (%) of steganalysis-based detection. Lower success rate indicate the adversarial samples are more robust.

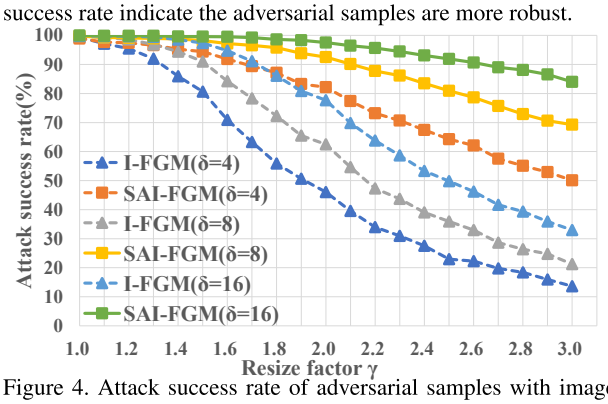

Figure 4. Attack success rate of adversarial samples with image resizing. Resize factor γ varies from 1.0 to 3.0.

model with ensemble adversarial training [38]. Here we set $\delta = 16$.

Result shows that the single step attack method FGM performs better than our SA-FGM , this means the influence of constrained perturbation space is inevitable. However, if we use multiple step attack method I-FGM or MI-FGM, our SAI-FGM and M-SAI-FGM can achieve similar white-box attack success rate (nearly 100%). This implies that our method can preserve the attack ability once multiple steps are used. For black-box attack, we find that our method performs better than baseline methods in most cases. For example, our SAI-FGM outperforms I-FGM by nearly 30% for all the black-box models. When we introduce momentum into the attack iterations, we find that performance of our method slightly increases, similar to MI-FGM.

4.2. Attack Robustness Comparison

To evaluate the attack robustness, we do comparisons on two different types of methods: steganalysis based adversarial detection methods and image processing based adversarial defense methods.

Robustness to Steganalysis based Detection Methods Steganalysis-based adversarial detection method [24] uses steganalysis features as the main indicators to detect adversarial samples. We follow their experiment setting and train detector for every adversarial sample generation method including ours for $\delta = 2, 4, 6, 8$ respectively. As shown in Table 2, our SAI-FGM outperforms the baseline methods by a large margin for different δ. For example, if adversarial samples are generated with $\delta = 2$, detect success rate on SAI-FGM with $\beta = 0$ are only 75.50%, while pixel-wise baselines like FGM and I-FGM are 94.32% and 94.11% respectively, nearly 20% higher than SAI-FGM. This proves the superiority of the proposed superpixel-guided adversarial samples over the pixel-wise ones.

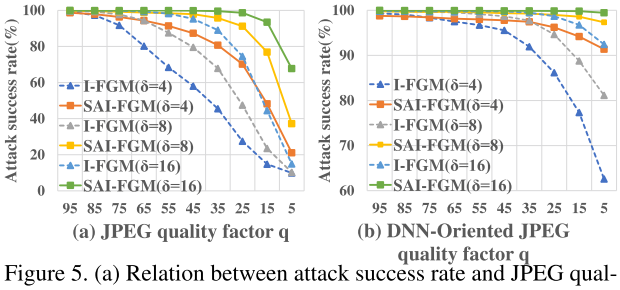

Figure 5. (a) Relation between attack success rate and JPEG quality factor q. (b) Relation between attack success rate and DNN-Oriented JPEG quality factor q.

We further compare SAI-FGM with different β to evaluate the proposed superpixel combination strategy. Obviously, by combining adjacent similar superpixels, SAI-FGM can further reduce the detect success rate for all δ. For example, when $\delta = 8$, detect success rate on SAI-FGM with $\beta = 0$ is 87.20%, while on SAI-FGM with $\beta = 4$, the detect success rate decreases to 80.90%. This is because the statistical difference between adversarial samples and source images is significantly reduced when adjacent similar superpixels are merged.

Robustness to Image Processing based Defense Methods Another important type of adversarial defense methods leverage image processing techniques to remove the adversary before feeding images into the target model. Their underlying hypothesis is similar, i.e., the noisy distribution of perturbations does not match the distribution of real images. In this part, many different image processing techniques are utilized for robustness evaluation, including resizing, JPEG compression [7], DNN-Oriented JPEG Compression [25], pooling, total variance minimization (TVM)[13] and Bit-depth Reduction [42]. To ensure the mis-classification are caused by the adversary of adversarial samples instead of these image processing themselves. We select 2000 robust images from the ImageNet [4] dataset which can be classified correctly after image processing.

Resizing. Resizing is a common and easy image processing method, it can reduce the effectiveness of adversarial samples with local interpolation. Given an adversarial sample with size $H \times W$, we first downscale its size into $\frac{H}{\gamma} \times \frac{W}{\gamma}$ then upscale back to the origin size $H \times W$. Figure 4 is the attack success rate curve when varying the scale factor γ from 1 to 3 with step 0.1. Obviously, for different perturbation scales $\delta = 4, 8, 16$ and different baseline methods, our superpixel-guided attentional versions "SA*" are always much more robust and achieve a higher attack success rate. Especially when $\delta = 16$, the success rate of our SAI-FGM is always at a high level over 80% while that of the baseline I-FGM decreases quickly when γ increases.

Operation	Threshold	Attack method	Kernel Size		
			3	5	7
AVG	$4\sqrt{N}$	I-FGM	73.67	20.55	7.07
		SAI-FGM	**93.27**	**65.14**	**36.91**
	$8\sqrt{N}$	I-FGM	85.94	31.21	11.47
		SAI-FGM	**97.44**	**79.96**	**58.03**
	$16\sqrt{N}$	I-FGM	95.13	46.02	20.36
		SAI-FGM	**99.57**	**91.48**	**77.35**
MAX	$4\sqrt{N}$	I-FGM	41.45	16.67	13.47
		SAI-FGM	**70.37**	**39.20**	**34.67**
	$8\sqrt{N}$	I-FGM	54.79	27.29	20.39
		SAI-FGM	**85.10**	**61.64**	**55.73**
	$16\sqrt{N}$	I-FGM	68.36	37.06	30.38
		SAI-FGM	**93.85**	**83.29**	**79.82**

Table 3. The gray-box attack success rate (%) of adversarial samples after average/max pooling operation on the ImageNet [4] dataset with different ϵ and different kernel size l.

JPEG Compression and DNN-Oriented JPEG Compression. Based on the local smoothness property, JPEG compression significantly compresses the high-frequency components of one image while ensuring similar visual quality. As adversarial perturbations are also of high frequency, their attack ability will be weakened by JPEG compression in most cases. As shown in Figure 5 (a), when the images are only compressed a little with high quality factor $q = 95$, both I-FGM and SAI-FGM can still achieve a quite high attack success rate. But as the quality factor decreases from 95 to 5, attack success rate of I-FGM decrease quickly while SAI-FGM still keep a high success rate. For example, when $q = 55$, success rate of I-FGM with $\delta = 4$ decrease 31%(from 99.4% to 68.4%) and SAI-FGM only decrease 7.5%. Especially at low quality regions($q < 25$), performance of both I-FGM and SAI-FGM decrease greatly as the low quality compression removes most detail of the images, but our method still performs better than I-FGM. For DNN-Oriented JPEG compression shown in Figure 5 (b), the superiority of our method is even more obvious.

Pooling. Pooling is a popular operation in DNNs, which is often conducted by sampling the maximum or average pixels within each grid kernel. Similar to smoothing/resizing, this operation will change the original distribution of adversary, so it can reduce the attack ability especially for large grid kernel size. As shown in Table 3, the proposed superpixel-guided attentional adversaril attack is always much better than the baseline methods for different kernel sizes and perturbation levels. For example, when $\delta = 16$ and the kernel size is 5, our SAI-FGM can still achieve 79.82% attack success rate while the baseline I-FGM only has 30.38% success rate, far behind us with about 50%.

TVM. TVM [13] randomly selects a small set of pixels and reconstructs the "simplest" image that is consistent with the selected pixels. Here we follow the setting in [13] with pixel drop rate as 0.5 and tvm weight as 0.03. From Figure 6(a), it can be seen that SAI-FGM outperforms I-FGM

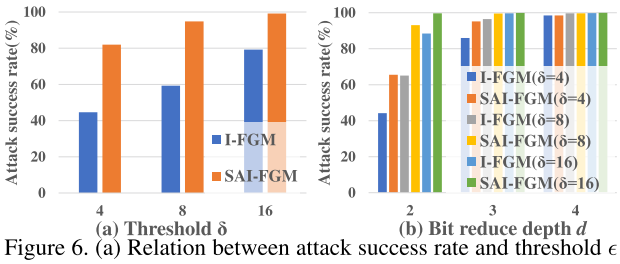

Figure 6. (a) Relation between attack success rate and threshold ϵ after TVM operation. (b) Relation between attack success rate and image reduce depth d.

Method	White-Box	Detection	Resizing	JPEG	TVM
I-FGM	99.40	94.74	46.05	68.40	44.60
SLIC-adv	98.90	82.10	82.17	91.40	82.00
LSC-adv	98.90	82.75	83.10	91.80	83.50
SEEDS-adv	98.75	82.00	81.65	91.21	81.70

Table 4. Attack ability and robustness of adversarial samples generated by SAI-FGM with different superpixel algorithms.

in all settings. For example, when $\delta = 4$, attack success rate of SAI-FGM are 82.00%, while attack performance of I-FGM dropped from 99.48% to 44.60%.

Bit-depth Reduction. Bit-depth Reduction [42] is a simple quantization method to remove small adversarial noises. Here we vary the depth factor d from 2 to 4(64 colors to 4096 colors). In all the cases as shown in Figure 6 (b), our method outperforms baseline methods consistently.

4.3. Visual Results

Besides the visual result shown in Figure 1, we further show several adversarial samples generated by our method SAI-FGM and the baseline I-FGM with $\delta = 4, 16$ in Figure 7. It can be seen that these adversarial samples look overall similar, but the adversarial perturbations of our method are more attentional and superpixel-wise smooth.

4.4. Ablation Study

Superpixel Size S. In this experiment, we study the relationship between the superpixel size S and the attack performance. Intuitively, large S leads to better robustness while small S leads to better attack ability. To prove it, we adopt the attack success rate of original adversarial samples and resized adversarial samples (downscale and upscale) as the indicators of attack ability and attack robustness respectively. To have a clearer difference, $\delta = 4$ is used here. As shown in Figure 8(a), when S varies from 2 to 8, the attack ability decreases slightly while the robustness first increases then decreases (best at $S = 5$). The reason why the robustness of too large S is worse is because the absolute attack ability decreases a lot in such case.

Binarization Factor ϕ. Here we explore the influence of the binarization factor ϕ. Intuitively, if ϕ is too small, most parts of the image are modified so the attentional attack is meaningless. And if ϕ is too large, only a small part of the image will be modified and it performs badly. To show the

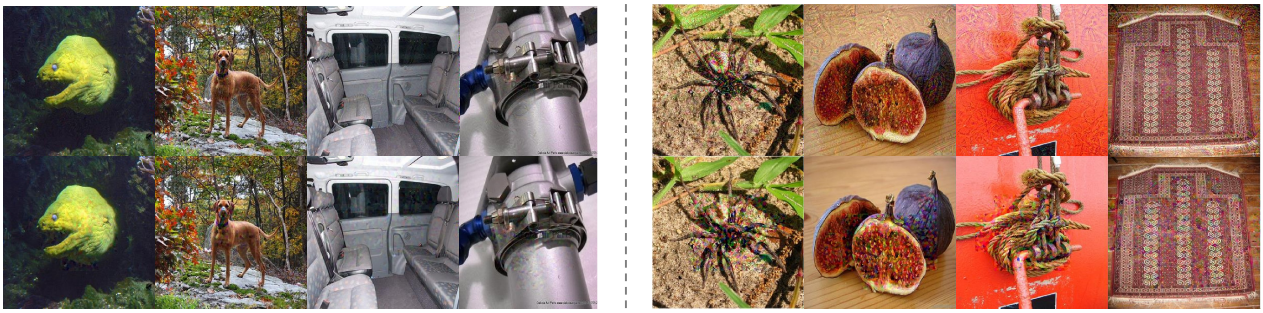

Figure 7. Some visual comparison about the adversarial samples generated by the baseline method I-FGM (top) and our SAI-FGM (bottom) with $\delta = 4$ (left) and $\delta = 16$ (right) respectively.

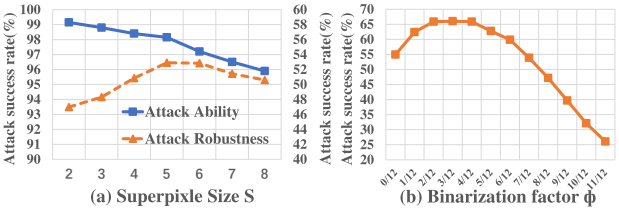

Figure 8. (a) Relation between superpixel size S and attack success rate. Success rate is evaluated by both original and scaled adversarial samples with scale factor $\gamma = 3.0$. (b) Relation between binarization factor k and black-box attack success rate.

performance difference, we use the black-box attack success rate as the indicator and vary ϕ from 0 to $11/12$. It can be seen from Figure 8(b) that the attack success rate first increases then decreases. The best performance is achieved when $\phi = 4/12$, which is better than the case that does not use attention ($\phi = 0$) by 10%. When ϕ is too large, the perturbation space is too small to achieve good attack ability.

Superpixel Algorithms. To evaluate the influence of robustness of different superpixel algorithms, besides the default SLIC used in Sec.4, we further consider another two classic superpixel segmentation methods LSC [21] and SEEDS[39]. In the following, we denote SLIC-adv, LSC-adv, and SEEDS-adv as the adversarial samples generated with the corresponding method. We follow the setting in Sec.4 with $\delta = 4$. For the robustness to resizing, we set $\gamma = 2.0$, and for the robustness to JPEG Compression, we set $q = 55$. As shown in Table.4, we can find that the performance of adversarial samples generated with different superpixel methods are very similar. It indicates that our method is very robust and insensitive to different superpixel segmentation methods. Because the inherent key reason for our robustness is based on the local consistency constraint, as long as the superpixel results follow the rule that pixels within each cluster have similar pixel values (satisfy the requirement of local consistency), our method can always achieve very good performance.

Image Gradient *vs*. Noise Gradient. As we stated in the method part, calculating max or average gradient of pixels within one superpixel as the gradient of the superpixel(here we denote these two method as MAX and AVG respectively) is neither efficient nor effective. In this section, we use the white-box performance to prove that our proposed method are better. When $\delta = 4$, success rate of our method are 98.90%, while success rate of MAX are only 84.20% and AVG are 95.95%. Meanwhile, it's hard to calculate max or average value of irregular image parts parallelly, so we have to use a large memory cost method as substitute, this leads to the result that both MAX and AVG are nearly 30 times slower than our noise gradient method under the same computation resources.

5. Conclusion

In this paper, the limitations of existing "pixel-wise and global" adversarial attack methods are analyzed, which are shown to be the reasons why they are not robust to image processing based defense and steganalysis based detection methods. To address these limitations, we propose the first superpixel-guided attentional adversarial attack method. It constrains the perturbations are only added into the foreground regions and pixels within each superpixel have the same perturbation. Even with such a highly constrained perturbation space, experiments demonstrate that the proposed method can still preserve the original attack ability. Because of better statistical consistency between adversarial samples and source images, our method shows much better robustness to both adversarial detection and defense.

Acknowledgement

This work is supported in part by the Natural Science Foundation of China under Grant U1636201 and 61572452, and by Anhui Initiative in Quantum Information Technologies under Grant AHY150400, and by National Key Research and Development Program of China under Grant 2018YFB0804100, and by Exploration Fund Project of University of Science and Technology of China under Grant YD3480002001. Jiangfan Han, Hongsheng Li, Xiaogang Wang are partially supported by SenseTime Group Limited and the General Research Fund through the Research Grants Council of Hong Kong under Grants CUHK14202217,14203118,14205615,14207814,14213616, 14207319,14208619, and Research Impact Fund R5001-18.

References

[1] Shumeet Baluja and Ian Fischer. Adversarial transformation networks: Learning to generate adversarial examples. *arXiv*, 2017.

[2] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *SP*, pages 39–57. IEEE, 2017.

[3] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Li Chen, Michael E Kounavis, and Duen Horng Chau. Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression. *arXiv*, 2017.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009.

[5] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Xiaolin Hu, Jianguo Li, and Jun Zhu. Boosting adversarial attacks with momentum. *arXiv*, 2017.

[6] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. *CoRR*, abs/1904.02884, 2019.

[7] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv*, 2016.

[8] Zhitao Gong, Wenlu Wang, and Wei-Shinn Ku. Adversarial and clean data are not twins. *arXiv*, 2017.

[9] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

[10] Rafael Gouveia, Aristotle Spyropoulos, and Philippos Mordohai. Confidence estimation for superpixel-based stereo matching. In *3dv*, pages 180–188. IEEE, 2015.

[11] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv*, 2017.

[12] Shixiang Gu and Luca Rigazio. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014.

[13] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv*, 2017.

[14] Jiangfan Han, Xiaoyi Dong, Ruimao Zhang, Dongdong Chen, Weiming Zhang, Nenghai Yu, Ping Luo, and Xiaogang Wang. Once a man: Towards multi-target attack via learning multi-target adversarial network once. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5158–5167, 2019.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[16] Dan Hendrycks and Kevin Gimpel. Visible progress on adversarial images and a new saliency map. *arXiv*, 2016.

[17] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv*, 2016.

[18] Bin Li, Ming Wang, Jiwu Huang, and Xiaolong Li. A new cost function for spatial image steganography. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4206–4210. IEEE, 2014.

[19] Suichan Li, Dapeng Chen, Bin Liu, Nenghai Yu, and Rui Zhao. Memory-based neighbourhood embedding for visual recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[20] Xin Li and Fuxin Li. Adversarial examples detection in deep networks with convolutional filter statistics. In *CVPR*, pages 5764–5772, 2017.

[21] Zhengqin Li and Jiansheng Chen. Superpixel segmentation using linear spectral clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1356–1363, 2015.

[22] Bin Liang, Hongcheng Li, Miaoqiang Su, Xirong Li, Wenchang Shi, and Xiaofeng Wang. Detecting adversarial image examples in deep networks with adaptive noise reduction. *arXiv*, 2017.

[23] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *CVPR*, pages 1778–1787, 2018.

[24] Jiayang Liu, Weiming Zhang, Yiwei Zhang, Dongdong Hou, Yujia Liu, Hongyue Zha, and Nenghai Yu. Detection based defense against adversarial examples from the steganalysis point of view. In *CVPR*, pages 4825–4834, 2019.

[25] Zihao Liu, Qi Liu, Tao Liu, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. *arXiv*, 2018.

[26] S Mahmoudpour and M Kim. Depth from defocus using superpixel-based affinity model and cellular automata. *Electronics Letters*, 52(12):1020–1022, 2016.

[27] Alastair P Moore, Simon JD Prince, Jonathan Warrell, Umar Mohammed, and Graham Jones. Superpixel lattices. In *CVPR*, pages 1–8. Citeseer, 2008.

[28] Margarita Osadchy, Julio Hernandez-Castro, Stuart Gibson, Orr Dunkelman, and Daniel Pérez-Cabo. No bot expects the deepcaptcha! introducing immutable adversarial examples, with applications to captcha generation. *TIFs*, 12(11):2640–2653, 2017.

[29] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *ASIACCS*, pages 506–519. ACM, 2017.

[30] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. Towards the science of security and privacy in machine learning. *arXiv*, 2016.

[31] Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. *arXiv*, 2017.

[32] Achanta Radhakrishna, Shaji Appu, Smith Kevin, Lucchi Aurelien, Fua Pascal, and Süsstrunk Sabine. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 34(11):2274–2282, 2012.

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[34] Ionut Schiopu, Moncef Gabbouj, Atanas Gotchev, and Miska M Hannuksela. Lossless compression of subaperture images using context modeling. In *3DTV-CON*, pages 1–4. IEEE, 2017.

[35] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[36] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

[37] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv*, 2013.

[38] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv*, 2017.

[39] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *European conference on computer vision*, pages 13–26. Springer, 2012.

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[41] Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan L. Yuille. Improving transferability of adversarial examples with input diversity. *CoRR*, abs/1803.06978, 2018.

[42] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv*, 2017.

[43] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016.

[44] Hang Zhou, Kejiang Chen, Weiming Zhang, Han Fang, Wenbo Zhou, and Nenghai Yu. Dup-net: Denoiser and upsampler network for 3d adversarial point clouds defense. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1961–1970, 2019.