

# Enhancing the Transferability of Adversarial Attacks through Variance Tuning

Xiaosen Wang   Kun He \*

School of Computer Science and Technology, Huazhong University of Science and Technology

{xiaosen, brooklet60}@hust.edu.cn

## Abstract

Deep neural networks are vulnerable to adversarial examples that mislead the models with imperceptible perturbations. Though adversarial attacks have achieved incredible success rates in the white-box setting, most existing adversaries often exhibit weak transferability in the black-box setting, especially under the scenario of attacking models with defense mechanisms. In this work, we propose a new method called variance tuning to enhance the class of iterative gradient based attack methods and improve their attack transferability. Specifically, at each iteration for the gradient calculation, instead of directly using the current gradient for the momentum accumulation, we further consider the gradient variance of the previous iteration to tune the current gradient so as to stabilize the update direction and escape from poor local optima. Empirical results on the standard ImageNet dataset demonstrate that our method could significantly improve the transferability of gradient-based adversarial attacks. Besides, our method could be used to attack ensemble models or be integrated with various input transformations. Incorporating variance tuning with input transformations on iterative gradient-based attacks in the multi-model setting, the integrated method could achieve an average success rate of 90.1% against nine advanced defense methods, improving the current best attack performance significantly by 85.1%. Code is available at <https://github.com/JHL-HUST/VT>.

## 1. Introduction

Deep Neural Networks (DNNs) are known to be vulnerable to adversarial examples [31, 8], which are indistinguishable from legitimate ones by adding small perturbations, but lead to incorrect model prediction. In recent years, it has garnered an increasing interest to craft adversarial examples [22, 3, 14, 1, 16], because it not only can identify the model vulnerability [3, 1], but also can help improve the model robustness [8, 21, 32, 37]. Moreover, adversarial examples

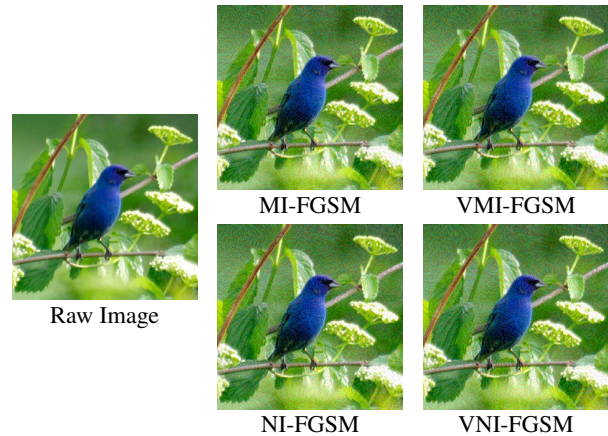


Figure 1: Adversarial examples crafted by MI-FGSM [6], NI-FGSM [18], the proposed VMI-FGSM and VNI-FGSM on the Inc-v3 model [30] with the maximum perturbation of  $\epsilon = 16$ . VMI-FGSM and VNI-FGSM generate visually similar adversaries as other attacks but lead to much higher transferability.

also exhibit good transferability across the models [25, 19], *i.e.* the adversaries crafted for one model can still fool other models, which enables black-box attacks in the real-world applications without any knowledge of the target model.

In the white-box setting that the attacker can access the architecture and parameters of the target model, existing adversarial attacks have exhibited great effectiveness [14, 3, 1] but with low transferability, especially for models equipped with defense mechanisms [32, 34, 17, 23]. To address this issue, recent works focus on improving the transferability of adversarial examples by advanced gradient calculation (*e.g.* Momentum, Nesterov’s accelerated gradient, *etc.*) [6, 18], attacking multiple models [19], or adopting various input transformations (*e.g.* random resizing and padding, translation, scale, admix, *etc.*) [35, 7, 18, 33]. However, there still exists a big gap between white-box attacks and transfer-based black-box attacks with regard to attack performance.

In this work, we propose a novel variance tuning iterative gradient-based method to enhance the transferability of the generated adversarial examples. Different from existing

\*Corresponding author.

gradient-based methods that perturb the input in the gradient direction of the loss function, or momentum iterative gradient-based methods that accumulate a velocity vector in the gradient direction, at each iteration our method additionally tunes the current gradient with the gradient variance in the neighborhood of the previous data point. The key idea is to reduce the variance of the gradient at each iteration so as to stabilize the update direction and escape from poor local optima during the search process. Empirical results on the standard ImageNet dataset demonstrate that, compared with state-of-the-art momentum-based adversarial attacks [6, 18], the proposed method could achieve significantly higher success rates for black-box models, meanwhile maintain similar success rates for white-box models. For instance, the proposed method improves the success rates of the momentum based attack [6] for more than 20% using adversarial examples generated on the Inc-v3 model [30]. The adversarial examples crafted by various attacks are illustrated in Figure 1.

To further demonstrate the effectiveness of our method, we combine variance tuning with several gradient-based attacks for ensemble models [19] and integrate these attacks with various input transformations [35, 7, 18]. Extensive experiments show that our integrated method could remarkably improve the attack transferability. In addition, we compare our attack method with the state-of-the-art attack methods [6, 35, 7, 18] against nine advanced defense methods [17, 34, 36, 9, 20, 12, 5, 23]. Our integrated method yields an average success rate of 67.0%, which outperforms the baselines by a large margin of 17.5% in the single model setting, and an average success rate of 90.1%, which outperforms the baselines by a clear margin of 6.6% in the multi-model setting.

## 2. Related Work

Let  $x$  be a benign image,  $y$  the corresponding true label and  $f(x; \theta)$  the classifier with parameters  $\theta$  that outputs the prediction result. Let  $J(x, y; \theta)$  denote the loss function of classifier  $f$  (e.g. the cross-entropy loss). We define the adversarial attack as finding an example  $x^{adv}$  that satisfies  $\|x - x^{adv}\|_p < \epsilon$  but misleads the model prediction,  $f(x; \theta) \neq f(x^{adv}; \theta)$ . Here  $\|\cdot\|_p$  denotes the  $p$ -norm distance and we focus on  $p = \infty$  to align with previous works.

### 2.1. Adversarial Attacks

Numerous adversarial attack methods have been proposed in recent years, including gradient-based methods [8, 14, 21, 6, 18], optimization-based methods [31, 3], score-based methods [11, 16] and decision-based methods [2, 4]. In this work, we mainly focus on the attack transferability and provide a brief overview on two branches of transfer-based attacks in this subsection.

#### 2.1.1 Gradient-based Attacks

The first branch focuses on improving the transferability of gradient-based attacks by advanced gradient calculation.

**Fast Gradient Sign Method (FGSM).** FGSM [8] is the first gradient-based attack that crafts an adversarial example  $x^{adv}$  by maximizing the loss function  $J(x^{adv}, y; \theta)$  with a one-step update:

$$x^{adv} = x + \epsilon \cdot \text{sign}(\nabla_x J(x, y; \theta)),$$

where  $\nabla_x J(x, y; \theta)$  is the gradient of loss function w.r.t.  $x$  and  $\text{sign}(\cdot)$  denotes the sign function.

**Iterative Fast Gradient Sign Method (I-FGSM).** I-FGSM [14] extends FGSM to an iterative version with a small step size  $\alpha$ :

$$\begin{aligned} x_{t+1}^{adv} &= x_t^{adv} + \alpha \cdot \text{sign}(\nabla_{x_t^{adv}} J(x_t^{adv}, y; \theta)), \\ x_0^{adv} &= x. \end{aligned} \quad (1)$$

**Momentum Iterative Fast Gradient Sign Method (MI-FGSM).** MI-FGSM [6] integrates the momentum into I-FGSM and achieves much higher transferability:

$$\begin{aligned} g_{t+1} &= \mu \cdot g_t + \frac{\nabla_{x_t^{adv}} J(x_t^{adv}, y; \theta)}{\|\nabla_{x_t^{adv}} J(x_t^{adv}, y; \theta)\|_1}, \\ x_{t+1}^{adv} &= x_t^{adv} + \alpha \cdot \text{sign}(g_{t+1}), \end{aligned} \quad (2)$$

where  $g_0 = 0$  and  $\mu$  is the decay factor.

**Nesterov Iterative Fast Gradient Sign Method (NI-FGSM).** NI-FGSM [18] adopts Nesterov's accelerated gradient [24], and substitutes  $x_t^{adv}$  in Eq. (2) with  $x_t^{adv} + \alpha \cdot \mu \cdot g_t$  to further improve the transferability of MI-FGSM.

#### 2.1.2 Input Transformations

The second branch focuses on adopting various input transformations to enhance the attack transferability.

**Diverse Input Method (DIM).** DIM [35] applies random resizing and padding to the inputs with a fixed probability, and feeds the transformed images into the classifier for the gradient calculation to improve the transferability.

**Translation-Invariant Method (TIM).** TIM [7] adopts a set of images to calculate the gradient, which performs well especially for black-box models with defense mechanisms. To reduce the calculation on gradients, Dong *et al.* [7] shift the image within small magnitude and approximately calculate the gradients by convolving the gradient of untranslated images with a kernel matrix.

**Scale-Invariant Method (SIM).** SIM [18] introduces the scale-invariant property and calculates the gradient over a set of images scaled by factor  $1/2^i$  on the input image to enhance the transferability of the generated adversarial examples, where  $i$  is a hyper-parameter.

Note that different input transformations, DIM, TIM and SIM, can be naturally integrated with gradient-based attack methods. Lin *et al.* [18] have shown that the combination of these methods, denoted as Composite Transformation Method (CTM), is the current strongest transfer-based black-box attack method. In this work, the proposed variance tuning based method aims to improve the transferability of the gradient-based attacks (*e.g.* MI-FGSM, NI-FGSM), and can be combined with various input transformations to further improve the attack transferability.

## 2.2. Adversarial Defenses

In contrast, to mitigate the threat of adversarial examples, various adversarial defense methods have been proposed. One promising defense method is called *adversarial training* [8, 15, 21] that injects the adversarial examples into the training data to improve the model robustness. Tramèr *et al.* [32] propose *ensemble adversarial training* by augmenting the training data with perturbations transferred from several models, which can further improve the robustness against transfer-based black-box attacks. However, such adversarial training methods, as one of the most powerful and extensively investigated defense methods, often result in high computation cost and are difficult to be scaled to large-scale datasets and complex neural networks [15].

Guo *et al.* [9] utilize a set of image transformations (*e.g.* JPEG compression, Total Variance Minimization, *etc.*) on the inputs to eliminate adversarial perturbations before feeding the images to the models. Xie *et al.* [34] adopt random resizing and padding (R&P) on the inputs to mitigate the adversarial effect. Liao *et al.* [17] propose to train a high-level representation denoiser (HGD) to purify the input images. Xu *et al.* [36] propose two feature squeezing methods: bit reduction (Bit-Red) and spatial smoothing to detect adversarial examples. Feature distillation (FD) [20] is a JPEG-based defensive compression framework against adversarial examples. ComDefend [12] is an end-to-end image compression model to defend adversarial examples. Cohen *et al.* [5] adopt randomized smoothing (RS) to train a certifiably robust ImageNet classifier. Naseer *et al.* [23] design a neural representation purifier (NRP) model that learns to purify the adversarially perturbed images based on the automatically derived supervision.

## 3. Methodology

For the methodology section, we first introduce our motivation, then provide a detailed description of the proposed method. In the end, we formulize the relationship between existing transfer-based attacks and the proposed method.

### 3.1. Motivation

Given a target classifier  $f$  with parameters  $\theta$  and a benign image  $x \in \mathcal{X}$  where  $x$  is in  $d$  dimensions and  $\mathcal{X}$  denotes all

the legitimate images, the adversarial attack aims to find an adversarial example  $x^{adv} \in \mathcal{X}$  that satisfies:

$$f(x; \theta) \neq f(x^{adv}; \theta) \quad \text{s.t.} \quad \|x - x^{adv}\| < \epsilon. \quad (3)$$

For white-box attacks, we can regard the attack as an optimization problem that searches an example in the neighborhood of  $x$  so as to maximize the loss function  $J$  of the target classifier  $f$ :

$$x^{adv} = \underset{\|x' - x\|_p < \epsilon}{\operatorname{argmax}} J(x', y; \theta). \quad (4)$$

Lin *et al.* [18] analogize the adversarial example generation process to the standard neural model training process, where the input  $x$  can be viewed as parameters to be trained and the target model can be treated as the training set. From this perspective, the transferability of adversarial examples is equivalent to the generalization of the normally trained models. Therefore, existing works mainly focus on better optimization algorithms (*e.g.* MI-FGSM, NI-FGSM) [14, 6, 18] or data augmentation (*e.g.* ensemble attack on multiple models or input transformations) [19, 35, 7, 18, 33] to improve the attack transferability.

In this work, we treat the iterative gradient-based adversarial attack as a stochastic gradient decent (SGD) optimization process, in which at each iteration, the attacker always chooses the target model for update. As presented in previous works [26, 28, 13], SGD introduces large variance due to the randomness, leading to slow convergence. To address this issue, various variance reduction methods have been proposed to accelerate the convergence of SGD, *e.g.* SAG (stochastic average gradient) [26], SDCA (stochastic dual coordinate ascent) [28], and SVRG (stochastic variance reduced gradient) [13], which adopt the information from the training set to reduce the variance. Moreover, Nesterov's accelerated gradient [24] that boosts the convergence, is beneficial to improve the attack transferability [18].

Based on above analysis, we attempt to enhance the adversarial transferability with the gradient variance tuning strategy. The major difference between our method and SGD with variance reduction methods (SGDVRLMs) [26, 28, 13] is three-fold. First, we aim to craft highly transferable adversaries, which is equivalent to improving the generalization of the training models, while SGDVRLMs aim to accelerate the convergence. Second, we consider the gradient variance of the examples sampled in the neighborhood of input  $x$ , which is equivalent to the one in the parameter space for training the neural models but SGDVRLMs utilize variance in the training set. Third, our variance tuning strategy is more generalized and can be used to improve the performance of MI-FGSM and NI-FGSM.

### 3.2. Variance Tuning Gradient-based Attacks

Typical gradient-based iterative attacks (*e.g.* I-FGSM) greedily search an adversarial example in the direction of

---

**Algorithm 1** VMI-FGSM

---

**Input:** A classifier  $f$  with parameters  $\theta$ , loss function  $J$   
**Input:** A raw example  $x$  with ground-truth label  $y$   
**Input:** The magnitude of perturbation  $\epsilon$ ; number of iteration  $T$  and decay factor  $\mu$   
**Input:** The factor  $\beta$  for the upper bound of neighborhood and number of example  $N$  for variance tuning  
**Output:** An adversarial example  $x^{adv}$

- 1:  $\alpha = \epsilon/T$
- 2:  $g_0 = 0; v_0 = 0; x_0^{adv} = x$
- 3: **for**  $t = 0 \rightarrow T - 1$  **do**
- 4:   Calculate the gradient  $\hat{g}_{t+1} = \nabla_{x_t^{adv}} J(x_t^{adv}, y; \theta)$
- 5:   Update  $g_{t+1}$  by variance tuning based momentum

$$g_{t+1} = \mu \cdot g_t + \frac{\hat{g}_{t+1} + v_t}{\|\hat{g}_{t+1} + v_t\|_1} \quad (5)$$

- 6:   Update  $v_{t+1} = V(x_t^{adv})$  by Eq. (7)
- 7:   Update  $x_{t+1}^{adv}$  by applying the sign of gradient

$$x_{t+1}^{adv} = x_t^{adv} + \alpha \cdot \text{sign}(g_{t+1}) \quad (6)$$

- 8: **end for**

- 9:  $x^{adv} = x_T^{adv}$

- 10: **return**  $x^{adv}$

---

the sign of the gradient at each iteration, as shown in Eq. (1), which may easily fall into poor local optima and “overfit” the model [6]. MI-FGSM [6] integrates momentum into I-FGSM for the purpose of stabilizing the update directions and escaping from poor local optima to improve the attack transferability. NI-FGSM [18] further adopts Netserov’s accelerated gradient [24] into I-FGSM to improve the transferability by leveraging its looking ahead property.

We observe that the above methods only consider the data points along the optimization path, denoted as  $x_0^{adv} = x, x_1^{adv}, \dots, x_{t-1}^{adv}, x_t^{adv}, \dots, x_T^{adv} = x^{adv}$ . In order to avoid overfitting and further improve the transferability of the adversarial attacks, we adopt the gradient information in the neighborhood of the previous data point to tune the gradient of the current data point at each iteration. Specifically, for any input  $x \in \mathcal{X}$ , we define the gradient variance as follows.

**Definition 1 Gradient Variance.** Given a classifier  $f$  with parameters  $\theta$  and loss function  $J(x, y; \theta)$ , an arbitrary image  $x \in \mathcal{X}$  and an upper bound  $\epsilon'$  for the neighborhood, the gradient variance can be defined as:

$$V_{\epsilon'}^g(x) = \mathbb{E}_{\|x' - x\|_p < \epsilon'} [\nabla_{x'} J(x', y; \theta)] - \nabla_x J(x, y; \theta).$$

We simply use  $V(x)$  to denote  $V_{\epsilon'}^g(x)$  without ambiguity in the following and we set  $\epsilon' = \beta \cdot \epsilon$  where  $\beta$  is a hyper-parameter and  $\epsilon$  is the upper bound of the perturbation mag-

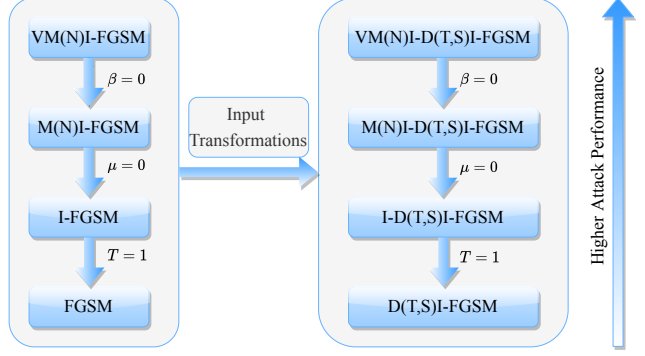


Figure 2: Relationships among various adversarial attacks. We can adjust the value of some hyper-parameters to relate the attacks derived from FGSM. We can also integrate various input transformations into these gradient-based attacks to further enhance the transferability. Here M(N)I-FGSM denotes MI-FGSM or NI-FGSM and D(T,S)I-FGSM denotes DI-FGSM, TI-FGSM or SI-FGSM.

nitude. In practice, however, due to the continuity of the input space, we cannot calculate  $\mathbb{E}_{\|x' - x\|_p < \epsilon'} [\nabla_{x'} J(x', y; \theta)]$  directly. Therefore, we approximate its value by sampling  $N$  examples in the neighborhood of  $x$  to calculate  $V(x)$ :

$$V(x) = \frac{1}{N} \sum_{i=1}^N \nabla_{x^i} J(x^i, y; \theta) - \nabla_x J(x, y; \theta). \quad (7)$$

Here  $x^i = x + r_i$ ,  $r_i \sim U[-(\beta \cdot \epsilon)^d, (\beta \cdot \epsilon)^d]$ , and  $U[a^d, b^d]$  stands for the uniform distribution in  $d$  dimensions.

After obtaining the gradient variance, we can tune the gradient of  $x_t^{adv}$  at the  $t$ -th iteration with the gradient variance  $V(x_{t-1}^{adv})$  at the  $(t - 1)$ -th iteration to stabilize the update direction. The algorithm of variance tuning MI-FGSM, denoted as VMI-FGSM, is summarized in Algorithm 1. Note that our method is generally applicable to any gradient-based attack method. We can easily extend VMI-FGSM to variance tuning NI-FGSM (VNI-FGSM), and integrate these methods with DIM, TIM and SIM as in [18].

### 3.3. Relationships among Various Attacks

This work focuses on the transferability of adversarial attacks derived from FGSM. Here we summarize the relationships among these attacks, as illustrated in Figure 2. If the factor  $\beta$  for the upper bound of neighborhood is set to 0, VMI-FGSM and VNI-FGSM degrade to MI-FGSM and NI-FGSM, respectively. If the decay factor  $\mu = 0$ , both MI-FGSM and NI-FGSM degrade to I-FGSM. If the iteration number  $T = 1$ , I-FGSM degrades to FGSM. Moreover, we can integrate the above attacks with various input transformations, *i.e.* DIM, TIM, SIM, to obtain more powerful adversarial attacks and these derived methods follow the same discipline.



Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-101	Inc-v3 <sub>ens3</sub>	Inc-v3 <sub>ens4</sub>	IncRes-v2 <sub>ens</sub>
Inc-v3	MI-FGSM	<b>100.0*</b>	43.6	42.4	35.7	13.1	12.8	6.2
	VMI-FGSM	<b>100.0*</b>	<b>71.7</b>	<b>68.1</b>	<b>60.2</b>	<b>32.8</b>	<b>31.2</b>	<b>17.5</b>
	NI-FGSM	<b>100.0*</b>	51.7	50.3	41.3	13.5	13.2	6.0
	VNI-FGSM	<b>100.0*</b>	<b>76.5</b>	<b>74.9</b>	<b>66.0</b>	<b>35.0</b>	<b>32.8</b>	<b>18.8</b>
Inc-v4	MI-FGSM	56.3	99.7*	46.6	41.0	16.3	14.8	7.5
	VMI-FGSM	<b>77.9</b>	<b>99.8*</b>	<b>71.2</b>	<b>62.2</b>	<b>38.2</b>	<b>38.7</b>	<b>23.2</b>
	NI-FGSM	63.1	<b>100.0*</b>	51.8	45.8	15.4	13.6	6.7
	VNI-FGSM	<b>83.4</b>	99.9*	<b>76.1</b>	<b>66.9</b>	<b>40.0</b>	<b>37.7</b>	<b>24.5</b>
IncRes-v2	MI-FGSM	60.7	51.1	<b>97.9*</b>	46.8	21.2	16.0	11.9
	VMI-FGSM	<b>77.9</b>	<b>72.1</b>	<b>97.9*</b>	<b>67.7</b>	<b>46.4</b>	<b>40.8</b>	<b>34.4</b>
	NI-FGSM	62.8	54.7	<b>99.1*</b>	46.0	20.0	15.1	9.6
	VNI-FGSM	<b>80.8</b>	<b>76.9</b>	98.5*	<b>69.8</b>	<b>47.9</b>	<b>40.3</b>	<b>34.2</b>
Res-101	MI-FGSM	58.1	51.6	50.5	<b>99.3*</b>	23.9	21.5	12.7
	VMI-FGSM	<b>75.1</b>	<b>68.9</b>	<b>70.5</b>	99.2*	<b>45.2</b>	<b>41.4</b>	<b>30.1</b>
	NI-FGSM	65.6	58.3	57.0	99.4*	24.5	21.4	11.7
	VNI-FGSM	<b>79.8</b>	<b>74.6</b>	<b>73.2</b>	<b>99.7*</b>	<b>46.1</b>	<b>42.5</b>	<b>32.1</b>

Table 1: The success rates (%) on seven models in the single model setting by various gradient-based iterative attacks. The adversarial examples are crafted on Inc-v3, Inc-v4, IncRes-v2, and Res-101 respectively. \* indicates the white-box model.

## 4. Experiments

To validate the effectiveness of the proposed variance tuning based attack method, we conduct extensive experiments on standard ImageNet dataset [27]. In this section, we first specify the experimental setup, then we compare our method with competitive baselines under various experimental settings and quantify the attack effectiveness on nine advanced defense models. Experimental results demonstrate that our method can significantly improve the transferability of the baselines in various settings. Finally, we provide further investigation on hyper-parameters  $N$  and  $\beta$  used for variance tuning.

### 4.1. Experimental Setup

**Dataset.** We randomly pick 1,000 clean images pertaining to the 1,000 categories from the ILSVRC 2012 validation set [27], which are almost correctly classified by all the testing models as in [6, 18].

**Models.** We consider four normally trained networks, including Inception-v3 (Inc-v3) [30], Inception-v4 (Inc-v4), Inception-Resnet-v2 (IncRes-v2) [29] and Resnet-v2-101 (Res-101) [10] and three adversarially trained models, namely Inc-v3<sub>ens3</sub>, Inc-v3<sub>ens4</sub> and IncRes-v2<sub>ens</sub> [32]. Besides, we include nine advanced defense models that are robust against black-box adversarial attacks on the ImageNet dataset, *i.e.* HGD [17], R&P [34], NIPS-r3<sup>1</sup>, Bit-Red [36], JPEG [9], FD [20], ComDefend [12], RS [5] and NRP [23]. For HGD, R&P, NIPS-r3 and RS, we adopt the official models provided in corresponding papers. For all the other de-

fense methods, we adopt Inc-v3<sub>ens3</sub> as the target model.

**Baselines.** We take two popular momentum based iterative adversarial attacks as our baselines, *i.e.* MI-FGSM [6] and NI-FGSM [18], that exhibit better transferability than other white-box attacks [8, 14, 3]. In addition, we integrate the proposed method with various input transformations, *i.e.* DIM [35], TIM [7], SIM and CTM [18], denoted as VM(N)I-DI-FGSM, VM(N)I-TI-FGSM, VM(N)I-SI-FGSM and VM(N)I-CT-FGSM respectively, to further validate the effectiveness of our method.

**Hyper-parameters.** We follow the attack setting in [6] with the maximum perturbation of  $\epsilon = 16$ , number of iteration  $T = 10$  and step size  $\alpha = 1.6$ . For MI-FGSM and NI-FGSM, we set the decay factor  $\mu = 1.0$ . For DIM, the transformation probability is set to 0.5. For TIM, we adopt the Gaussian kernel with kernel size  $7 \times 7$ . For SIM, the number of scale copies is 5 (*i.e.*  $i = 0, 1, 2, 3, 4$ ). For the proposed method, we set  $N = 20$  and  $\beta = 1.5$ .

### 4.2. Attack a Single Model

We first perform four adversarial attacks, namely MI-FGSM, NI-FGSM, the proposed variance tuning based methods VMI-FGSM and VNI-FGSM, on a single neural network. We craft adversarial examples on normally trained networks and test them on all the seven neural networks we consider. The *success rates*, which are the misclassification rates of the corresponding models on adversarial examples, are shown in Table 1. The models we attack are on rows and the seven models we test are on columns.

We can observe that VMI-FGSM and VNI-FGSM outperform the baseline attacks by a large margin on all the black-box models, while maintain high success rates on

<sup>1</sup><https://github.com/anlthms/nips-2017/tree/master/mmd>

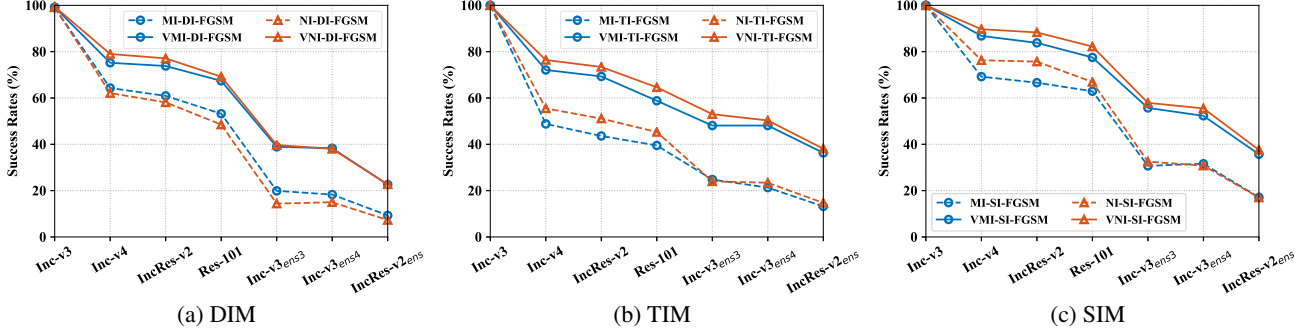


Figure 3: The success rates (%) on seven models in the single model setting by various gradient-based iterative attacks enhanced by DIM, TIM and SIM respectively. The adversarial examples are generated on Inc-v3 model.

Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-101	Inc-v3 <sub>ens3</sub>	Inc-v3 <sub>ens4</sub>	IncRes-v2 <sub>ens</sub>
Inc-v3	MI-CT-FGSM	98.7*	85.4	80.6	76.0	64.1	62.1	45.2
	VMI-CT-FGSM	<b>99.3*</b>	<b>88.6</b>	<b>86.7</b>	<b>82.9</b>	<b>78.6</b>	<b>76.2</b>	<b>64.7</b>
	NI-CT-FGSM	98.9*	84.1	80.0	74.5	60.0	56.2	41.0
	VNI-CT-FGSM	<b>99.5*</b>	<b>91.2</b>	<b>89.0</b>	<b>85.3</b>	<b>78.6</b>	<b>76.7</b>	<b>65.3</b>
Inc-v4	MI-CT-FGSM	87.2	98.6*	83.3	78.3	72.2	67.2	57.3
	VMI-CT-FGSM	<b>90.0</b>	<b>98.8*</b>	<b>86.6</b>	<b>81.9</b>	<b>78.3</b>	<b>76.6</b>	<b>68.3</b>
	NI-CT-FGSM	87.8	<b>99.4*</b>	82.5	75.9	65.8	62.6	51.3
	VNI-CT-FGSM	<b>92.1</b>	99.2*	<b>89.2</b>	<b>85.1</b>	<b>80.1</b>	<b>78.3</b>	<b>70.4</b>
IncRes-v2	MI-CT-FGSM	87.9	85.7	<b>97.1*</b>	83.0	77.6	74.6	72.0
	VMI-CT-FGSM	<b>88.9</b>	<b>87.0</b>	97.0*	<b>85.0</b>	<b>83.4</b>	<b>80.5</b>	<b>79.4</b>
	NI-CT-FGSM	90.2	87.0	<b>99.4*</b>	83.2	75.0	68.9	65.1
	VNI-CT-FGSM	<b>92.9</b>	<b>90.6</b>	99.0*	<b>88.2</b>	<b>85.2</b>	<b>82.5</b>	<b>81.8</b>
Res-101	MI-CT-FGSM	86.5	81.8	83.2	<b>98.9*</b>	77.0	72.3	61.9
	VMI-CT-FGSM	<b>86.9</b>	<b>84.2</b>	<b>86.4</b>	98.6*	<b>81.0</b>	<b>78.6</b>	<b>71.6</b>
	NI-CT-FGSM	86.1	82.2	83.3	98.5*	70.0	68.5	54.6
	VNI-CT-FGSM	<b>90.7</b>	<b>85.5</b>	<b>87.2</b>	<b>99.1*</b>	<b>82.6</b>	<b>79.7</b>	<b>73.3</b>

Table 2: The success rates (%) on seven models in the single model setting by various gradient-based iterative attacks enhanced by CTM. \* indicates the white-box model.

all the white-box models. For instance, if we craft adversarial examples on Inc-v3 model in which all the attacks can achieve 100% success rates in the white-box setting, VMI-FGSM yields 71.7% success rate on Inc-v4 and 32.8% success rate on Inc-v3<sub>ens3</sub>, while the baseline MI-FGSM only obtains the corresponding success rates of 43.6% and 13.1%, respectively. This convincingly validates the high effectiveness of the proposed method. We also illustrate several adversarial images generated on Inc-v3 model by various attack methods in Appendix A, showing that these generated adversarial perturbations are all human imperceptible but our method leads to higher transferability.

### 4.3. Attack with Input Transformations

Several input transformations, *e.g.* DIM, [7], TIM [35] and SIM [18], have been incorporated into the gradient-based adversarial attacks, which are effective to improve the transferability. Here we integrate our methods into these input transformations and demonstrate the proposed vari-

ance tuning strategy could further enhance the transferability. We report the success rates of black-box attacks in Figure 3, where the adversarial examples are generated on Inc-v3 model. The results for adversarial examples generated on other three models are reported in Appendix B.

The results show that the success rates against black-box models are improved by a large margin with the variance tuning strategy regardless of the attack algorithms or the white-box models to be attacked. In general, the methods equipped with our variance tuning strategy consistently outperform the baseline attacks by 10% ~ 30%.

Lin *et al.* [18] have shown that CTM, the combination of DIM, TIM and SIM, could help the gradient-based attacks achieve great transferability. We also combine CTM with our method to further improve the transferability. As depicted in Table 2, the success rates could be further improved remarkably on various models, especially against adversarially trained models, which further demonstrates the high effectiveness and generalization of our method.

Attack	Inc-v3	Inc-v4	IncRes-v2	Res-101	Inc-v3 <sub>ens3</sub>	Inc-v3 <sub>ens4</sub>	IncRes-v2 <sub>ens</sub>
MI-FGSM	<b>99.9*</b>	98.2*	95.3*	<b>99.9*</b>	39.4	35.3	24.2
VMI-FGSM	99.7*	<b>98.5*</b>	<b>96.0*</b>	<b>99.9*</b>	<b>67.6</b>	<b>62.9</b>	<b>50.7</b>
NI-FGSM	99.8*	<b>99.8*</b>	<b>98.9*</b>	99.8*	41.0	33.5	23.1
VNI-FGSM	<b>99.9*</b>	99.6*	98.6*	<b>99.9*</b>	<b>71.3</b>	<b>66.0</b>	<b>52.9</b>
MI-CT-FGSM	99.6*	99.1*	97.4*	99.7*	91.3	89.6	86.8
VMI-CT-FGSM	<b>99.7*</b>	<b>99.2*</b>	<b>98.4*</b>	<b>99.9*</b>	<b>93.6</b>	<b>92.4</b>	<b>91.0</b>
NI-CT-FGSM	<b>100.0*</b>	<b>100.0*</b>	<b>100.0*</b>	<b>100.0*</b>	92.8	89.6	83.6
VNI-CT-FGSM	<b>100.0*</b>	99.9*	99.6*	<b>100.0*</b>	<b>95.5</b>	<b>94.5</b>	<b>92.3</b>

Table 3: The success rates (%) on seven models in the multi-model setting by various gradient-based iterative attacks. The adversarial examples are generated on the ensemble models, *i.e.* Inc-v3, Inc-v4, IncRes-v2 and Res-101.

Model	Attack	HGD	R&P	NIPS-r3	Bit-Red	JPEG	FD	ComDefend	RS	NRP	Average
Inc-v3	MI-CT-FGSM	56.6	44.9	52.5	36.2	77.3	60.0	80.1	40.3	29.3	53.0
	VMI-CT-FGSM	<b>73.1</b>	<b>65.1</b>	<b>70.3</b>	<b>49.5</b>	<b>85.4</b>	<b>72.4</b>	<b>86.0</b>	<b>51.9</b>	<b>45.2</b>	<b>66.5</b>
	NI-CT-FGSM	50.4	39.4	47.4	34.3	76.0	58.6	77.7	36.9	24.8	49.5
	VNI-CT-FGSM	<b>73.4</b>	<b>64.5</b>	<b>70.6</b>	<b>51.2</b>	<b>86.8</b>	<b>73.5</b>	<b>87.3</b>	<b>52.1</b>	<b>43.9</b>	<b>67.0</b>
Ens	MI-CT-FGSM	91.0	87.7	89.0	75.9	94.2	88.8	95.1	68.1	76.1	85.1
	VMI-CT-FGSM	<b>92.9</b>	<b>91.0</b>	<b>92.3</b>	<b>80.9</b>	<b>95.4</b>	<b>91.0</b>	<b>96.2</b>	<b>77.0</b>	<b>83.2</b>	<b>88.9</b>
	NI-CT-FGSM	91.3	85.6	89.0	72.3	95.9	89.5	95.4	63.2	69.5	83.5
	VNI-CT-FGSM	<b>94.7</b>	<b>92.4</b>	<b>93.4</b>	<b>82.3</b>	<b>97.1</b>	<b>92.6</b>	<b>97.4</b>	<b>77.4</b>	<b>84.0</b>	<b>90.1</b>

Table 4: The success rates (%) on nine models with advanced defense mechanism by various gradient-based iterative attacks enhanced by CTM. The adversarial examples are generated on Inc-v3 model and the ensemble of models respectively.

#### 4.4. Attack an Ensemble of Models

Liu *et al.* [19] have shown that attacking multiple models simultaneously could improve the transferability of the generated adversarial examples. In this subsection, we adopt the ensemble attack method in [6], which fuses the logit outputs of different models, to demonstrate that our variance tuning method could further improve the transferability of adversarial attacks in the multi-model setting. Specifically, we attack the ensemble of four normally trained models, *i.e.* Inc-v3, Inc-v4, IncRes-v2 and Res-101 by averaging the logit outputs of the models using various attacks with or without input transformations.

As shown in Table 3, our methods (VMI-FGSM, VNI-FGSM) could significantly enhance the transferability of the baselines more than 25% for MI-FGSM and 30% for NI-FGSM on the adversarially trained models. Even though the attacks with CTM could achieve good enough transferability, our methods (VMI-CT-FGSM, VNI-CT-FGSM) could further improve the transferability significantly. In particular, VNI-CT-FGSM achieves the success rates of 92.3%  $\sim$  95.5% against three adversarially trained models, indicating the vulnerability of current defense mechanisms. Besides, in the white-box setting, our methods could still maintain similar success rates as the baselines.

#### 4.5. Attack Advanced Defense Models

To further validate the effectiveness of the proposed method in practice, except for the normally trained mod-

els and adversarially trained models, we also evaluate our methods on nine extra models with advanced defenses, including the top-3 defense methods in the NIPS competition (HGD (rank-1) [17], R&P (rank-2) [34] and NIPS-r3 (rank-3)), and six recently proposed defense methods, namely Bit-Red [36], JPEG [9], FD [20], ComDefend [12], RS [5] and NRP [23].

Since MI-CT-FGSM and NI-CT-FGSM exhibit the best transferability among the existing attack methods [18], we compare our methods with the two attacks with adversarial examples crafted on Inc-v3 model and the ensemble models as in Section 4.4, respectively. The results are shown in Table 4. In the single model setting, the proposed methods achieve an average success rate of 66.5% for VMI-CT-FGSM and 67.0% for VNI-CT-FGSM, which outperforms the baseline attacks for more than 13.5% and 17.5% respectively. In the multi-model setting, our methods achieve an average success rate of 88.9% for VMI-CT-FGSM and 90.1% for VNI-CT-FGSM, which outperforms the baseline attacks for more than 3.8% and 6.6% respectively. Note that in the multi-model setting, our methods achieve the success rates of more than 77% against the defense model with Randomize Smoothing (RS) [5] that provides certified defense. And our methods achieve the success rates of more than 83% against the defense model with Neural Representation Purifier (NRP) which is a recently proposed powerful defense method and exhibits great robustness against DIM and DI-TIM [23], raising a new security issue for the development of more robust deep learning models.

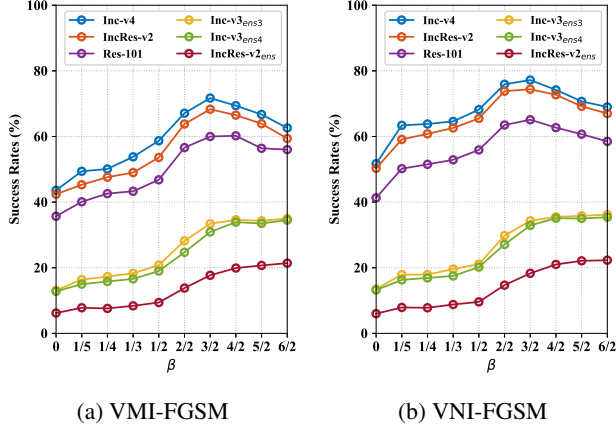


Figure 4: The success rates (%) on the other six models with adversarial examples generated by VMI-FGSM and VNI-FGSM on Inc-v3 model when varying factor  $\beta$  for the upper bound of the neighborhood.

#### 4.6. Ablation Study on Hyper-parameters

We conduct a series of ablation experiments to study the impact of two hyper-parameters of the proposed variance tuning,  $N$  and  $\beta$ . All the adversarial examples are generated on Inc-v3 model without input transformations, which achieve success rates of 100% for different values of the two parameters in the white-box setting.

**On the upper bound of neighborhood.** In Figure 4, we study the influence of the neighborhood size, determined by parameter  $\beta$ , on the success rates in the black-box setting where  $N$  is fixed to 20. When  $\beta = 0$ , VMI-FGSM and VNI-FGSM degrade to MI-FGSM and NI-FGSM, respectively, and achieve the lowest transferability. When  $\beta = 1/5$ , although the neighborhood is very small, our variance tuning strategy could improve the transferability remarkably. As we increase  $\beta$ , the transferability increases and achieves the peak for normally trained models when  $\beta = 3/2$  but it is still increasing against the adversarially trained models. In order to achieve the trade-off for the transferability on normally trained models and adversarially trained models, we choose  $\beta = 3/2$  in our experiments.

**On the number of sampled examples in the neighborhood.** We then study the influence of the number of sampled examples  $N$  on the success rates in the black-box setting ( $\beta$  is fixed to  $3/2$ ). As depicted in Figure 5, when  $N = 0$ , VMI-FGSM and VNI-FGSM also degrade to MI-FGSM and NI-FGSM, respectively, and achieve the lowest transferability. When  $N = 20$ , the transferability of adversarial examples is improved significantly and the transferability increases slowly when we continually increase  $N$ . Note that we need  $N$  forward and backward propagations for gradient variance at each iteration as shown in Eq. 7, thus a bigger value of  $N$  means a higher computation cost.

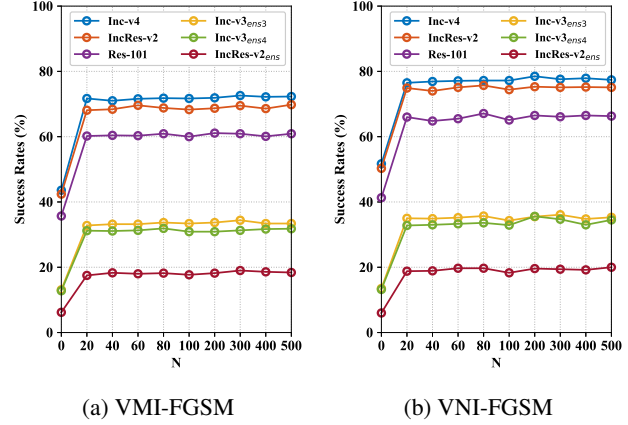


Figure 5: The success rates (%) on the other six models with adversarial examples generated by VMI-FGSM and VNI-FGSM on Inc-v3 model when varying the number of sampled example  $N$ .

To balance the transferability and computation cost, we set  $N = 20$  in our experiments.

In summary,  $\beta$  plays a key role in improving the transferability while  $N$  exhibits little impact when  $N > 20$ . In our experiments, we set  $\beta = 3/2$  and  $N = 20$ .

## 5. Conclusion

In this work, we propose a variance tuning method to enhance the transferability of the iterative gradient-based adversarial attacks. Specifically, for any input of a neural classifier, we define its gradient variance as the difference between the mean gradient of the neighborhood and its own gradient. Then we adopt the gradient variance of the data point at the previous iteration along the optimization path to tune the current gradient. Extensive experiments demonstrate that the variance tuning method could significantly improve the transferability of the existing competitive attacks, MI-FGSM and NI-FGSM, while maintain similar success rates in the white-box setting.

The variance tuning method is generally applicable to any iterative gradient-based attacks. We employ our method to attack ensemble models and then integrate with advanced input transformation methods (e.g. DIM, TIM, SIM) to further enhance the transferability. Empirical results on nine advanced defense models show that our integrated method could reach an average success rate of at least 90.1%, outperforming the state-of-the-art attacks for 6.6% on average, indicating the insufficiency of current defense techniques.

## Acknowledgements

This work is supported by National Natural Science Foundation (62076105) and Microsoft Research Asia Collaborative Research Fund (99245180).



## References

- [1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *International Conference on Machine Learning*, 2018. 1
- [2] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *International Conference on Learning Representations*, 2018. 2
- [3] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017. 1, 2, 5
- [4] Weilun Chen, Zhaoxiang Zhang, Xiaolin Hu, and Baoyuan Wu. Boosting decision-based black-box adversarial attacks with random sign flip. In *Proceedings of the European Conference on Computer Vision*, 2020. 2
- [5] Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. *International Conference on Machine Learning*, 2019. 2, 3, 5, 7
- [6] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018. 1, 2, 3, 4, 5, 7, 12
- [7] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4312–4321, 2019. 1, 2, 3, 5, 6
- [8] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations*, 2015. 1, 2, 3, 5
- [9] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *International Conference on Learning Representations*, 2018. 2, 3, 5, 7
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5
- [11] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *International Conference on Machine Learning*, 2018. 2
- [12] Xiaojun Jia, Xingxing Wei, Xiaochun Cao, and Hassan Foroosh. Comendefend: An efficient image compression model to defend adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6084–6092, 2019. 2, 3, 5, 7
- [13] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013. 3
- [14] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *International Conference on Learning Representations, Workshop Track Proceedings*, 2017. 1, 2, 3, 5
- [15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *International Conference on Learning Representations*, 2017. 3
- [16] Yandong Li, Lijun Li, Liqiang Wang, Tong Zhang, and Boqing Gong. Nattack: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks. *International Conference on Machine Learning*, 2019. 1, 2
- [17] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018. 1, 2, 3, 5, 7
- [18] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *International Conference on Learning Representations*, 2020. 1, 2, 3, 4, 5, 6, 7, 12
- [19] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *International Conference on Learning Representations*, 2017. 1, 2, 3, 7
- [20] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 860–868. IEEE, 2019. 2, 3, 5, 7
- [21] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations*, 2018. 1, 2, 3
- [22] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016. 1
- [23] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 262–271, 2020. 1, 2, 3, 5, 7
- [24] Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence  $O(1/k^2)$ . *Doklady AN USSR*, 269:543–547, 1983. 2, 3, 4
- [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*, pages 506–519, 2017. 1
- [26] Nicolas L Roux, Mark Schmidt, and Francis R Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in neural information processing systems*, pages 2663–2671, 2012. 3
- [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. [5](#)
- [28] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013. [3](#)
- [29] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *AAAI Conference on Artificial Intelligence*, 2017. [5](#)
- [30] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. [1](#), [2](#), [5](#), [12](#)
- [31] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations*, 2014. [1](#), [2](#)
- [32] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *International Conference on Learning Representations*, 2018. [1](#), [3](#), [5](#)
- [33] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. *arXiv preprint arXiv:2102.00436*, 2021. [1](#), [3](#)
- [34] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. *International Conference on Learning Representations*, 2018. [1](#), [2](#), [3](#), [5](#), [7](#)
- [35] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019. [1](#), [2](#), [3](#), [5](#), [6](#)
- [36] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *Network and Distributed System Security Symposium*, 2018. [2](#), [3](#), [5](#), [7](#)
- [37] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. *International Conference on Machine Learning*, 2019. [1](#)