Evading Defenses to Transferable Adversarial Examples by Translation-Invariant Attacks

Yinpeng Dong, Tianyu Pang, Hang Su, Jun Zhu*
Dept. of Comp. Sci. and Tech., BNRist Center, State Key Lab for Intell. Tech. & Sys.,
Institute for AI, THBI Lab, Tsinghua University, Beijing, 100084, China
{dyp17, pty17}@mails.tsinghua.edu.cn, {suhangss, dcszj}@mail.tsinghua.edu.cn

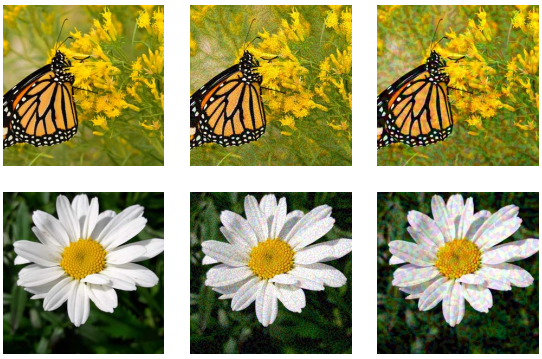
Abstract

Deep neural networks are vulnerable to adversarial examples, which can mislead classifiers by adding imperceptible perturbations. An intriguing property of adversarial examples is their good transferability, making black-box attacks feasible in real-world applications. Due to the threat of adversarial attacks, many methods have been proposed to improve the robustness. Several state-of-the-art defenses are shown to be robust against transferable adversarial examples. In this paper, we propose a translation-invariant attack method to generate more transferable adversarial examples against the defense models. By optimizing a perturbation over an ensemble of translated images, the generated adversarial example is less sensitive to the white-box model being attacked and has better transferability. To improve the efficiency of attacks, we further show that our method can be implemented by convolving the gradient at the untranslated image with a pre-defined kernel. Our method is generally applicable to any gradient-based attack method. Extensive experiments on the ImageNet dataset validate the effectiveness of the proposed method. Our best attack fools eight state-of-the-art defenses at an 82% *success rate on average based only on the transferability, demonstrating the insecurity of the current defense techniques.*

1. Introduction

Despite the great success, deep neural networks have been shown to be highly vulnerable to adversarial examples [3, 32, 10]. These maliciously generated adversarial examples are indistinguishable from legitimate ones by adding small perturbations, but make deep models produce unreasonable predictions. The existence of adversarial examples, even in the physical world [15, 8, 2], has raised concerns in security-sensitive applications, *e.g.*, self-driving cars, healthcare and finance.

*Corresponding author.



Raw Image FGSM TI-FGSM

Figure 1. The adversarial examples generated by the fast gradient sign method (FGSM) [10] and the proposed translation-invariant FGSM (TI-FGSM) for the Inception v3 [31] model.

Attacking deep neural networks has drawn an increasing attention since the generated adversarial examples can serve as an important surrogate to evaluate the robustness of different models [5] and improve the robustness [10, 20]. Several methods have been proposed to generate adversarial examples with the knowledge of the gradient information of a given model, such as fast gradient sign method [10], basic iterative method [15], and Carlini & Wagner's method [5], which are known as *white-box* attacks. Moreover, it is shown that adversarial examples have cross-model transferability [19], *i.e.*, the adversarial examples crafted for one model can fool a different model with a high probability. The transferability enables practical *black-box* attacks to real-world applications and induces serious security issues.

The threat of adversarial examples has motivated extensive research on building robust models or techniques to defend against adversarial attacks. These include training with adversarial examples [10, 33, 20], image denoising/transformation [18, 36, 11], theoretically-certified defenses [26, 35], and others [24, 29, 28]. Although the non-certified defenses have demonstrated robustness against common attacks, they do so by causing obfuscated gradients, which can be easily circumvented by new attacks [1].

978-1-7281-3293-8/19/$31.00 ©2019 IEEE
DOI 10.1109/CVPR.2019.00444

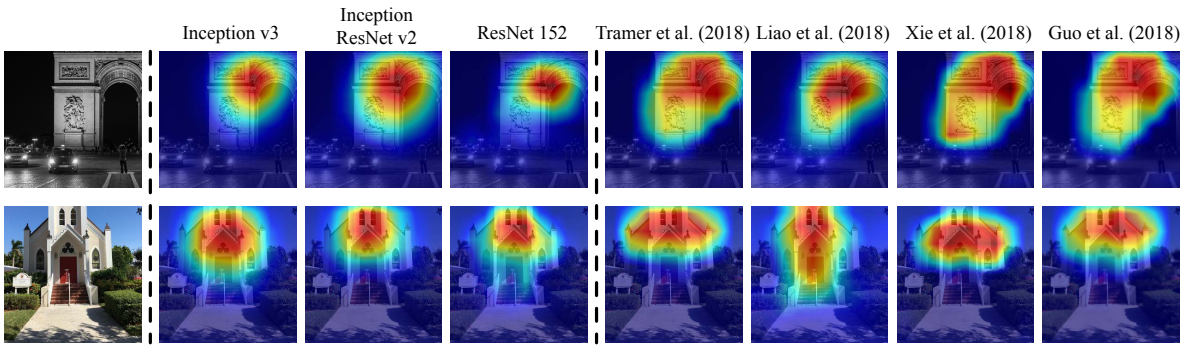


Figure 2. Demonstration of the different discriminative regions of the defense models compared with normally trained models. We adopt *class activation mapping* [38] to visualize the attention maps of three normally trained models—Inception v3 [31], Inception ResNet v2 [30], ResNet 152 [12] and four defense models [33, 18, 36, 11]. These defense models rely on different discriminative regions for predictions compared with normally trained models, which could affect the transferability of adversarial examples.

However, some of the defenses [33, 18, 36, 11] claim to be resistant to transferable adversarial examples, making it difficult to evade them by black-box attacks.

The resistance of the defense models against transferable adversarial examples is largely due to the phenomenon that the defenses make predictions based on different discriminative regions compared with normally trained models. For example, we show the attention maps of several normally trained models and defense models in Fig. 2, to represent the discriminative regions for their predictions. It can be seen that the normally trained models have similar attention maps while the defenses induce different attention maps. A similar observation is also found in [34] that the gradients of the defenses in the input space align well with human perception, while those of normally trained models appear very noisy. This phenomenon of the defenses is caused by either training under different data distributions [33] or transforming the inputs before classification [18, 36, 11]. For black-box attacks based on the transferability [10, 19, 7], an adversarial example is usually generated for a single input against a white-box model. So the generated adversarial example is highly correlated with the discriminative region or gradient of the white-box model at the given input point, making it hard to transfer to other defense models that depend on different regions for predictions. Therefore, the transferability of adversarial examples is largely reduced to the defenses.

To mitigate the effect of different discriminative regions between models and evade the defenses by transferable adversarial examples, we propose a **translation-invariant attack** method. In particular, we generate an adversarial example for an ensemble of images composed of a legitimate one and its translated versions. We expect that the resultant adversarial example is less sensitive to the discriminative region of the white-box model being attacked, and has a higher probability to fool another black-box model with a defense mechanism. However, to generate such a perturbation, we need to calculate the gradients for all images in the ensemble, which brings much more computations. To improve the efficiency of our attacks, we further show that our method can be implemented by convolving the gradient at the untranslated image with a pre-defined kernel under a mild assumption. By combining the proposed method with any gradient-based attack method (*e.g.*, fast gradient sign method [10], *etc.*), we obtain more transferable adversarial examples with similar computation complexity.

Extensive experiments on the ImageNet dataset [27] demonstrate that the proposed translation-invariant attack method helps to improve the success rates of black-box attacks against the defense models by a large margin. Our best attack reaches an average success rate of 82% to evade eight state-of-the-art defenses based only on the transferability, thus demonstrating the insecurity of the current defenses.

2. Related Work

Adversarial examples. Deep neural networks have been shown to be vulnerable to adversarial examples first in the visual domain [32]. Then several methods are proposed to generate adversarial examples for the purpose of high success rates and minimal size of perturbations [10, 15, 5]. They also exist in the physical world [15, 8, 2]. Although adversarial examples are recently crafted for many other domains, we focus on image classification tasks in this paper.

Black-box attacks. Black-box adversaries have no access to the model parameters or gradients. The transferability [19] of adversarial examples can be used to attack a black-box model. Several methods [7, 37] have been proposed to improve the transferability, which enable powerful black-box attacks. Besides the transfer-based black-box attacks, there is another line of work that performs attacks based on adaptive queries. For example, Papernot *et al*. [25] use queries to distill the knowledge of the target model and train a surrogate model. They therefore turn the black-box attacks to the white-box attacks. Recent methods use queries to estimate the gradient or the decision boundary of the black-box model [6, 4] to generate adversarial examples.

However, these methods usually require a large number of queries, which is impractical in real-world applications. In this paper, we resort to transfer-based black-box attacks.

Attacks for an ensemble of examples. An adversarial perturbation can be generated for an ensemble of legitimate examples. In [22], the universal perturbations are generated for the entire data distribution, which can fool the models on most of natural images. In [2], the adversarial perturbation is optimized over a distribution of transformations, which is similar to our method. The major difference between the method in [2] and ours is three-fold. First, we want to generate transferable adversarial examples against the defense models, while the authors in [2] propose to synthesize robust adversarial examples in the physical world. Second, we only use the translation operation, while they use a lot of transformations such as rotation, translation, addition of noise, *etc*. Third, we develop an efficient algorithm for optimization that only needs to calculate the gradient for the untranslated image, while they calculate the gradients for a batch of transformed images by sampling.

Defend against adversarial attacks. A large variety of methods have been proposed to increase the robustness of deep learning models. Besides directly making the models produce correct predictions for adversarial examples, some methods attempt to detect them instead [21, 23]. However, most of the non-certified defenses demonstrate the robustness by causing obfuscated gradients, which can be successfully circumvented by new attacks [1]. Although these defenses are not robust in the white-box setting, some of them [33, 18, 36, 11] empirically show the resistance against transferable adversarial examples in the black-box setting. In this paper, we focus on generating more transferable adversarial examples against these defenses.

3. Methodology

In this section, we provide the detailed description of our algorithm. Let $\boldsymbol{x}^{real}$ denote a real example and y denote the corresponding ground-truth label. Given a classifier $f(\boldsymbol{x}) : \mathcal{X} \rightarrow \mathcal{Y}$ that outputs a label as the prediction for an input, we want to generate an adversarial example $\boldsymbol{x}^{adv}$ which is visually indistinguishable from $\boldsymbol{x}^{real}$ but fools the classifier, *i.e.*, $f(\boldsymbol{x}^{adv}) \neq y$.[1] In most cases, the L_p norm of the adversarial perturbation is required to be smaller than a threshold ϵ as $||\boldsymbol{x}^{adv} - \boldsymbol{x}^{real}||_p \leq \epsilon$. In this paper, we use the L_∞ norm as the measurement. For adversarial example generation, the objective is to maximize the loss function $J(\boldsymbol{x}^{adv}, y)$ of the classifier, where J is often the cross-entropy loss. So the constrained optimization problem can be written as

$$\arg\max_{\boldsymbol{x}^{adv}} J(\boldsymbol{x}^{adv}, y), \quad \text{s.t. } \|\boldsymbol{x}^{adv} - \boldsymbol{x}^{real}\|_\infty \leq \epsilon. \quad (1)$$

[1]This corresponds to untargeted attack. The method in this paper can be simply extended to targeted attack.

To solve this optimization problem, the gradient of the loss function with respect to the input needs to be calculated, termed as white-box attacks. However, in some cases, we cannot get access to the gradients of the classifier, where we need to perform attacks in the black-box manner. We resort to transferable adversarial examples which are generated for a different white-box classifier but have high transferability for black-box attacks.

3.1. Gradient-based Adversarial Attack Methods

Several methods have been proposed to solve the optimization problem in Eq. (1). We give a brief introduction of them in this section.

Fast Gradient Sign Method (FGSM) [10] generates an adversarial example $\boldsymbol{x}^{adv}$ by linearizing the loss function in the input space and performing one-step update as

$$\boldsymbol{x}^{adv} = \boldsymbol{x}^{real} + \epsilon \cdot \text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}^{real}, y)), \quad (2)$$

where $\nabla_{\boldsymbol{x}} J$ is the gradient of the loss function with respect to $\boldsymbol{x}$. $\text{sign}(\cdot)$ is the sign function to make the perturbation meet the L_∞ norm bound. FGSM can generate more transferable adversarial examples but is usually not effective enough for attacking white-box models [16].

Basic Iterative Method (BIM) [15] extends FGSM by iteratively applying gradient updates multiple times with a small step size α, which can be expressed as

$$\boldsymbol{x}^{adv}_{t+1} = \boldsymbol{x}^{adv}_t + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}^{adv}_t, y)), \quad (3)$$

where $\boldsymbol{x}^{adv}_0 = \boldsymbol{x}^{real}$. To restrict the generated adversarial examples within the ϵ-ball of $\boldsymbol{x}^{real}$, we can clip $\boldsymbol{x}^{adv}_t$ after each update, or set $\alpha = \epsilon/T$, with T being the number of iterations. It has been shown that BIM induces much more powerful white-box attacks than FGSM at the cost of worse transferability [16, 7].

Momentum Iterative Fast Gradient Sign Method (MI-FGSM) [7] proposes to improve the transferability of adversarial examples by integrating a momentum term into the iterative attack method. The update procedure is

$$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} J(\boldsymbol{x}^{adv}_t, y)}{\|\nabla_{\boldsymbol{x}} J(\boldsymbol{x}^{adv}_t, y)\|_1}, \quad (4)$$

$$\boldsymbol{x}^{adv}_{t+1} = \boldsymbol{x}^{adv}_t + \alpha \cdot \text{sign}(\boldsymbol{g}_{t+1}), \quad (5)$$

where $\boldsymbol{g}_t$ gathers the gradient information up to the t-th iteration with a decay factor μ.

Diverse Inputs Method [37] applies random transformations to the inputs and feeds the transformed images into the classifier for gradient calculation. The transformation includes random resizing and padding with a given probability. This method can be combined with the momentum-based method to further improve the transferability.

Carlini & Wagner's method (C&W) [5] is a powerful optimization-based method, which solves

$$\arg\min_{\boldsymbol{x}^{adv}} \|\boldsymbol{x}^{adv} - \boldsymbol{x}^{real}\|_p - c \cdot J(\boldsymbol{x}^{adv}, y), \tag{6}$$

where the loss function J could be different from the cross-entropy loss. This method aims to find adversarial examples with minimal size of perturbations, to measure the robustness of different models. It also lacks the effectiveness for black-box attacks like BIM.

3.2. Translation-Invariant Attack Method

Although many attack methods [7, 37] can generate adversarial examples with very high transferability across normally trained models, they are less effective to attack defense models in the black-box manner. Some of the defenses [33, 18, 36, 11] are shown to be quite robust against black-box attacks. So we want to answer that: *Are these defenses really free from transferable adversarial examples?*

We find that the discriminative regions used by the defenses to identify object categories are different from those used by normally trained models, as shown in Fig. 2. When generating an adversarial example by the methods introduced in Sec. 3.1, the adversarial example is only optimized for a single legitimate example. So it may be highly correlated with the discriminative region or gradient of the white-box model being attacked at the input data point. For other black-box defense models that have different discriminative regions or gradients, the adversarial example can hardly remain adversarial. Therefore, the defenses are shown to be robust against transferable adversarial examples.

To generate adversarial examples that are less sensitive to the discriminative regions of the white-box model, we propose a **translation-invariant attack** method. In particular, rather than optimizing the objective function at a single point as Eq. (1), the proposed method uses a set of translated images to optimize an adversarial example as

$$\begin{aligned} &\arg\max_{\boldsymbol{x}^{adv}} \sum_{i,j} w_{ij} J(T_{ij}(\boldsymbol{x}^{adv}), y), \\ &\text{s.t. } \|\boldsymbol{x}^{adv} - \boldsymbol{x}^{real}\|_\infty \le \epsilon, \end{aligned} \tag{7}$$

where $T_{ij}(\boldsymbol{x})$ is the translation operation that shifts image $\boldsymbol{x}$ by i and j pixels along the two-dimensions respectively, *i.e.*, each pixel (a, b) of the translated image is $T_{ij}(\boldsymbol{x})_{a,b} = x_{a-i,b-j}$, and w_{ij} is the weight for the loss $J(T_{ij}(\boldsymbol{x}^{adv}), y)$. We set $i, j \in \{-k, ..., 0, ..., k\}$ with k being the maximal number of pixels to shift. With this method, the generated adversarial examples are less sensitive to the discriminative regions of the white-box model being attacked, which may be transferred to another model with a higher success rate. We choose the translation operation in this paper rather than other transformations (*e.g.*, rotation, scaling, *etc.*), because we can develop an efficient algorithm to calculate the gradient of the loss function by the assumption of the translation-invariance [17] in convolutional neural networks.

3.2.1 Gradient Calculation

To solve the optimization problem in Eq. (7), we need to calculate the gradients for $(2k+1)^2$ images, which introduces much more computations. Sampling a small number of translated images for gradient calculation is a feasible way [2]. But we show that we can calculate the gradient for only one image under a mild assumption.

Convolutional neural networks are supposed to have the translation-invariant property [17], that an object in the input can be recognized in spite of its position. In practice, CNNs are not truly translation-invariant [9, 14]. So we make an assumption that the translation-invariant property is nearly held with very small translations (which is empirically validated in Sec. 4.2). In our problem, we shift the image by no more than 10 pixels along each dimension (*i.e.*, $k \le 10$). Therefore, based on this assumption, the translated image $T_{ij}(\boldsymbol{x})$ is almost the same as $\boldsymbol{x}$ as inputs to the models, as well as their gradients

$$\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=T_{ij}(\hat{\boldsymbol{x}})} \approx \nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}. \tag{8}$$

We then calculate the gradient of the loss function defined in Eq. (7) at a point $\hat{\boldsymbol{x}}$ as

$$\begin{aligned} &\nabla_{\boldsymbol{x}}\big(\sum_{i,j} w_{ij} J(T_{ij}(\boldsymbol{x}), y)\big)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}} \\ =&\sum_{i,j} w_{ij} \nabla_{\boldsymbol{x}} J(T_{ij}(\boldsymbol{x}), y)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}} \\ =&\sum_{i,j} w_{ij} \big(\nabla_{T_{ij}(\boldsymbol{x})} J(T_{ij}(\boldsymbol{x}), y) \cdot \frac{\partial T_{ij}(\boldsymbol{x})}{\partial \boldsymbol{x}}\big)\Big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}} \\ =&\sum_{i,j} w_{ij} T_{-i-j}\big(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=T_{ij}(\hat{\boldsymbol{x}})}\big) \\ \approx&\sum_{i,j} w_{ij} T_{-i-j}(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}). \end{aligned} \tag{9}$$

Given Eq. (9), we do not need to calculate the gradients for $(2k+1)^2$ images. Instead, we only need to get the gradient at the untranslated image $\hat{\boldsymbol{x}}$ and then average all the shifted gradients. This procedure is equivalent to convolving the gradient with a kernel composed of all the weights w_{ij} as

$$\sum_{i,j} w_{ij} T_{-i-j}(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}) \Leftrightarrow \boldsymbol{W} * \nabla_{\boldsymbol{x}} J(\boldsymbol{x}, y)\big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}},$$

where $\boldsymbol{W}$ is the kernel matrix of size $(2k+1) \times (2k+1)$, with $W_{i,j} = w_{-i-j}$. We will specify $\boldsymbol{W}$ in the next section.

3.2.2 Kernel Matrix

There are many options to generate the kernel matrix $\boldsymbol{W}$. A basic design principle is that the images with bigger shifts

should have relatively lower weights to make the adversarial perturbation fool the model at the untranslated image effectively. In this paper, we consider three different choices:

(1) A uniform kernel that $W_{i,j} = 1/(2k+1)^2$;

(2) A linear kernel that $\tilde{W}_{i,j} = (1 - |i|/k+1) \cdot (1 - |j|/k+1)$, and $W_{i,j} = \tilde{W}_{i,j} / \sum_{i,j} \tilde{W}_{i,j}$;

(3) A Gaussian kernel that $\tilde{W}_{i,j} = \frac{1}{2\pi\sigma^2} \exp(-\frac{i^2+j^2}{2\sigma^2})$ where the standard deviation $\sigma = k/\sqrt{3}$ to make the radius of the kernel be 3σ, and $W_{i,j} = \tilde{W}_{i,j} / \sum_{i,j} \tilde{W}_{i,j}$.

We will empirically compare the three kernels in Sec. 4.3.

3.2.3 Attack Algorithms

Note that in Sec. 3.2.1, we only illustrate how to calculate the gradient of the loss function defined in Eq. (7), but do not specify the update algorithm for generating adversarial examples. This indicates that our method can be integrated into any gradient-based attack method, *e.g.*, FGSM, BIM, MI-FGSM, *etc*. For gradient-based attack methods presented in Sec. 3.1, in each step we calculate the gradient $\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^{adv}, y)$ at the current solution $\boldsymbol{x}_t^{adv}$, then convolve the gradient with the pre-defined kernel $\boldsymbol{W}$, and finally obtain the new solution $\boldsymbol{x}_{t+1}^{adv}$ following the update rule in different attack methods. For example, the combination of our translation-invariant method and the fast gradient sign method [10] (TI-FGSM) has the following update rule

$$\boldsymbol{x}^{adv} = \boldsymbol{x}^{real} + \epsilon \cdot \text{sign}(\boldsymbol{W} * \nabla_{\boldsymbol{x}} J(\boldsymbol{x}^{real}, y)). \quad (10)$$

Also, the integration of the translation-invariant method into the basic iterative method [15] yields the TI-BIM algorithm

$$\boldsymbol{x}_{t+1}^{adv} = \boldsymbol{x}_t^{adv} + \alpha \cdot \text{sign}(\boldsymbol{W} * \nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^{adv}, y)). \quad (11)$$

The translation-invariant method can be similarly integrated into MI-FGSM [7] and DIM [37] as TI-MI-FGSM and TI-DIM, respectively.

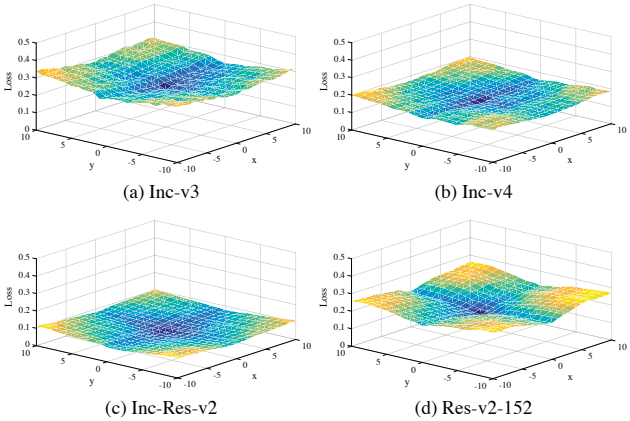
4. Experiments

In this section, we present the experimental results to demonstrate the effectiveness of the proposed method. We first specify the experimental settings in Sec. 4.1. Then we validate the translation-invariant property of convolutional neural networks in Sec. 4.2. We further conduct two experiments to study the effects of different kernels and size of kernels in Sec. 4.3 and Sec. 4.4. We finally compare the results of the proposed method with baseline methods in Sec. 4.5 and Sec. 4.6.

4.1. Experimental Settings

We use an ImageNet-compatible dataset[2] comprised of 1,000 images to conduct experiments. This dataset was used

[2] https://github.com/tensorflow/cleverhans/tree/master/examples/nips17_adversarial_competition/dataset



(a) Inc-v3 (b) Inc-v4

(c) Inc-Res-v2 (d) Res-v2-152

Figure 3. We show the loss surfaces of Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 given the translated images at each position.

in the NIPS 2017 adversarial competition. We include eight defense models which are shown to be robust against black-box attacks on the ImageNet dataset. These are

- Inc-v3$_{ens3}$, Inc-v3$_{ens4}$, IncRes-v2$_{ens}$ [33];
- high-level representation guided denoiser (HGD, rank-1 submission in the NIPS 2017 defense competition) [18];
- input transformation through random resizing and padding (R&P, rank-2 submission in the NIPS 2017 defense competition) [36];
- input transformation through JPEG compression or total variance minimization (TVM) [11];
- rank-3 submission[3] in the NIPS 2017 defense competition (NIPS-r3).

To attack these defenses based on the transferability, we also include four normally trained models—Inception v3 (Inc-v3) [31], Inception v4 (Inc-v4), Inception ResNet v2 (IncRes-v2) [30], and ResNet v2-152 (Res-v2-152) [13], as the white-box models to generate adversarial examples.

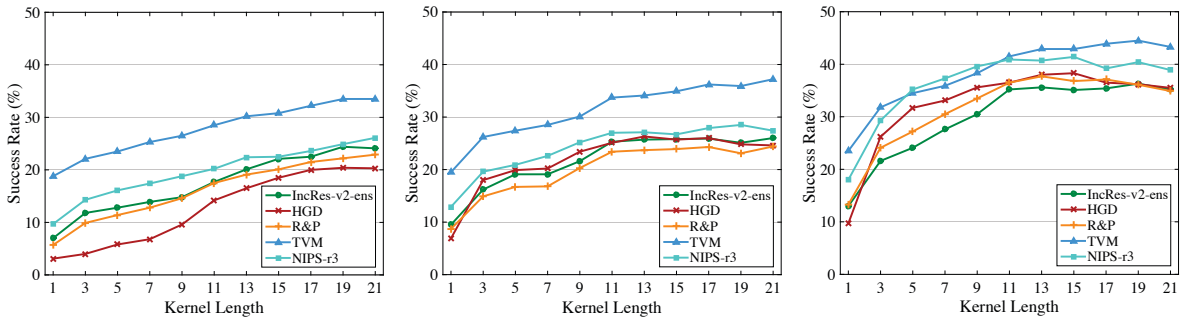
In our experiments, we integrate our method into the fast gradient sign method (FGSM) [10], momentum iterative fast gradient sign method (MI-FGSM) [7], and diverse inputs method (DIM) [37]. We do not include the basic iterative method [15] and C&W's method [5] since that they are not good at generating transferable adversarial examples [7]. We denote the attacks combined with our translation-invariant method as TI-FGSM, TI-MI-FGSM, and TI-DIM, respectively.

For the settings of hyper-parameters, we set the maximum perturbation to be $\epsilon = 16$ among all experiments with pixel values in $[0, 255]$. For the iterative attack methods, we set the number of iteration as 10 and the step size as $\alpha = 1.6$. For MI-FGSM and TI-MI-FGSM, we adopt the default decay factor $\mu = 1.0$. For DIM and TI-DIM, the

[3] https://github.com/anlthms/nips-2017/tree/master/mmd

	Attack	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	HGD	R&P	JPEG	TVM	NIPS-r3
TI-FGSM	Uniform	25.0	27.9	21.1	15.7	19.1	24.8	32.3	21.9
	Linear	**30.7**	**32.4**	**24.2**	**20.9**	**23.3**	**28.1**	**34.6**	**25.8**
	Gaussian	28.2	28.9	22.3	18.4	19.8	25.5	30.7	24.5
TI-MI-FGSM	Uniform	30.0	32.2	22.8	21.7	22.8	26.4	32.7	25.9
	Linear	**35.8**	35.0	**26.8**	25.5	23.4	**29.0**	**35.8**	**27.5**
	Gaussian	**35.8**	**35.1**	25.8	**25.7**	**23.9**	28.2	34.9	26.7
TI-DIM	Uniform	32.6	34.6	25.6	24.1	27.2	30.2	34.9	28.8
	Linear	45.2	47.0	34.9	35.6	35.2	**38.5**	43.6	39.7
	Gaussian	**46.9**	**47.1**	**37.4**	**38.3**	**36.8**	37.0	**44.2**	**41.4**

Table 1. The success rates (%) of black-box attacks against eight defenses with different choices of kernels. The adversarial examples are crafted for Inc-v3 by TI-FGSM, TI-MI-FGSM and TI-DIM with the uniform kernel, the linear kernel, and the Gaussian kernel, respectively.



(a) TI-FGSM (b) TI-MI-FGSM (c) TI-DIM

Figure 4. The success rates (%) of black-box attacks against IncRes-v2$_{ens}$, HGD, R&P, TVM, and NIPS-r3. The adversarial examples are generated for Inc-v3 with the kernel length ranging from 1 to 21.

transformation probability is set to 0.7. Please note that the settings for each attack method and its translation-invariant version are the same, because our method is not concerned with the specific attack procedure.

4.2. Translation-Invariant Property of CNNs

We first verify the translation-invariant property of convolutional neural networks in this section. We use the original 1,000 images from the dataset and shift them by -10 to 10 pixels in each dimension. We input the original images as well as the translated images into Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152, respectively. The loss of each input image is given by the models. We average the loss over all translated images at each position, and show the loss surfaces in Fig. 3.

It can be seen that the loss surfaces are generally smooth with the translations going from -10 to 10 in each dimension. So we could make the assumption that the translation-invariant property is almost held within a small range. In our attacks, the images are shifted by no more than 10 pixels along each dimension. The loss values would be very similar for the original and translated images. Therefore, we regard that a translated image is almost the same as the corresponding original image as inputs to the models.

4.3. The Results of Different Kernels

In the section, we show the experimental results of the proposed translation-invariant attack method with different choices of kernels. We attack the Inc-v3 model by TI-FGSM, TI-MI-FGSM, and TI-DIM with three types of kernels, *i.e.*, uniform kernel, linear kernel, and Gaussian kernel, as introduced in Sec. 3.2.2. In Table 1, we report the success rates of black-box attacks against the eight defense models we study, where the success rates are the misclassification rates of the corresponding defense models with the generated adversarial images as inputs.

We can see that for TI-FGSM, the linear kernel leads to better results than the uniform kernel and the Gaussian kernel. And for more powerful attacks such as TI-MI-FGSM and TI-DIM, the Gaussian kernel achieves similar or even better results than the linear kernel. However, both of the linear kernel and the Gaussian kernel are more effective than the uniform kernel. It indicates that we should design the kernel that has lower weights for bigger shifts, as discussed in Sec. 3.2.2. We simply adopt the Gaussian kernel in the following experiments.

4.4. The Effect of Kernel Size

The size of the kernel $\boldsymbol{W}$ also plays a key role for improving the success rates of black-box attacks. If the kernel size equals to 1×1, the translation-invariant based attacks degenerate to their vanilla versions. Therefore, we conduct an ablation study to examine the effect of kernel sizes.

We attack the Inc-v3 model by TI-FGSM, TI-MI-FGSM, and TI-DIM with the Gaussian kernel, whose length ranges from 1 to 21 with a granularity 2. In Fig. 4, we show the suc-

	Attack	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	HGD	R&P	JPEG	TVM	NIPS-r3
Inc-v3	FGSM	15.6	14.7	7.0	2.1	6.5	19.9	18.8	9.8
	TI-FGSM	**28.2**	**28.9**	**22.3**	**18.4**	**19.8**	**25.5**	**30.7**	**24.5**
Inc-v4	FGSM	16.2	16.1	9.0	2.6	7.9	21.8	19.9	11.5
	TI-FGSM	**28.2**	**28.3**	**21.4**	**18.1**	**21.6**	**27.9**	**31.8**	**24.6**
IncRes-v2	FGSM	18.0	17.2	10.2	3.9	9.9	24.7	23.4	13.3
	TI-FGSM	**32.8**	**33.6**	**28.1**	**25.4**	**28.1**	**32.4**	**38.5**	**31.4**
Res-v2-152	FGSM	20.2	17.7	9.9	3.6	8.6	24.0	22.0	12.5
	TI-FGSM	**34.6**	**34.5**	**27.8**	**24.4**	**27.4**	**32.7**	**38.1**	**30.1**

Table 2. The success rates (%) of black-box attacks against eight defenses. The adversarial examples are crafted for Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 respectively using FGSM and TI-FGSM.

	Attack	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	HGD	R&P	JPEG	TVM	NIPS-r3
Inc-v3	MI-FGSM	20.5	17.4	9.5	6.9	8.7	20.3	19.4	12.9
	TI-MI-FGSM	**35.8**	**35.1**	**25.8**	**25.7**	**23.9**	**28.2**	**34.9**	**26.7**
Inc-v4	MI-FGSM	22.1	20.1	12.1	9.6	12.1	26.0	24.8	15.6
	TI-MI-FGSM	**36.7**	**39.2**	**28.7**	**27.8**	**28.0**	**31.6**	**38.4**	**29.5**
IncRes-v2	MI-FGSM	31.3	27.2	19.7	19.6	18.6	31.6	34.4	22.7
	TI-MI-FGSM	**50.7**	**51.7**	**49.3**	**45.1**	**45.2**	**45.9**	**55.4**	**46.2**
Res-v2-152	MI-FGSM	25.1	23.7	13.3	15.1	14.6	31.2	24.5	18.0
	TI-MI-FGSM	**39.9**	**37.7**	**32.8**	**31.8**	**31.1**	**38.3**	**41.2**	**34.4**

Table 3. The success rates (%) of black-box attacks against eight defenses. The adversarial examples are crafted for Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 respectively using MI-FGSM and TI-MI-FGSM.

Figure 5. The adversarial examples generated for Inc-v3 by TI-FGSM with different kernel sizes.

cess rates against five defense models—IncRes-v2$_{ens}$, HGD, R&P, TVM, and NIPS-r3. The success rate continues increasing at first, and turns to remain stable after the kernel size exceeds 15×15. Therefore, the size of the kernel is set to 15×15 in the following.

We also show the adversarial images generated for the Inc-v3 model by TI-FGSM with different kernel sizes in Fig. 5. Due to the smooth effect given by the kernel, we can see that the adversarial perturbations are smoother when using a bigger kernel.

4.5. Single-Model Attacks

In this section, we compare the black-box success rates of the translation-invariant based attacks with baseline attacks. We first perform adversarial attacks for Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 respectively using FGSM, MI-FGSM, DIM, and their extensions by combining with the translation-invariant attack method as TI-FGSM, TI-MI-FGSM, and TI-DIM. We adopt the 15×15 Gaussian kernel in this set of experiments. We then use the generated adversarial examples to attack the eight defense models we consider based only on the transferability. We report the success rates of black-box attacks in Table 2 for FGSM and TI-FGSM, Table 3 for MI-FGSM and TI-MI-FGSM, and Table 4 for DIM and TI-DIM.

From the tables, we observe that the success rates against the defenses are improved by a large margin when using the proposed method regardless of the attack algorithms or the white-box models being attacked. In general, the translation-invariant based attacks consistently outperform the baseline attacks by $5\% \sim 30\%$. In particular, when using TI-DIM, the combination of our method and DIM, to attack the IncRes-v2 model, the resultant adversarial examples have about 60% success rates against the defenses (as shown in Table 4). It demonstrates the vulnerability of the current defenses against black-box attacks. The results also validate the effectiveness of the proposed method. Although we only compare the results of our attack method with base-

	Attack	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	HGD	R&P	JPEG	TVM	NIPS-r3
Inc-v3	DIM	24.2	24.3	13.0	9.7	13.3	30.7	24.4	18.0
	TI-DIM	**46.9**	**47.1**	**37.4**	**38.3**	**36.8**	**37.0**	**44.2**	**41.4**
Inc-v4	DIM	28.3	27.5	15.6	14.6	17.2	38.6	29.1	14.1
	TI-DIM	**48.6**	**47.5**	**38.7**	**40.3**	**39.3**	**43.5**	**45.6**	**41.9**
IncRes-v2	DIM	41.2	40.0	27.9	32.4	30.2	47.2	41.7	37.6
	TI-DIM	**61.3**	**60.1**	**59.5**	**58.7**	**61.4**	**55.7**	**66.2**	**61.5**
Res-v2-152	DIM	40.5	36.0	24.1	32.6	26.4	42.4	36.8	34.4
	TI-DIM	**56.1**	**55.5**	**49.5**	**51.8**	**50.4**	**50.8**	**55.7**	**52.9**

Table 4. The success rates (%) of black-box attacks against eight defenses. The adversarial examples are crafted for Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 respectively using DIM and TI-DIM.

Attack	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	HGD	R&P	JPEG	TVM	NIPS-r3
FGSM	27.5	23.7	13.4	4.9	13.8	38.1	30.0	19.8
TI-FGSM	**39.1**	**38.8**	**31.6**	**29.9**	**31.2**	**43.3**	**39.8**	**33.9**
MI-FGSM	50.5	48.3	32.8	38.6	32.8	67.7	50.1	43.9
TI-MI-FGSM	**76.4**	**74.4**	**69.6**	**73.3**	**68.3**	**77.2**	**72.1**	**71.4**
DIM	66.0	63.3	45.9	57.7	51.7	82.5	64.1	63.7
TI-DIM	**84.8**	**82.7**	**78.0**	**82.6**	**81.4**	**83.4**	**79.8**	**83.1**

Table 5. The success rates (%) of black-box attacks against eight defenses. The adversarial examples are crafted for the ensemble of Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 using FGSM, TI-FGSM, MI-FGSM, TI-MI-FGSM, DIM, and TI-DIM.

line methods against the defense models, our attacks remain the success rates of baseline attacks in the white-box setting and the black-box setting against normally trained models, which will be shown in the Appendix.

We show two adversarial images generated for the Inc-v3 model by FGSM and TI-FGSM in Fig. 1. It can be seen that by using TI-FGSM, in which the gradients are convolved by a kernel $\boldsymbol{W}$ before applying to the raw images, the adversarial perturbations are much smoother than those generated by FGSM. The smooth effect also exists in other translation-invariant based attacks.

4.6. Ensemble-based Attacks

In this section, we further present the results when adversarial examples are generated for an ensemble of models. Liu *et al*. [19] have shown that attacking multiple models at the same time can improve the transferability of the generated adversarial examples. It is due to that if an example remains adversarial for multiple models, it is more likely to transfer to another black-box model.

We adopt the ensemble method proposed in [7], which fuses the logit activations of different models. We attack the ensemble of Inc-v3, Inc-v4, IncRes-v2, and Res-v2-152 with equal ensemble weights using FGSM, TI-FGSM, MI-FGSM, TI-MI-FGSM, DIM, and TI-DIM respectively. We also use the 15×15 Gaussian kernel in the translation-invariant based attacks.

In Table 5, we show the results of black-box attacks against the eight defenses. The proposed method also improves the success rates across all experiments over the baseline attacks. It should be noted that *the adversarial examples generated by TI-DIM can fool the state-of-the-art defenses at an* 82% *success rate on average based on the transferability*. And the adversarial examples are generated for normally trained models unaware of the defense strategies. The results in the paper demonstrate that the current defenses are far from real security, and cannot be deployed in real-world applications.

5. Conclusion

In this paper, we proposed a translation-invariant attack method to generate adversarial examples that are less sensitive to the discriminative regions of the white-box model being attacked, and have higher transferability against the defense models. Our method optimizes an adversarial image by using a set of translated images. Based on an assumption, our method is efficiently implemented by convolving the gradient with a pre-defined kernel, and can be integrated into any gradient-based attack method. We conducted experiments to validate the effectiveness of the proposed method. Our best attack, TI-DIM, the combination of the proposed translation-invariant method and diverse inputs method [37], can fool eight state-of-the-art defenses at an 82% success rate on average, where the adversarial examples are generated against four normally trained models. The results identify the vulnerability of the current defenses, and thus raise security issues for the development of more robust deep learning models. We make our codes public at https://github.com/dongyp13/Translation-Invariant-Attacks.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (No. 2017YFA0700904), NSFC Projects (Nos. 61620106010, 61621136008, 61571261), Beijing NSF Project (No. L172037), DITD Program JCKY2017204B064, Tiangong Institute for Intelligent Computing, NVIDIA NVAIL Program, and the projects from Siemens and Intel.

References

[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018. 1, 3

[2] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *ICML*, 2018. 1, 2, 3, 4

[3] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 387–402, 2013. 1

[4] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *ICLR*, 2018. 2

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 2017. 1, 2, 4, 5

[6] Pin Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *ACM Workshop on Artificial Intelligence and Security (AISec)*, pages 15–26, 2017. 2

[7] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018. 2, 3, 4, 5, 8

[8] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *CVPR*, 2018. 1, 2

[9] Ian Goodfellow, Honglak Lee, Quoc V Le, Andrew Saxe, and Andrew Y Ng. Measuring invariances in deep networks. In *NIPS*, 2009. 4

[10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 1, 2, 3, 5

[11] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. In *ICLR*, 2018. 1, 2, 3, 4, 5

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 5

[14] Eric Kauderer-Abrams. Quantifying translation-invariance in convolutional neural networks. *arXiv preprint arXiv:1801.01450*, 2017. 4

[15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016. 1, 2, 3, 5

[16] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *ICLR*, 2017. 3

[17] Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. *Handbook of Brain Theory and Neural Networks*, 1995. 4

[18] 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*, 2018. 1, 2, 3, 4, 5

[19] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *ICLR*, 2017. 1, 2, 8

[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1

[21] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. In *ICLR*, 2017. 3

[22] Seyed Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *CVPR*, 2017. 3

[23] Tianyu Pang, Chao Du, Yinpeng Dong, and Jun Zhu. Towards robust detection of adversarial examples. In *NeurIPS*, 2018. 3

[24] Tianyu Pang, Chao Du, and Jun Zhu. Max-mahalanobis linear discriminant analysis networks. In *ICML*, 2018. 1

[25] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, 2017. 2

[26] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. In *ICLR*, 2018. 1

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 2

[28] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. In *ICLR*, 2018. 1

[29] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018. 1

[30] 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. 2, 5

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 1, 2, 5

[32] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014. 1, 2

[33] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *ICLR*, 2018. 1, 2, 3, 4, 5

[34] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may

be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018. 2

[35] Eric Wong and J. Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *ICML*, 2018. 1

[36] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. In *ICLR*, 2018. 1, 2, 3, 4, 5

[37] Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan Yuille. Improving transferability of adversarial examples with input diversity. *arXiv preprint arXiv:1803.06978*, 2018. 2, 3, 4, 5, 8

[38] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2