# Strong Transferable Adversarial Attacks via Ensembled Asymptotically Normal Distribution Learning

Zhengwei Fang[1,2], Rui Wang[1,2,3]*, Tao Huang[1,2], Liping Jing[1,2]
[1]School of Computer Science and Technology, Beijing Jiaotong University, Beijing, China
[2]Beijing Key Lab of Traffic Data Analysis and Mining, Beijing, China
[3]Collaborative Innovation Center of Railway Traffic Safety, Beijing, China
jankinfmail@gmail.com, {rui.wang,thuang,lpjing}@bjtu.edu.cn

## Abstract

*Strong adversarial examples are crucial for evaluating and enhancing the robustness of deep neural networks. However, the performance of popular attacks is usually sensitive, for instance, to minor image transformations, stemming from limited information — typically only one input example, a handful of white-box source models, and undefined defense strategies. Hence, the crafted adversarial examples are prone to overfit the source model, which hampers their transferability to unknown architectures. In this paper, we propose an approach named Multiple Asymptotically Normal Distribution Attacks (MultiANDA) which explicitly characterize adversarial perturbations from a learned distribution. Specifically, we approximate the posterior distribution over the perturbations by taking advantage of the asymptotic normality property of stochastic gradient ascent (SGA), then employ the deep ensemble strategy as an effective proxy for Bayesian marginalization in this process, aiming to estimate a mixture of Gaussians that facilitates a more thorough exploration of the potential optimization space. The approximated posterior essentially describes the stationary distribution of SGA iterations, which captures the geometric information around the local optimum. Thus, MultiANDA allows drawing an unlimited number of adversarial perturbations for each input and reliably maintains the transferability. Our proposed method outperforms ten state-of-the-art black-box attacks on deep learning models with or without defenses through extensive experiments on seven normally trained and seven defense models.*

## 1. Introduction

Albeit the excellent performance of deep neural networks (DNNs) across a broad spectrum of tasks in machine vision and natural language processing, a rich stream of work [8, 14, 39, 50, 51] shows the non-negligible vulnerability of DNNs to adversarial attacks, which craft imperceptible perturbations to synthesize malicious inputs for triggering unexpected model predictions. Such severe degradation of model robustness and stability greatly hinders the application of DNNs in security- or safety-critical domains [1, 2, 11]. Fortunately, studying how to create potent adversarial examples aids in evaluating DNN-based systems and enhancing their robustness by devising defensive strategies, such as adversarial training [14, 29].

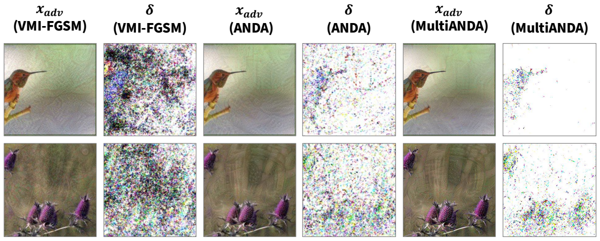


Figure 1. Perturbation visualization of the adversarial examples generated by ANDA/MultiANDA that fool all selected normally trained (first row) and defense (second row) models.

This paper focuses on the black-box and transfer-based threat model. The corresponding adversarial attack algorithms intend to synthesize the adversarial examples that can successfully fool the unknown DNNs without accessing any information on model architectures and outputs. Researchers implement such attacks by maximizing the non-concave loss function of a substitute model trained for the same task with, for example, one step [14], or iterative steps [20] of gradient ascent optimization. However, prior work has shown the generated adversarial examples trend to overfit the substitute model, which hardly attacks other models [21]. Meanwhile, they are often sensitive to minor transformations [4, 26, 27].

Hence, researchers have been exploring various data

[*] Corresponding author.

augmentation strategies [4, 9, 25, 49] and improved SGA optimization methods [8, 17, 25, 45] to enhance the performance of transferable attacks. Nonetheless, these efforts exhibit limited generalization capabilities on unknown DNN architectures, particularly those trained with defense strategies. This suggests that a deterministic optimization procedure, initiated from a single image, is insufficient to thoroughly explore the high-dimensional perturbation space. Furthermore, the projected gradient descent (PGD) method proposed by Madry *et al*. [29] initializes the attack algorithm with random restarts around the legitimate image. However, these random samples represent homogeneous deviations from the original input, lacking diversity [10], which limits the transferability of adversarial examples.

To resolve these issues, in this paper, we propose **Multi**ple **A**symptotically **N**ormal **D**istribution **A**ttacks (**MultiANDA**), a novel method that explicitly characterizes perturbations inferred from a learned distribution. Firstly, we formulate a single ANDA by taking advantage of the nice property of stochastic gradient ascent (SGA), the asymptotic normality [5, 18, 28, 30], to learn the true optimal posterior distribution of adversarial perturbations. Specifically, ANDA calculates and stores the first two moments of iterations captured along the optimizing trajectory to approximate the distribution of the adversarial perturbation as a Gaussian one. Moreover, we apply the deep ensemble strategy [22] on ANDA to devise a mixture of Gaussians, approximating the Bayesian posterior. As each Gaussians positioned at a different basin of attraction, the proposed MultiANDA leads to a further improved generalization performance of the attacks with more diverse adversarial examples. As shown in Figure 1, the perturbations crafted by our proposed method focus more on the semantic and decisive areas of objects than the baseline approach. In particular, the contributions of this work are the following:

- We propose MultiANDA, an approximate Bayesian inference method to realize transfer-based adversarial attacks. We learn the perturbation distribution instead of finding a specific adversary for each input example. Thus, a good approximation of the true perturbation posterior boosts the generalization (i.e., transferability) of the generated adversarial examples.
- We devise a stochastic adversarial attacking process instead of the original deterministic one initialized from one input by random image translations. In this way, we profit the asymptotic normality of SGA for a better perturbation distribution learning. Adopting the deep ensemble strategy, an effective mechanism for approximating Bayesian marginalization [47], MultiANDA accurately and efficiently estimate the posterior distribution over the perturbations by using the geometry information along the attacking trajectory.
- The learned distributions of perturbation allow drawing an unlimited number of adversarial examples for one input. We experimentally validate that the sampled adversaries have comparably high attack success rates with the singly generated ones, even on the defense models. This result demonstrates the great potential of our method for designing defense strategies grounded in adversarial training.
- Extensive experiment results show that the proposed method outperforms ten state-of-the-art black-box attacks on deep learning models with or without advanced defenses. Significantly, MultiANDA is more advantageous when attacking the advanced defense models. Thus, ANDA and MultiANDA are expected to benchmark new defense methods in the future. Our code is available at https://github.com/CLIAgroup/ANDA.

## 2. Related Work and Preliminaries

Our study mainly focuses on the black-box and transfer-based threat model, as they may bring more security and safety risks due to their practical applicability [8, 34].

### 2.1. Black-Box Adversarial Attacks

Essentially, the adversarial attack algorithms intend to maximize the loss function $\mathcal{L}(\cdot)$, e.g., the cross entropy loss for classification tasks, to search for the adversaries $x_{adv}$ [14, 39], with the constraints of $l_\infty$ norm bound $\varepsilon$, as shown in (1):

$$\max_{x_{adv}} \mathcal{L}(x_{adv}, y) \quad \text{s.t.}\ ||x_{adv} - x||_\infty \leq \varepsilon, \tag{1}$$

where $x$, $x_{adv}$, $y$ are the legitimate input, the adversarial example and the true label, respectively.

Among prior work, the fast gradient sign method (FGSM) [14] and its iterative version, the basic iterative method (BIM) [20] resort to linearizing the non-concave loss function and realize the attack with one-step update or iterative updates, as shown in (2):

$$x_{adv}^{(t+1)} = \Phi(x_{adv}^{(t)} + \alpha \cdot \text{sign}(\underbrace{\nabla_x \mathcal{L}(x_{adv}^{(t)}, y)}_{\delta^{(t)}})), \tag{2}$$

where $x_{adv}^{(t+1)}$, $\delta^{(t)}$ are the adversarial example and perturbation at the $t$-th step, $x_{adv}^{(0)}$ is the legitimate input, and $\Phi(\cdot)$ is the projection function on the $\varepsilon$-ball.

These attacks are more effective under white-box scenarios, whereas they tend to overfit the model parameters and hardly transfer to unknown architectures [21]. Thus, many works have been proposed to improve the transferability of adversarial examples, which can be divided into three categories. The first group focuses on the iterative optimizing process by considering momentum [8], Nesterov accelerated gradient [25], and variance reducing [45] to search for the optimal solutions. The second group attempts to attack the mid-layers by disrupting the critical object-aware features [12, 16, 31, 46, 52]

Moreover, the third stream of methods adopt the commonly used generalization techniques by introducing diversity. These techniques encompass data augmentation [15, 19], such as resize and padding [49], translation [9]; and other integrated augmentation strategies, including combining multiple linear transformations [4], modifying the pixel values with multiple scales [25], integrating more information from other images [7], and using the sum of the gradients from intensity augmented images [17].

Similar to our study, existing attack approaches also try to determine the explicit adversarial distributions. PGD [29] resolves the inner maximization problem by the random restart strategy. However, the uniformly drawn samples tend to lie densely around the natural image [10]. Li *et al*. [23] propose $\mathcal{N}$attack to define and learn an adversarial distribution. However, their work is under a query-based threat model, i.e., it has to access the output scores for each input from the black-box defense model, which weakens its scalability in real-world applications. Additionally, Dong et al. [10] propose to formulate the adversarial perturbations via a learned distribution for adversarial distributional training (ADT). Similar to $\mathcal{N}$attack, ADT crafts the adversarial examples centered with the natural input $x$, which may fail in generating adversarial examples once the input is a little bit far away from the boundary. Conversely, ANDA generates adversaries around the optimal $x^*_{adv}$. In this case, the adversaries are more prone to successful attacks. From the facets of implementation, ADT focuses on learning a distribution from a large amount of training data for adversarial training. Our proposed method tries to collect the trajectory information during the optimization process for each natural input. Thus, these crucial differences make two approaches suitable for distinct tasks: inner maximization problem for adversarial training (ADT) and transfer-based black-box attack (ANDA and MultiANDA).

### 2.2. Asymptotic Normality of SGD

Maddox et al. [28] point out that the trajectory of stochastic gradient descent (SGD) can be regarded as the Bayesian posterior distribution over the random variables to be optimized, e.g., the model parameters in the case of the DNN training. Following this theory, they propose a Gaussian posterior approximation method (SWAG) based on the first two moments of SGD iterations for model calibration or uncertainty quantification. In the prior work, Mandt et al. [30] provide a more detailed analysis of the stationary distribution of SGD iterations with a constant learning rate, also termed the asymptotic normality of SGD [44]. Such analysis can be traced back to the work by Ruppert [36], Polyak and Juditsky [35] known as Polyak-Ruppert averaging, and still an active research subject for SGD [5]. Furthermore, inspired by deep ensembles [22], Wilson and Izmailov [47] carried out an extensive empirical study, demonstrating that applying SWAG only with multiple randomly initialized models can significantly improve the model performance. They emphasize that this benefits from the multimodal effects of the mixture Gaussian distributions and, in turn, boosts the contribution of each additional sample in the optimization procedure. In this paper, we adopt these theories to innovatively model the posterior distribution of perturbations to search for potent adversarial examples.

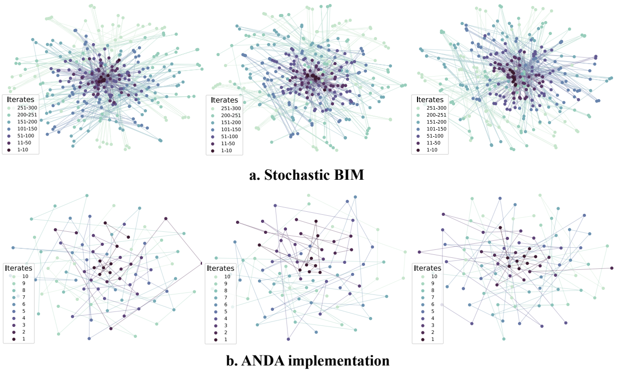
## 3. Proposed Method

Drawing on the comprehensive research previously discussed, we hypothesize that utilizing the asymptotic normality of SGD (SGA in our problem formulation) to marginalize an approximate Bayesian posterior could enhance generalization (transferability in our context) beyond what is achievable with a fixed set of parameter (one specific adversarial perturbation). However, optimizing the objective (1) using attack algorithms such as BIM (2) is a deterministic gradient ascent process. In this process, no gradient noise is introduced to form a stationary distribution of SGD. Thus, to bridge this gap, we exploit random data augmentation in the standard optimization process, devising a novel attack method for accurately and efficiently estimating the posterior distribution over perturbations.

In Section 3.1, we first empirically analyze the asymptotically normal distribution property in the stochastic adversarial attack process, then proceed to formalize the optimization objective for the task of adversarial example generation. Finally, inspired by the advantageous properties of SGD and the effective multimodal inference of the ensemble strategy, we introduce the new attack methods ANDA and MultiANDA in Sections 3.2 and 3.3, respectively.

### 3.1. Asymptotic Normality of Stochastic BIM

Under specific conditions—decaying learning rates, smooth gradients, and a full rank stationary distribution—it has been theoretically established that SGD asymptotically converges to a Gaussian distribution centered at the local optimum [3, 5, 28]. Expanding on this, Mandt et al. [30] demonstrated that under certain assumptions, SGD with a constant and sufficiently small learning rate in its final search phase, i.e., when updates are proximal to the local optimum, converges to a stationary distribution. This convergence implies that iterations during this phase can approximate the posterior distribution of the optimized variables. Crucially, the gradient noise, introduced by the randomness of the sampled training batches, is instrumental in forming this distribution, provided the learning rate is adequately small [28]. These theoretical underpinnings motivate our approach in the algorithm design.



a. Stochastic BIM

b. ANDA implementation

Figure 2. Asymptotically normal distribution examples with iterative trajectories of adversarial perturbations generated by stochastic BIM (a) and the proposed approach (b).

As demonstrated in (2), the optimization process, beginning from a specific input, is a deterministic gradient ascent, differing from a standard SGA approach. Therefore, we introduce stochastic gradient noises to emulate Gaussian limiting distributions by integrating a random data augmentation technique AUG, like image translation used in our experiments, at each iterative step:

$$x_{adv}^{(t+1)} = \Phi(x_{adv}^{(t)} + \alpha \cdot \mathrm{sign}(\underbrace{\nabla_x \mathcal{L}(\mathtt{AUG}(x_{adv}^{(t)}), y)}_{\delta^{(t)}})). \quad (3)$$

Notably, a small constant learning rate schedule ($\alpha = \epsilon/T$) is commonly adopted, where $\epsilon \in \mathbb{R}$ is a sufficiently small number used to restrict the $L_\infty$-norm for creating invisible adversarial perturbations, and $T$ is the number of iterative steps. Hence, we speculate the devised stochastic BIM shown in (3) can be used as an approximate Bayesian inference algorithm. In particular, Mandt et al. [30] assume that the gradients along the optimizing trajectory follow the Gaussian distribution stemming from the central limit theorem. We visualized the adversarial perturbations $\delta^{(t)}$—the loss gradients in our task—using the t-SNE technique [43] to examine their distribution. 300 iterative steps for 3 randomly-drawn ImageNet [37] images shown in Fig. 2 **a** illustrates the Gaussian-shaped distributions which underpins our conjecture of the asymptotically normality of stochastic BIM algorithm. We notice the obvious increased gradient noise along the optimizing trajectory, which may due to the use of the $\mathrm{sign}()$ function and projection operations. Nevertheless, this does not affect the convergence of perturbations. Importantly, the first 10 steps converging with tight covariance justifies another speculation that the small value of $T$ (often =10 for non-targeted attacks) and the non-trivial first step in our task (e.g., referring to strong attack performance of FGSM) indicate that the early iterates of are fairly close to the local optimum (i.e., the final search phase discussed above).

Inspired by the results of this empirical study, we propose employing $n$ image transformations $\mathtt{AUG}_i(\cdot)$ to form a minibatch for an adversarial attack, thereby further enhancing stochasticity. Then, the optimization objective is formulated as

$$\max_{x_{adv}} \sum_{i=1}^{n} \mathcal{L}(\mathtt{AUG}_i(x_{adv}), y) \quad \text{s.t. } ||x_{adv} - x||_\infty \leq \varepsilon. \quad (4)$$

Furthermore, optimizing (4) to craft just one optimal adversarial example does not sufficiently explore the high-dimensional potential perturbation space. To further improve the transferability of adversarial examples to unknown architectures, we propose to explicitly model the adversarial perturbation ($\delta$) via a distribution ($\Pi_\delta$). Thus, our optimization objective is formalized as

$$\max_{\Pi_\delta} \mathbb{E}_{\delta \sim \Pi_\delta} \sum_{i=1}^{n} \mathcal{L}(\mathtt{AUG}_i(x + \delta), y), \text{s.t. } ||\delta||_\infty \leq \varepsilon. \quad (5)$$

Objective (5) aims to maximize the expected loss over the adversarial perturbation distribution. It is noteworthy that (5) is a generalized version of (4) and is expected to characterize sufficient information about perturbations for transferable adversarial example generation.

## 3.2. Asymptotically Normal Distribution Attack

To better solve the optimization problem formulated in (5), we take advantage of the asymptotically normality of SGA and iteratively search for the adversarial examples similarly to (2). We omit the projection function $\Phi(\cdot)$ and $\mathrm{sign}(\cdot)$ because they do not effect the asymptotic normality of SGA as discussed in the previous subsection. Then, we calculate the gradient of the adversary with each transformation and denote it as the augmentation-aware perturbation $\delta_i^{(t)}$:

$$\delta_i^{(t)} = \nabla_{x^{(t)}} \mathcal{L}(\mathtt{AUG}_i(x_{adv}^{(t)}), y). \quad (6)$$

Let $\mathcal{S}$ be a set of samples augmented in each iteration, i.e., $\mathcal{S} = \{\mathtt{AUG}_i(x_{adv}^{(t)})\}_{i=1}^n$. Here $\mathcal{S}$ can be taken as a minibatch to form a stochastic gradient $\hat{\delta}_S$ which is a sum of contributions from all samples in $\mathcal{S}$. Hence, referring to the central limit theorem, $\hat{\delta}_S$ can be taken as a function of sample $z \in \mathcal{S}$,

$$\hat{\delta}_S(z) \approx \delta(z) + \frac{1}{\sqrt{n}} \Delta\delta(z), \quad (7)$$

where the gradient noises $\Delta\delta(z)$ at each update can be conveniently assumed following a Gaussian distribution with covariance $C(z)$,

$$\Delta\delta(z) \sim \mathcal{N}(0, C(z)), \quad \hat{\delta}_S(z) \sim \mathcal{N}(\delta(z), \frac{1}{n}C(z)). \quad (8)$$

**Algorithm 1:** Asymptotically Normal Distribution Attack (ANDA)

```
Input: $x$: clean image $x \in \mathbb{R}^d$; $y$: ground-truth label; $f$: pre-trained source model;
Parameters: $T$: # iterations; $\epsilon$: perturbation magnitude; $n$: # batch samples for augmentation; $M$: # sampling examples;
Output: $x_{adv}$: Adversarial examples;
Initialize $\alpha \leftarrow \epsilon/T$, $\bar{\delta}^{(0)} \leftarrow \mathbf{0}^d$, $\mathbf{D} \leftarrow \emptyset$, $x_0 \leftarrow x$
for $t = 0$ to $T - 1$ do
    Calculate the mean of gradients and store the deviation of gradients throughout previous $t$ iterations:
        $\delta_i^{(t)} = \nabla_{x^{(t)}} J(f(\mathtt{AUG}_i(x^{(t)})), y), i = 1, \ldots, n$
        $\bar{\delta}^{(t+1)} = \frac{(t \times n)\bar{\delta}^{(t)} + \sum_{i=1}^{n} \delta_i^{(t)}}{(t+1) \times n}$
        $\mathtt{APPEND\_COLS}(\mathbf{D}, \{\delta_i^{(t)} - \bar{\delta}^{(t+1)}\}), i = 1, \ldots, n$
    Update the adversarial example along the average direction:
        $x^{(t+1)} = \mathtt{Clip}_{x,\epsilon}\{x^{(t)} + \alpha \cdot \mathtt{Sign}(\bar{\delta}^{(t+1)})\}$
end for
Output option (a): Craft one $x_{adv}$
    $x_{adv} = x^{(T)}$
Output option (b): Craft M $x_{adv}$ by sampling from the learned perturbation distribution $\mathcal{N}(\bar{\delta}, \sigma)$
    $C = \mathtt{NUMS\_COLS}(\mathbf{D}) = n \times T$
    $\bar{\delta} = \bar{\delta}^{(T)}, \quad \sigma = \frac{\mathbf{D}\mathbf{D}^T}{C - 1}$
Generate adversarial examples
    $\{\delta_m\}_{m=1}^{M} \sim \mathcal{N}\left(\bar{\delta}, \sigma\right),$
    $\{x_{adv}^m\}_{m=1}^{M} = \{\mathtt{Clip}_{x,\epsilon}\{x^{(T-1)} + \alpha \cdot \mathtt{Sign}(\delta_m)\}\}_{m=1}^{M}$
```

In this case, the expectation of the stochastic gradient is the full gradient, i.e., $\delta(z) = \mathbb{E}[\hat{\delta}_S(z)]$. Hence, the prior distribution over adversarial perturbations is naturally a Gaussian one at each iteration. Benefiting from the asymptotic normality of SGD [30], we propose a novel method to approximate the posterior distribution of perturbations. Our main idea is to estimate the mean and covariance matrix of the stationary distribution of stochastic gradient ascent during the optimizing procedure. The iterative averaging is adopted to approximate the mean of adversarial perturbations using the sequence of stochastic gradients

$$\bar{\delta}^{(t+1)} = \frac{(t \times n)\bar{\delta}^{(t)} + \sum_{i=1}^{n} \delta_i^{(t)}}{(t+1) \times n}. \qquad (9)$$

Then, $\bar{\delta} = \bar{\delta}^{(T)}$, where $n$ is the augmented samples in the $t$-th iteration ($t = 0, \cdots, T-1$) and $T$ is the total iterations.

Let $\mathbf{D}$ be the deviation matrix with column $\mathbf{D}_j = \delta_i^{(t)} - \bar{\delta}^{(t+1)}$, where $j = t \times n + i$ and $i = 1, \cdots, n$. Thus, the covariance matrix of the posterior distribution is estimated by considering all stochastic gradients in iterative processing,

$$\sigma = \frac{\mathbf{D}\mathbf{D}^T}{n \times T - 1}. \qquad (10)$$

Finally, the adversarial example can be iteratively obtained by (3). Note that, ANDA allows to craft one or an unlimited number of adversarial examples for one input by sampling perturbations from the estimated distribution $\mathcal{N}\left(\bar{\delta}, \sigma\right)$. The complete procedure for the proposed ANDA method is shown in Algorithm 1. Following the empirical study in Section 3.1, we visualized the perturbations $\delta_i^{(t)}$ of 10 iterates with multiple data augmentations (shown in Fig. 2 **b**), which validate our previous hypothesis.

### 3.3. Multiple Implementations of ANDA

With the proposed ANDA, we find a solution to approximating the true posterior distribution centered at the the optimal $x_{adv}^*$ for the maximization problem introduced in (2). Nevertheless, only exploring the unimodal optimization space largely limits the the diversity of generated adversarial examples. Profiting the excellent generalization capability of Bayesian marginalization approximated with the deep ensemble strategy, we propose MultiANDA via multiple (e.g., K times) repeats of ANDA by add random initial values on origin sample $x$. Especially, we average the adversarial perturbation $\bar{\delta}_k$ from each ANDA process, and obtain $\bar{\delta}_{mean} = \frac{1}{K}\sum_{k=0}^{K-1}\bar{\delta}_k$. This realizes the multimodal marginalization of this Gaussian mixture distribution. Similarly, adversarial examples can be iteratively obtained by (2). MultiANDA also enables producing infinite adversarial examples with high effectiveness by sampling from each approximated Gaussian distribution $\mathcal{N}\left(\bar{\delta}_k, \sigma_k\right)$. Remarkable improvements of the attacking ability, especially on the defense models are shown in Sections 4.2 and 4.3.

## 4. Experiments

In this section, we present the results of an extensive empirical study to validate the performance of the proposed methods. We first specify the experimental settings in Section 4.1. Then, the results of ANDA and MultiANDA compared with the state-of-the-art approaches on various types of DNNs with or without defenses are presented in Section 4.2 and 4.3. Section 4.4 shows the attacking effectiveness of the multiple adversarial examples generated from each input via the proposed methods. Due to page limitations, more detailed experiment settings, results and the ablation study are presented in Appendix.

| Attack | Inc-v3 $\Longrightarrow$ | | | | | | ResNet-50 $\Longrightarrow$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Inc-v3 | ResNet-50 | ResNet-152 | IncRes-v2 | VGG-19 | Avg. | Inc-v3 | ResNet-50 | ResNet-152 | IncRes-v2 | VGG-19 | Avg. |
| BIM | **100.0*** | 20.3 | 15.7 | 15.6 | 34.3 | 21.5 | 33.2 | 99.7* | 63.9 | 20.9 | 43.4 | 40.4 |
| TIM | 64.3* | 35.9 | 30.6 | 25.4 | 70.4 | 45.3 | 47.3 | 77.0* | 47.6 | 30.8 | 68.8 | 54.3 |
| SIM | **100.0*** | 38.2 | 31.1 | 35.9 | 42.2 | 36.9 | 48.6 | **100.0*** | 83.7 | 35.3 | 50.9 | 54.6 |
| DIM | **100.0*** | 31.7 | 25.5 | 31.4 | 45.5 | 33.5 | 61.9 | 99.9* | 83.6 | 47.5 | 61.1 | 63.5 |
| FIA | 98.3* | 78.4 | 75.3 | 81.2 | **83.5** | **79.6** | 86.2 | 99.6* | 94.5 | 80.4 | 88.3 | 87.4 |
| TAIG | 99.7* | 53.3 | 45.9 | 56.7 | 54.2 | 52.5 | 62.4 | **100.0*** | 86.1 | 51.9 | 58.7 | 64.8 |
| NI-FGSM | **100.0*** | 40.0 | 35.2 | 39.9 | 56.9 | 43.0 | 59.6 | 99.7* | 85.4 | 48.1 | 67.0 | 65.0 |
| MI-FGSM | **100.0*** | 40.2 | 35.1 | 40.3 | 57.1 | 43.2 | 58.5 | 99.7* | 85.9 | 48.6 | 67.4 | 65.1 |
| VMI-FGSM | **100.0*** | 63.0 | 59.3 | 68.6 | 70.3 | 65.3 | 75.3 | 99.9* | 93.4 | 68.3 | 76.4 | 78.4 |
| VNI-FGSM | **100.0*** | 62.4 | 58.7 | 67.7 | 69.7 | 64.6 | 75.4 | 99.8* | 92.9 | 67.9 | 75.6 | 78.0 |
| ANDA | **100.0*** | 76.1 | 72.8 | 82.3 | 77.0 | 77.1 | 95.6 | **100.0*** | 98.9 | 94.0 | 89.5 | 94.5 |
| MultiANDA | **100.0*** | **79.2** | **76.0** | **84.5** | 78.8 | **79.6** | **96.5** | **100.0*** | **99.2** | **95.0** | **90.1** | **95.2** |
| Attack | IncRes-v2 $\Longrightarrow$ | | | | | | VGG-19 $\Longrightarrow$ | | | | | |
| | Inc-v3 | ResNet-50 | ResNet-152 | IncRes-v2 | VGG-19 | Avg. | Inc-v3 | ResNet-50 | ResNet-152 | IncRes-v2 | VGG-19 | Avg. |
| BIM | 36.3 | 25.6 | 20.6 | 99.3* | 37.8 | 30.1 | 23.5 | 18.7 | 13.7 | 9.9 | 99.9* | 16.5 |
| TIM | 43.4 | 36.5 | 32.0 | 36.0* | 66.2 | 42.8 | 41.5 | 34.8 | 30.2 | 23.6 | **100.0*** | 46.0 |
| SIM | 58.9 | 47.8 | 41.4 | 99.6* | 49.8 | 49.5 | 37.7 | 34.4 | 25.2 | 23.6 | **100.0*** | 30.2 |
| DIM | 54.6 | 41.4 | 36.6 | 98.2* | 50.0 | 45.7 | 31.7 | 26.4 | 19.2 | 16.4 | 99.9* | 23.4 |
| FIA | 82.2 | 75.3 | 72.4 | 89.2* | 80.7 | 77.7 | 57.4 | 50.7 | 40.9 | 42.7 | **100.0*** | 47.9 |
| TAIG | 73.9 | 63.4 | 58.4 | 95.0* | 57.4 | 63.3 | 48.8 | 43.6 | 34.7 | 33.9 | **100.0*** | 40.3 |
| NI-FGSM | 61.9 | 49.5 | 44.7 | 99.2* | 64.7 | 55.2 | 43.6 | 39.5 | 29.2 | 30.4 | 99.9* | 35.7 |
| MI-FGSM | 60.3 | 49.3 | 43.0 | 98.8* | 64.6 | 54.3 | 44.7 | 39.4 | 28.9 | 30.8 | 99.9* | 36.0 |
| VMI-FGSM | 81.1 | 69.6 | 66.4 | 99.3* | 73.5 | 72.7 | 62.7 | 56.7 | 46.5 | 48.6 | **100.0*** | 53.6 |
| VNI-FGSM | 80.9 | 70.0 | 65.8 | 99.4* | 73.8 | 72.6 | 63.2 | 56.7 | 46.6 | 48.8 | **100.0*** | 53.8 |
| ANDA | 93.0 | 86.4 | 83.7 | **99.8*** | 82.8 | 86.5 | 74.4 | 64.1 | 56.4 | 61.5 | **100.0*** | 64.1 |
| MultiANDA | **93.9** | **87.1** | **85.6** | **99.8*** | **84.3** | **87.7** | **75.4** | **66.1** | **58.6** | **63.5** | **100.0*** | **65.9** |

Table 1. The success rates (%) of the proposed and baseline attacks on five normally trained models. Sign * indicates the results on white-box source models. The best/second results are shown in bold/underlined. Due to the space limit, performance on five target models are presented here. See full results on seven targets in Appendix.

## 4.1. Experimental Setup

The datasets, models, and baseline methods employed in our experiments are detailed below.

**Datasets:** Following the prior work [8, 13, 46], we implemented our experiments on ImageNet1k dataset [33] containing 1000 images, which are randomly drawn from the ImageNet dataset [37].

**Models:** We considered two categories of target models for evaluation: seven *normally trained models* and seven *advanced defense models*. The first category includes Inception-v3 (Inc-v3) [40], Resnet-v2-50 (ResNet-50), Resnet-v2-101 (ResNet-101), Resnet-v2-152 (ResNet-152) [15], Inception-v4 (Inc-v4), Inception-ResNet-v2 (IncRes-v2) [41] and VGG-19 [38], where Inc-v3, ResNet-50, IncRes-v2 and VGG-19 are also used as the white-box source models to generate adversarial examples. The defense models contain three adversarially trained models: Inc-v3$_{ens3}$, Inc-v3$_{ens4}$ and IncRes-v2$_{ens}$ [42]; the High-level representation Guided Denoiser (HGD) [24]; the Neural Representation Purifier (NRP) [32]; the Randomized Smoothing (RS) [6]; and the 'Rand-3' submission in the NIPS 2017 defense competition (NIPS-r3).

**Baselines:** Three streams of transfer-based attack methods are used as baseline methods. The first stream focuses on the data augmentations including DIM [49], TIM [9], SIM [45] and a recent work Transferable Attack based on Integrated Gradients (TAIG) [17]. For the second group, we choose the feature importance-aware (FIA)[46], which considers the middle-layer information of the source model and achieves the competitive performance. The last stream focuses on the optimization enhanced methods, including BIM [20], its accelerated version: MI-FGSM [8], NI-FGSM [25], and the approaches using variance reduction strategy, VMI-FGSM, VNI-FGSM [45].

## 4.2. Attack Normally Trained Models

We cross-validated our method by generating and testing the adversarial examples on the selected source (on rows) and target (on columns) networks (shown in Table 1). To be a fair comparison, our proposed methods generated one adversarial example for each input, aligned with the first output option of Algorithm 1, in the following experiments if not specified. From the Table 1, we observe that ANDA and MultiANDA significantly outperforms other methods in almost all cases, no matter in white-box or black-box settings. As the numerical results demonstrate, they consistently defeat all other methods when taking IncRes-v2, ResNet-50 and VGG-19 as source models, and for Inc-v3, ANDA is competitive with the current SOTA methods, and MultiANDA always keep higher success rate.

We further investigated the adversarial examples generated with ResNet-50, shown in Figure 3. The other six models were used as black-box targets for evaluating the performance of these examples. A higher number of successfully deceived target models implies more capable adversaries and therefore stronger attacks. The rightmost bars shows that ANDA, MultiANDA generated 840, 863/1000 adversaries that fooled all six targeted models, respectively, while 517/1000 are for VMI-FGSM. Naturally, the remaining sets of bars for this baseline method are higher, meaning that it deceived fewer targeted models. Moreover, we visualized the generated examples and the perturbations by these three methods (see Figure 1). The results show that the perturbations crafted by ANDA and MultiANDA focus more on the semantic areas of objects than VMI-FGSM, which dominates the decision of the prediction model. More visualization results are provided in Appendix.

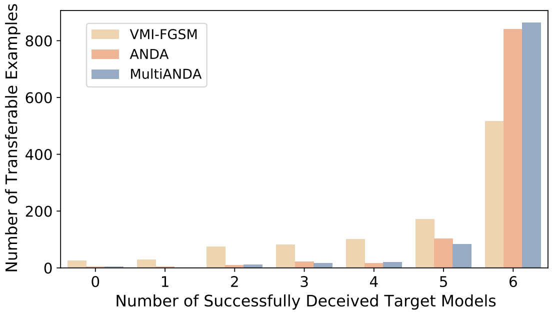


Figure 3. Most examples crafted by our proposed methods successfully deceived all 6 black-box models.

### 4.3. Attack Defense Models

We evaluate our proposed method targeting on the selected advanced defenses models. As shown in Table 2, the success rates of our proposed methods consistently improve over these various defense models in most cases. They provides a significant performance increase in general compared with the other attacks, especially reaches the overall optimal success rates with the source model ResNet-50. An intriguing finding is that using ResNet-50 as the source model always yields the best attacking performance. This aligns with the conclusion drawn by Wu *et al*. [48] that the skip connections in ResNet-like neural networks enhance black-box attack transferability.

Notably, the fluctuating performance and the generally reduced effectiveness of all baseline methods can be observed with the source model VGG-19. The average maximum attack success rate in this context is only 28.3%, suggesting that the feature extraction capabilities of VGG-19 are not optimal, thereby leading to diminished transferability for black-box attacks. This observation highlights the critical role of source model selection in maximizing the transferability of adversarial attacks. Hence, the choice of source models and enhancing their capability for generalized feature extraction become promising directions for the future research.

Similarly, we visualized the generated examples and the perturbations by VMI-FGSM, ANDA and MultiANDA that fool the defense models. The perturbed area by our proposed methods concentrates more on the informative region of images than the baseline method. Details are illustrated in Appendix. These compelling results of ANDA and MultiANDA on defenses valid their overall generalization ability in generating transferable and diverse adversaries.

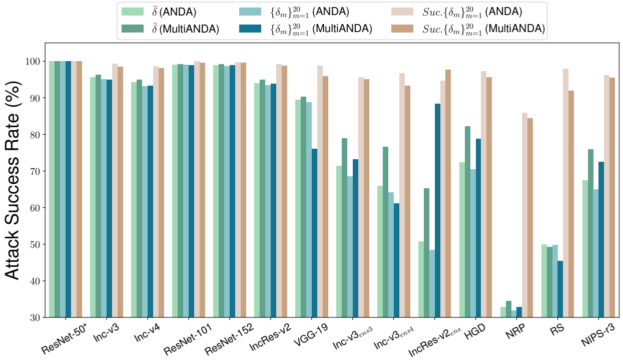


Figure 4. Attack success rates of sampled adversaries with single $\{\bar{\delta}\}$, multiple $\{\delta_m\}_{m=1}^{20}$ and the succeeded $Suc.\{\delta_m\}_{m=1}^{20}$ generated by ANDA and MultiANDA. $\{\delta_m\}_{m=1}^{20}$ means the 20 adversarial examples sampled from the learned perturbation distribution for each input image; whereas $Suc.\{\delta_m\}_{m=1}^{20}$ are the 20 sampled adversarial examples for the input image whose corresponding single adversarial example has succeeded in the attack.

### 4.4. Attack Performance of Sampled Adversaries

The analysis presented above shows the effectiveness of the single optimum adversarial example per input generated, denoted as $\bar{\delta}$, referring to *Step 6 (a)* shown in Algorithm 1. Besides, MultiANDA/ANDA allows learning the distribution of the perturbations $\mathcal{N}\left(\bar{\delta}, \sigma\right)$ via *Step 6 (b)*, and generate any number of adversarial examples for each input by sampling $M$ adversaries, denoted as $\{\delta_m\}_{m=1}^{M}$. To verify the attack performance of these sampled adversaries, we drawn 20 ($M = 20$) adversarial examples for each input image, i.e., $\{\delta_m\}_{m=1}^{20}$. Meanwhile, we analyzed the adversaries sampled for the input images whose corresponding adversaries $\bar{\delta}$ succeeded in the attack, denoted as $Suc.\{\delta_m\}_{m=1}^{20}$. Experiment results show in Figure 4 that the sampled adversaries have competitive attack success rates compared with the single adversary generated with $\bar{\delta}$,

| Attack | Inc-v3 $\Longrightarrow$ | | | | | | ResNet-50 $\Longrightarrow$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Inc-v3$_{ens3}$ | IncRes-v2$_{ens}$ | HGD | NRP | NIPS-r3 | Avg. | Inc-v3$_{ens3}$ | IncRes-v2$_{ens}$ | HGD | NRP | NIPS-r3 | Avg. |
| BIM | 11.1 | 4.6 | 3.7 | 13.4 | 4.8 | 7.5 | 12.3 | 7.1 | 7.4 | 13.7 | 8.1 | 9.7 |
| TIM | 27.5 | 21.3 | 16.9 | 22.8 | 21.1 | 21.9 | 34.0 | 27.4 | 24.2 | 31.1 | 30.6 | 29.5 |
| SIM | 18.1 | 8.4 | 8.6 | 15.2 | 10.8 | 12.2 | 21.4 | 13.1 | 15.2 | 16.1 | 15.1 | 16.2 |
| DIM | 13.1 | 6.7 | 5.8 | 12.8 | 8.6 | 9.4 | 20.8 | 12.0 | 16.9 | 15.7 | 15.3 | 16.1 |
| FIA | 37.4 | 21.3 | 11.6 | 23.5 | 29.2 | 24.6 | 44.4 | 27.2 | 30.0 | 25.9 | 35.3 | 32.6 |
| TAIG | 38.0 | 23.9 | 22.8 | **29.6** | 28.5 | $\underline{28.6}$ | 39.2 | 30.4 | 29.7 | 29.6 | 34.4 | 32.7 |
| MI-FGSM | 18.3 | 9.0 | 5.5 | 15.7 | 12.0 | 12.1 | 26.3 | 15.5 | 17.3 | 18.0 | 20.2 | 19.5 |
| NI-FGSM | 18.6 | 8.6 | 6.2 | 15.3 | 12.2 | 12.2 | 26.2 | 15.7 | 17.5 | 18.9 | 19.8 | 19.6 |
| VMI-FGSM | 36.9 | 21.2 | 19.1 | 24.7 | 27.7 | 25.9 | 47.4 | 31.3 | 37.7 | 27.1 | 38.7 | 36.4 |
| VNI-FGSM | 36.4 | 22.0 | 18.9 | $\underline{25.3}$ | 27.4 | 27.1 | 46.5 | 30.4 | 37.4 | 27.1 | 38.7 | 36.4 |
| ANDA | $\underline{44.4}$ | $\underline{25.9}$ | $\underline{36.5}$ | 23.2 | $\underline{37.0}$ | 20.3 | $\underline{71.4}$ | $\underline{50.7}$ | $\underline{72.4}$ | $\underline{32.9}$ | $\underline{67.4}$ | $\underline{41.6}$ |
| MultiANDA | **54.4** | **36.7** | **52.8** | 24.3 | **46.9** | **29.9** | **79.7** | **64.8** | **82.3** | **34.4** | **76.3** | **50.1** |
| Attack | IncRes-v2 $\Longrightarrow$ | | | | | | VGG-19 $\Longrightarrow$ | | | | | |
| | Inc-v3$_{ens3}$ | IncRes-v2$_{ens}$ | HGD | NRP | NIPS-r3 | Avg. | Inc-v3$_{ens3}$ | IncRes-v2$_{ens}$ | HGD | NRP | NIPS-r3 | Avg. |
| BIM | 11.1 | 7.0 | 5.1 | 13.0 | 6.0 | 8.4 | 9.6 | 4.4 | 3.9 | 13.4 | 4.6 | 7.2 |
| TIM | 27.9 | 21.4 | 18.4 | 23.9 | 23.8 | 23.1 | $\underline{23.0}$ | 27.4 | 15.0 | 31.1 | 30.6 | **25.4** |
| SIM | 23.8 | 16.4 | 15.9 | 18.3 | 18.1 | 18.5 | 11.4 | 5.6 | 8.7 | 13.4 | 7.4 | 9.3 |
| DIM | 16.2 | 10.1 | 11.7 | 14.9 | 12.3 | 13.0 | 10.7 | 4.7 | 5.6 | 13.6 | 5.9 | 8.1 |
| FIA | 48.9 | 34.7 | 24.2 | $\underline{30.8}$ | 42.8 | 36.3 | 13.3 | 8.3 | 7.1 | 14.6 | 9.8 | 10.6 |
| TAIG | 49.0 | 41.2 | 12.2 | 16.5 | 13.3 | 26.4 | 17.7 | 10.2 | **36.9** | **34.1** | **41.9** | 28.2 |
| MI-FGSM | 22.0 | 13.3 | 13.4 | 16.4 | 17.0 | 16.4 | 13.0 | 6.9 | 7.2 | 15.5 | 8.9 | 10.3 |
| NI-FGSM | 21.6 | 13.8 | 13.4 | 15.5 | 16.6 | 16.2 | 13.9 | 6.8 | 7.2 | 14.6 | 9.3 | 10.4 |
| VMI-FGSM | 49.2 | 38.8 | 36.0 | 25.2 | 39.2 | 37.7 | 21.9 | 11.6 | 15.9 | 18.4 | 15.9 | 16.7 |
| VNI-FGSM | 49.9 | 37.7 | 35.4 | 25.6 | 39.3 | $\underline{38.4}$ | 21.6 | $\underline{11.9}$ | 16.8 | $\underline{18.6}$ | 15.7 | $\underline{22.9}$ |
| ANDA | $\underline{63.3}$ | $\underline{47.3}$ | $\underline{57.8}$ | 28.5 | $\underline{57.9}$ | 32.6 | 20.9 | 10.8 | 22.2 | 15.4 | 16.2 | 3.7 |
| MultiANDA | **70.3** | **61.9** | **69.9** | **31.2** | **67.8** | **41.8** | **24.6** | **13.7** | $\underline{31.2}$ | 16.2 | $\underline{21.3}$ | 7.9 |

Table 2. The success rates (%) of the proposed and baseline attacks on five defense models. The best/second results are shown in bold/underlined. Due to the space limit, performance on five target models are presented here. See full results on seven targets in Appendix.

even on the defense models. This confirms the well approximation of the perturbation distributions and the great potential of our method for adversarial training or defense model designing.

## 5. Conclusions and Perspectives

In this work, we present the Multiple Asymptotically Normal Distribution Attacks (MultiANDA) to generate transferable adversarial examples. Our primary goal is to efficiently approximate the posterior distribution of adversarial perturbations to achieve better generalization across unknown deep learning models. We leverage the statistical information of gradients along the optimization trajectory to estimate the stationary distribution of the perturbations. With this learned distribution's aid, MultiANDA can significantly improve the transferability of adversarial examples to black-box target models. Extensive experiments demonstrate that MultiANDA outperforms popular and recently proposed black-box and transfer-based methods, further highlighting the insufficiency of current defense techniques. Notably, the computational overhead of both ANDA and MultiANDA, largely depending on the number of batch samples during augmentation, is on par with similar baseline methods (refer to Appendix for the computational overhead analysis). Compared to the potential risks these algorithms might pose if misused, their computational cost is relatively negligible. We advocate for the development of advanced deep learning models which align better with human visual preferences, thereby avoiding the fundamental defects present in modern models.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant 62176020, the Joint Foundation of the Ministry of Education (8091B042235), the Beijing Natural Science Foundation under Grant L211016, the Talent Fund of Beijing Jiaotong University under Grant 2023XKRC041, and the Fundamental Research Funds for the Central Universities (Science and technology leading talent team project) under Grant 2022JBXT003. We also thank Huafeng Liu and Zhekun Liu for their helpful advice and support.

## References

[1] Erin E Alves, Devesh Bhatt, Brendan Hall, Kevin Driscoll, Anitha Murugesan, and John Rushby. Considerations in assuring safety of increasingly autonomous systems. Technical report, 2018. NASA technical report. 1

[2] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Chris-

tiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016. 1

[3] Søren Asmussen and Peter W Glynn. *Stochastic simulation: algorithms and analysis*. Springer Science & Business Media, 2007. 3

[4] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International Conference on Machine Learning*, pages 284–293. PMLR, 2018. 1, 2, 3

[5] Xi Chen, Jason D Lee, Xin T Tong, and Yichen Zhang. Statistical inference for model parameters in stochastic gradient descent. *arXiv preprint arXiv:1610.08637*, 2016. 2, 3

[6] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019. 6

[7] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Admix: Enhancing the transferability of adversarial attacks. In *International Conference on Computer Vision*, 2021. 3

[8] 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, 6

[9] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4312–4321, 2019. 2, 3, 6

[10] Yinpeng Dong, Zhijie Deng, Tianyu Pang, Jun Zhu, and Hang Su. Adversarial distributional training for robust deep learning. *Advances in Neural Information Processing Systems*, 33:8270–8283, 2020. 2, 3

[11] José M Faria. Machine learning safety: An overview. In *Proceedings of the 26th Safety-Critical Systems Symposium, York, UK*, 2018. 1

[12] Aditya Ganeshan, Vivek BS, and R Venkatesh Babu. Fda: Feature disruptive attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8069–8079, 2019. 3

[13] Lianli Gao, Qilong Zhang, Jingkuan Song, Xianglong Liu, and Heng Tao Shen. Patch-wise attack for fooling deep neural network. In *European Conference on Computer Vision*, pages 307–322. Springer, 2020. 6

[14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1, 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016. 3, 6

[16] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4733–4742, 2019. 3

[17] Yi Huang and Adams Wai-Kin Kong. Transferable adversarial attack based on integrated gradients. *arXiv preprint arXiv:2205.13152*, 2022. 2, 3, 6

[18] P Izmailov, AG Wilson, D Podoprikhin, D Vetrov, and T Garipov. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pages 876–885, 2018. 2

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25:1097–1105, 2012. 3

[20] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016. 1, 2, 6

[21] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 1, 2

[22] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017. 2, 3

[23] 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. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3866–3876. PMLR, 2019. 3

[24] 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. 6

[25] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 2, 3, 6

[26] Jiajun Lu, Hussein Sibai, Evan Fabry, and David Forsyth. No need to worry about adversarial examples in object detection in autonomous vehicles. *arXiv preprint arXiv:1707.03501*, 2017. 1

[27] Yan Luo, Xavier Boix, Gemma Roig, Tomaso Poggio, and Qi Zhao. Foveation-based mechanisms alleviate adversarial examples. *arXiv preprint arXiv:1511.06292*, 2015. 1

[28] Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32:13153–13164, 2019. 2, 3

[29] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 1, 2, 3

[30] Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian infer-

ence. *Journal of Machine Learning Research*, 18:1–35, 2017. 2, 3, 4, 5

[31] Muzammal Naseer, Salman H Khan, Shafin Rahman, and Fatih Porikli. Task-generalizable adversarial attack based on perceptual metric. *arXiv preprint arXiv:1811.09020*, 2018. 3

[32] 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. 6

[33] NeurIPS2017. NIPS 2017: Non-targeted Adversarial Attack, 2017. https://www.kaggle.com/c/nips-2017-non-targeted-adversarial-attack/. 6

[34] 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. 2

[35] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on Control and Optimization*, 30(4):838–855, 1992. 3

[36] David Ruppert. Efficient estimations from a slowly convergent robbins-monro process. Technical report, Cornell University Operations Research and Industrial Engineering, 1988. 3

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. 4, 6

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[39] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014. 1, 2

[40] 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. 6

[41] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI Conference on Artificial Intelligence*, 2017. 6

[42] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017. 6

[43] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 4

[44] AW Van der Vaart. Cambridge series in statistical and probabilistic mathematics. *Asymptotics Statistics*, 1998. 3

[45] Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1924–1933, 2021. 2, 6

[46] Zhibo Wang, Hengchang Guo, Zhifei Zhang, Wenxin Liu, Zhan Qin, and Kui Ren. Feature importance-aware transferable adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7639–7648, 2021. 3, 6

[47] Andrew G Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective of generalization. *Advances in Neural Information Processing Systems*, 33:4697–4708, 2020. 2, 3

[48] Dongxian Wu, Yisen Wang, Shu-Tao Xia, James Bailey, and Xingjun Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 7

[49] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019. 2, 3, 6

[50] Qilong Zhang, Xiaodan Li, YueFeng Chen, Jingkuan Song, Lianli Gao, Yuan He, and Hui Xue'. Beyond imagenet attack: Towards crafting adversarial examples for black-box domains. In *International Conference on Learning Representations*, 2022. 1

[51] Yi Zhao, Ke Xu, Qi Li, Haiyang Wang, Dan Wang, and Min Zhu. Intelligent networking in adversarial environment: challenges and opportunities. *Science China Information Sciences*, 65(7):170301, 2022. 1

[52] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang. Transferable adversarial perturbations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–467, 2018. 3