Direction-Aggregated Attack for Transferable Adversarial Examples

TIANJIN HUANG, Eindhoven University of Technology, Eindhoven, the Netherlands
VLADO MENKOVSKI, Eindhoven University of Technology, Eindhoven, the Netherlands
YULONG PEI, Eindhoven University of Technology, Eindhoven, the Netherlands
YUHAO WANG, National University of Singapore, Singapore
MYKOLA PECHENIZKIY, Eindhoven University of Technology, Eindhoven, the Netherlands

Deep neural networks are vulnerable to adversarial examples that are crafted by imposing imperceptible changes to the inputs. However, these adversarial examples are most successful in white-box settings where the model and its parameters are available. Finding adversarial examples that are transferable to other models or developed in a black-box setting is significantly more difficult. In this paper, we propose the Direction-Aggregated adversarial attacks that deliver transferable adversarial examples. Our method utilizes aggregated direction during the attack process for avoiding the generated adversarial examples overfitting to the white-box model. Extensive experiments on ImageNet show that our proposed method improves the transferability of adversarial examples significantly and outperforms state-of-the-art attacks, especially against adversarial robust models. The best averaged attack success rates of our proposed method reaches 94.6% against three adversarial trained models and 94.8% against five defense methods. It also reveals that current defense approaches do not prevent transferable adversarial attacks.

CCS Concepts: • **General and reference** → **Reliability**; • **Computing methodologies** → **Adversarial learning**.

Additional Key Words and Phrases: adversarial examples, transferablility, deep neural network

ACM Reference Format:
Tianjin Huang, Vlado Menkovski, Yulong Pei, YuHao Wang, and Mykola Pechenizkiy. Under review. Direction-Aggregated Attack for Transferable Adversarial Examples. 1, 1 (April Under review), 16 pages. https://doi.org/---

1 INTRODUCTION

Deep Neural Networks (DNNs) have achieved a great success in many tasks, e.g. image classification [9, 12], object detection [6], segmentation [21], etc. However, these high-performing models have been shown to be vulnerable to adversarial examples [7, 26]. In other words, carefully crafted changes to the inputs can change the model's prediction drastically. This fragility has raised concerns on security-sensitive tasks such as autonomous cars, face recognition, and malware detection. Well

Authors' addresses: Tianjin Huang, Eindhoven University of Technology, Eindhoven, the Netherlands, t.huang@tue.nl; Vlado Menkovski, Eindhoven University of Technology, Eindhoven, the Netherlands, v.menkovski@tue.nl; Yulong Pei, Eindhoven University of Technology, Eindhoven, the Netherlands, y.pei.1@tue.nl; YuHao Wang, National University of Singapore, Singapore, yohanna.wang0924@gmail.com; Mykola Pechenizkiy, Eindhoven University of Technology, Eindhoven, the Netherlands, m.pechenizkiy@tue.nl.

XXXX-XXXX/Under review/4-ART $15.00
https://doi.org/---

designed adversarial examples are not only useful to evaluate the robustness of models against adversarial attacks but also beneficial to improve the robustness of them [7].

Plenty of ways have been proposed to craft adversarial examples, which can be divided into white-box and black-box attacks. White-box attacks utilize complete knowledge including model architecture, model parameters, training strategy and training method, e.g. fast gradient sign method (FGSM) [7], Iterative Fast Gradient Sign Method (I-FGSM) [14], Project gradient descent (PGD) [22], Deepfool [23], Momentum Iterative Fast Gradient Sign Method (MI-FGSM) [3] and Carlini & Wagner's attack [1]. On the contrary, black-box attacks fool model's prediction without any knowledge about the model. It has been shown that adversarial examples generated by white-box attacks have the ability to fool other black-box models, which is known as transferability property [26]. The transferability of adversarial examples enables practical black-box attacks and imposes huge threat on real-world applications. However, the transferability of adversarial examples usually is very low because these adversarial examples easily overfit to the specific model, i.e. the model for generating these adversarial examples. Therefore, avoiding the *overfitting* problem is the key to generate transferable adversarial examples.

Deep neural networks applied to high dimensional classification tasks are typically very complex models, in other words, the decision boundary is highly non-linear and tends to have high curvature, e.g., the decision boundary of model 1 in Fig. 1. We believe that it is the high curvature of decision boundary that make adversarial examples decrease their ability to attack other models especially adversarial robust models with smoothed decision boundary [2, 15]. As shown in Fig. 1, the adversarial attack direction at sample x (the black arrow line in Fig. 1) tends to overfit to model 1 because this attack direction is the best direction for attacking model 1, but not a good direction for attacking model 2. To mitigate the effect of adversarial examples easily overfitting to the white-box model, we propose to aggregate the attack directions of a set of examples perturbed with Gaussian noise. The green solid arrow line in Fig. 1 shows the aggregated direction. It is easy to see that the green solid arrow line is a good attack direction for both model 1 and model 2. Therefore, adversarial examples generated by aggregated direction can achieve good transferability. Based on this, we propose Direction-Aggregated attack for improving the transferability of adversarial examples. Extensive experiments in later sections show that our method achieves state-of-the-art results.

In detail, our contributions are summarized as follows:

- We propose to aggregate attack directions in order to guide the attack direction to generalized decision boundary and avoid overfitting to the specific model's decision boundary. Based on aggregated direction, we propose Direction-Aggregated attack.
- We demonstrate that our proposed method outperforms state-of-the-art attacks by extensive experiments on ImageNet. The best averaged attack success rates of our method achieves 94.6% against three ensemble adversarial trained models and 94.8% against five defense methods, which also reveals that current defense models are not safe to transferable adversarial attacks. We hope our attack strategy can serve as a benchmark for evaluating the effectiveness of adversarial defense methods in the future.
- We experimentally show that sampling times N, standard deviation σ, iterations T and perturbation size ϵ induced in our method play an important role in achieving the transferability of adversarial examples, and the influence of these parameters on achieving the transferability is insensitive to the white-box model.

The rest of this paper is organized as follows. Section 2 presents related researches. Section 3 describes our method in detail. Section 4 shows the extensive experiments for evaluating our proposed method. Section 5 shows discussions about our method. Finally, Section 6 draws conclusions of this study.

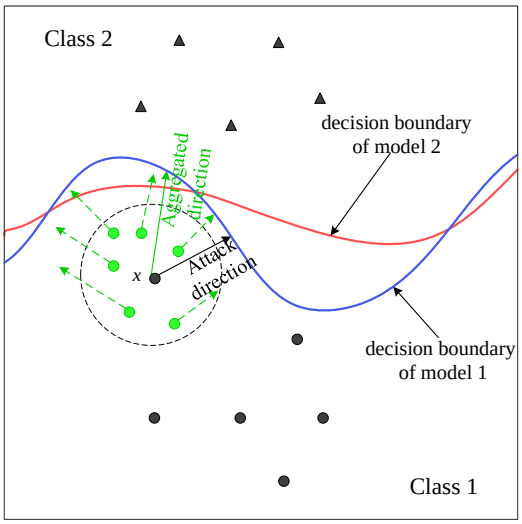


Fig. 1. A simple schematic diagram for explaining why aggregated direction can mitigate the overfitting problem of adversarial examples. Black circle and triangle markers denote samples of class 1 and class 2 respectively. Red and blue lines represent the decision boundary of model 1 and model 2. Circle with dotted line is the sets of the distribution $\mathcal{N}(x, \sigma^2\mathbf{I})$. Black arrow line denotes the attack direction $(sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}), y)))$ of model 1 at the sample x. Green arrow dotted lines are the attack direction at the perturbed sample with Gaussian noise. Green arrow solid lines denote the aggregated direction by the vector addition of the green arrow dotted lines.

.

2 RELATED WORK

Adversarial examples Since the phenomenon that DNNs are vulnerable to adversarial examples has been reported [26], many related researches have emerged. On the one hand, some of them propose to generate adversarial examples that can be applied on physical world [5, 14]. On the other hand, some of them focus on reducing the minimal size of adversarial perturbations and improving attack success rates [1, 3, 7, 23]. Among these researches, the success rate is still low under black-box setting, especially against adversarial robust models. Recently, several papers improve the success rate based on transferable adversarial attack. Inkawhich et al. [10] and Zhou et al. [30] improve the transferability by operating on intermediate feature maps. Li et al. [16] and Liu et al. [19] generate more transferable adversarial examples based on ensemble attacks. Xie et al. [29], Dong et al. [4] and Lin et al. [18] enhance the transferability of adversarial examples by imposing transformation on input images.

Defend against adversarial examples Correspondingly, many methods have been proposed to defend these adversarial examples. Goodfellow, Shlen and Szegedy [7], Madry et al. [22] augment training dataset by adversarial examples in order to improve adversarial robustness of models. Tramer et al. [27] further improve adversarial robustness by ensemble adversarial training. Meanwhile, Xie et al. [28] and Guo et al. [8] impose transformation on input images at inference time to mitigate adversarial effect. Liao et al. [17] propose high-level representation guided denoiser to purify the perturbed examples. Cohen, Rosenfeld and Kolter [2] build guaranteed adversarial robustness via randomized smoothing.

3 PROPOSED METHOD

In this section, we first give a description for notations. Then we give details for our method.

3.1 Notation

Let y denote true label, $\boldsymbol{x}$ denote real image and $\boldsymbol{x}^*$ denote the perturbed image. $f_\theta(\boldsymbol{x})$ denotes deep neural network and $\boldsymbol{x} \in R^d$. $L(f_\theta(\boldsymbol{x}), y)$ represents the Cross-Entropy loss. $sgn(\cdot)$ denotes the sign function. $\nabla_{\boldsymbol{x}} L(\cdot)$ denotes the gradient of $L(\cdot)$ with respect to $\boldsymbol{x}$. $Clip^{\epsilon}_{\boldsymbol{x}}(\cdot)$ function limit the generated adversarial example $\boldsymbol{x}^*$ to the ϵ max-norm ball of $\boldsymbol{x}$. ϵ is the allowed perturbation size. α is the step size. ε is Gaussian noise. $\mathcal{N}(\cdot)$ denotes a Gaussian distribution.

3.2 Direction-Aggregated Attack

As stated in Section 1, adversarial examples could overfit to white-box models due to the very complex decision boundary (Fig. 1). To generate more transferable adversarial examples, we propose Direction-Aggregated attack. In our method, we mitigate the overfitting problem of adversarial examples by aggregating the attack directions of a set of examples perturbed with Gaussian noise. We integrate the aggregated direction to basic adversarial attacks, i.e. Fast Gradient Sign Method (FGSM) [26], Iterative Fast Gradient Sign Method (I-FGSM) [13] and Momentum Iterative Fast Gradient Sign Method (MI-FGSM) [3], for improving their transferability. Besides, to further enhance the transferability, we combine our method with other transferable adversarial attacks, i.e. Diverse Input Method (DIM) [29], Translation-Invariant Method (TIM) [4], TI-DIM [4]. Concretely, the updating procedures for each attack are formalized as follows.

DA-FGSM To mitigate the effect of overfitting to the specific model and improve the transferability of adversarial examples for FGSM attack, we propose Direction-Aggregated FGSM (DA-FGSM). For FGSM, adversarial examples are generated by linearizing loss function. The updating procedure is expressed as follows:

$$\boldsymbol{x}^* = \boldsymbol{x} + \epsilon \cdot sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}), y)). \tag{1}$$

For our proposed DA-FGSM, the attack direction is replaced with aggregated direction which is achieved by aggregating the attack directions of a set of examples perturbed by Gaussian noise. Formally, it can be represented as follows:

$$\boldsymbol{x}^* = \boldsymbol{x} + \epsilon \cdot sgn(\sum_{i=0}^{N}(sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x} + \boldsymbol{\varepsilon}_i), y)))), \tag{2}$$

where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$ is Gaussian noise, σ denotes standard deviation of the Gaussian distribution and controls the size of perturbation imposed on the input images, and N denotes the sampling times from Gaussian distribution.

DAI-FGSM To improve the transferability for I-FGSM. We propose Direction-Aggregated I-FGSM (DAI-FGSM). For I-FGSM, it is a multi-step variant of FGSM and restricts the perturbed size to the ϵ max-norm ball. With the initialization $\boldsymbol{x}^*_0 = \boldsymbol{x}$, the perturbed data in t-th step $\boldsymbol{x}^*_t$ can be expressed as follows:

$$\boldsymbol{x}^*_t = Clip^{\epsilon}_{\boldsymbol{x}}\{\boldsymbol{x}^*_{t-1} + \alpha \cdot sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}^*_{t-1}), y))\}. \tag{3}$$

For our proposed DAI-FGSM, the attack direction at each iteration is replaced with the aggregated direction. The updating procedure can be formalized as follows:

$$\boldsymbol{x}^*_t = Clip^{\epsilon}_{\boldsymbol{x}}\{\boldsymbol{x}^*_{t-1} + \alpha \cdot sgn(\sum_{i=0}^{N}(sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}^*_{t-1} + \boldsymbol{\varepsilon}_i), y))))\}. \tag{4}$$

M-DAI-FGSM We integrate the momentum term into DAI-FGSM for improving the attack ability, which is called Momentum Direction-Aggregated I-FGSM (M-DAI-FGSM). The updating procedure of M-DAI-FGSM can be expressed as follows:

$$\boldsymbol{g}_a = \sum_{i=0}^{N} (sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}^*_{t-1} + \boldsymbol{\varepsilon}_i), y))) \tag{5}$$

$$\boldsymbol{g}_t = \mu \cdot \boldsymbol{g}_{t-1} + \frac{\boldsymbol{g}_a}{\|\boldsymbol{g}_a\|_1} \tag{6}$$

$$\boldsymbol{x}^*_t = Clip^{\epsilon}_{\boldsymbol{x}}\{\boldsymbol{x}^*_{t-1} + \alpha \cdot sgn(\boldsymbol{g}_t)\}, \tag{7}$$

where $\boldsymbol{g}_t$ is the accumulated gradient at iteration t and μ is the decay factor of momentum term, and $\boldsymbol{g}_a$ is the aggregated direction.

DA-DIM We combine our proposed M-DAI-FGSM with DIM for further improving the transferability of adversarial examples and denote it as Direction-Aggregated DIM (DA-DIM). The updating procedure is similar to M-DAI-FGSM, with the replacement of Eq. (5) by follows:

$$\boldsymbol{g}_a = \sum_{i=0}^{N} (sgn(\nabla_{\boldsymbol{x}} L(f_\theta(T(\boldsymbol{x}^*_{t-1} + \boldsymbol{\varepsilon}_i; p)), y))), \tag{8}$$

where $T(\cdot; p)$ is the stochastic transformation function and p is the transformation probability.

DA-TIM Similar to DA-DIM, we combine M-DAI-FGSM with TIM and denote it as Direction-Aggregated TIM (DA-TIM). Likewise, the updating procedure is similar to M-DAI-FGSM, with the replacement of Eq. (6) by follows:

$$\boldsymbol{g}_t = \mu \cdot \boldsymbol{g}_{t-1} + \frac{\mathcal{W} * \boldsymbol{g}_s}{\|\mathcal{W} * \boldsymbol{g}_s\|_1}, \tag{9}$$

where $*$ is the convolutional operation and $\mathcal{W}$ is the pre-defined kernel. A Gaussian kernel is chosen for our experiments.

DA-TI-DIM Following [18], we combine M-DAI-FGSM with TIM and DIM together and denote it as Direction-Aggregated TI-DIM (DA-TI-DIM). The updating procedure can be presented as follows:

$$\boldsymbol{g}_a = \sum_{i=0}^{N} (sgn(\nabla_{\boldsymbol{x}} L(f_\theta(T(\boldsymbol{x}^*_{t-1} + \boldsymbol{\varepsilon}_i; p)), y))) \tag{10}$$

$$\boldsymbol{g}_t = \mu \cdot \boldsymbol{g}_{t-1} + \frac{\mathcal{W} * \boldsymbol{g}_a}{\|\mathcal{W} * \boldsymbol{g}_a\|_1} \tag{11}$$

$$\boldsymbol{x}^*_t = Clip^{\epsilon}_{\boldsymbol{x}}\{\boldsymbol{x}^*_{t-1} + \alpha \cdot sgn(\boldsymbol{g}_t)\}. \tag{12}$$

The pseudocode of M-DAI-FGSM is summarized in Algorithm 1 and the code is provided[1].

4 EXPERIMENTS

In this section, we evaluate the effectiveness of our method empirically. We first introduce dataset and experimental settings. Then we show the performance of our proposed method against normal and defense models. Finally, a simple analysis is carried for analyzing the influence of the parameters N, σ, ϵ and T on achieving the transferability of adversarial examples.

[1] https://github.com/Juintin/DA-Attack.git

Algorithm 1 M-DAI-FGSM

Require: A input image x with true label y; a classifier f with loss function L; perturbation size ϵ; maximum iterations T; Standard deviation of Gaussian distribution σ; The decay factor μ; aggregate direction $\boldsymbol{g}_a$.

Ensure: An adversarial example $\boldsymbol{x}^*$

$\alpha = \epsilon/T$
$\boldsymbol{x}_0^*=\boldsymbol{x}$; $\boldsymbol{g}_0 = 0$
for $t = 1$ to T **do**
    $\boldsymbol{g_a}$=0
    for $i = 0$ to N **do**
        Get $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(0, \sigma^2 \mathcal{I})$
        Aggregate attack directions as $\boldsymbol{g_a}$=$\boldsymbol{g_a}$+$sgn(\nabla_{\boldsymbol{x}} L(f_\theta(\boldsymbol{x}_{t-1}^* + \boldsymbol{\varepsilon}_i), y))$
    end for
    Update $\boldsymbol{g_t}$=$\mu \cdot \boldsymbol{g}_{t-1}$+$\frac{\boldsymbol{g}_a}{\|\boldsymbol{g}_a\|_1}$
    Update $\boldsymbol{x}_t^* = Clip_x^{\epsilon}\{\boldsymbol{x}_{t-1}^* + \alpha \cdot sgn(\boldsymbol{g}_t)\}$
end for
$\boldsymbol{x}^* = \boldsymbol{x}_t^*$
return $\boldsymbol{x}^*$

4.1 Experimental Settings

Dataset Following the strategy used in [18], 1000 images that are correctly classified by all testing models are randomly selected from ILSVRC 2012 validation set. For a fair comparison with state-of-the-art methods, we use the same 1000 images[2] in [18].

Models Four normal trained models and three ensemble adversarial trained models are used for evaluating adversarial examples, which are Inception-V3 (Inc-V3) [25], Inception-v4 (Inc-V4) [24], Inception-Resnet-v2 (IncRes-V2) [24], Resnet-V2 (Res-101) [9], Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ [27] respectively. Besides, five advanced defense methods are considered for further evaluating the effectiveness of our method. Specifically, the selected advanced defense methods are High-level representation guided denoiser (HGD) [17], Random resizing and padding (R&P) [28], NIPS-r3[3], feature distillation (FD) [20] and purifying perturbations by image compression (Comdefend) [11].

Baseline Several most recently proposed methods aiming at generating transferable adversarial examples will be taken as baselines:

- DIM [29], which generates transferable examples by random resizing input images;
- TIM [4], which generates transferable examples by a set of translated images;
- SI-NI-FGSM [18], which generates transferable examples by scaled images and nesterov accelerated gradients; and
- The combinations of DIM, TIM and SI-NI-FGSM, namely TI-DIM [4], SI-NI-TIM [18], SI-NI-DIM [18] and SI-NI-TI-DIM [18] attacks.

Considering that we completely follow the experimental settings in [18], all the baseline results except for the attack success rates against FD and ComDefend in Table 6 in our experiments are from [18].

Hyper-Parameters We follow the settings in [18] for all hyper-parameters, the maximum perturbation ϵ is set to 16 and the number of iterations T is set to 12 as default values. Accordingly $\alpha = \epsilon/T$.

[2]https://github.com/JHL-HUST/SI-NI-FGSM
[3]https://github.com/anlthms/nips-2017/tree/master/mmd

The momentum parameter μ is set to 1.0. For DIM and TI-DIM methods, the transformation probability is set to 0.5. For TIM method, Gaussian kernel is adopted as our baseline experiments and kernel size is set to 7×7. For SI-NI-FGSM, SI-NI-TIM, SI-NI-DIM and SI-NI-TI-DIM methods, the number of scales is set to 5. For our proposed attacks, sampling times N and standard deviation σ are set to 30 and 0.05 respectively.

Criteria We use attack success rates to reflect the ability of adversarial examples attacking a model. Given a set of adversarial examples D^*. The attack success rates is defined as follows:

$$100 \times \frac{\sum_{i=1}^{M} [\arg\max_i f_i(x_i^*) \neq y_i]}{M}, \tag{13}$$

where $(x_i^*, y_i) \in D*$ and M is the number of adversarial examples in D^*.

4.2 Single-Model Attacks

We first evaluate the effectiveness of our proposed method based on the single model. DIM [29], TIM [4] and SI-NI-FGSM [18] and their combinations, i.e. SI-NI-TIM, TI-DIM, SI-NI-TI-DIM, are taken as baselines. Besides, several popular normal adversarial attacks, i.e. FGSM, I-FGSM, MI-FGSM, PGD, C&W, are utilized to show the effectiveness of our method.

Comparison with normal and transferable attacks The attack success rates of DIM, TIM, SI-NI-FGSM, normal attacks and our proposed method are shown in Table 1. The adversarial examples are crafted based on Inc-V3 model. From Table 1, it can be observed:

- Adversarial examples are much easier to attack normal trained models than adversarial trained models.
- Adversarial examples generated by transferable attacks have much higher attack success rates against black-box models than normal attacks.
- Our proposed M-ADI-FGSM attack outperforms the current state-of-the-art SI-NI-FGSM attack by 4.6% to 10.4%. Besides, DA-FGSM and DAI-FGSM attacks without momentum acceleration still achieve remarkable results compared with normal attacks, which demonstrate the effectiveness of aggregated direction.

Comparison with the extensions of DIM and TIM To fully evaluate DA-TIM, DA-DIM and DA-TI-DIM attacks, adversarial examples are crafted by these attacks based on Inc-V3, Inc-V4, IncRes-V2, Res-101 models respectively. We test it against the four normal trained and three ensemble adversarial trained models. The evaluation results are shown in Table 2, Table 3 and Table 4. It can be observed from these results:

- The combinations of our method and DIM, TIM methods can greatly improve the transferability of adversarial examples, which also indicates that our method is perpendicular to these methods.
- Our method outperforms the state-of-the-art attacks among all experiments, i.e. SI-NI-TIM, SI-NI-DIM and SI-NI-TI-DIM, except for adversarial examples crafted on IncRes-V2 model. Besides, the attack success rates of our method against the adversarial trained models outperform state-of-the-art attacks by large margins.
- Our method boosts the attack success rates more on adversarial trained models than normal trained models.

Visibility We visualize 5 randomly selected pairs of adversarial examples generated by TIM, DIM, SI-NI-FGSM and M-DAI-FGSM attacks respectively and their corresponding clean images in Fig. 2. We can see that the adversarial examples generated by our method are similar to these generated by other methods in visibility, and all these adversarial examples are hard to be distinguished from their corresponding clean images by humans.

Table 1. The attack success rates (%) against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models. The adversarial examples are generated based on Inc-V3 model by normal adversarial attacks including FGSM, I-FGSM, PGD, C&W and transferable adversarial attacks including DIM, TIM, SI-NI-FGSM, DA-FGSM, DAI-FGSM and M-DAI-FGSM atttacks. $*$ denotes the white-box model being attacked.

	Attack	Inc-V3*	Inc-V4	IncRes-V2	Res-101	Inc-V3$_{ens3}$	Inc-V3$_{ens4}$	IncRes-V2$_{ens}$
Normal	FGSM	67.1	26.7	25	24.4	10.5	10	4.5
	I-FGSM	99.9	20.7	18.5	15.3	3.6	5.8	2.9
	PGD	99.5	17.3	15.1	13.1	6.1	5.6	3.1
	MI-FGSM	100.0	40.0	38.2	32.3	12.5	12.8	6.8
	C&W	100	18.4	16.2	14.3	3.8	4.7	2.7
Transferable	DIM	98.7	67.7	62.9	54	20.5	18.4	9.7
	TIM	100	47.8	42.8	39.5	24	21.4	12.9
	SI-NI-FGSM	100	76	73.3	67.6	31.6	30	17.4
	DA-FGSM(Ours)	87.6	47	43.6	42	18.3	17.4	9.5
	DAI-FGSM(Ours)	99.8	44	39.2	34.3	23.7	22.4	12.4
	M-DAI-FGSM(Ours)	99.8	**80.6**	**78.5**	**72.2**	**40.6**	**40.4**	**26.5**

Table 2. Comparison with TIM, SI-NI-TIM and the DA-TIM extension. Attack success rates (%) are shown in the table. Adversarial examples are generated based on Inc-V3, Inc-V4, IncRes-V2 and Res-101 respectively. $*$ denotes the success rates under white-box attack.

Model	Attack	Inc-V3	Inc-V4	IncRes-V2	Res-101	Inc-V3$_{ens3}$	Inc-V3$_{ens4}$	IncRes-V2$_{ens}$
Inc-V3	TIM	100*	47.8	42.8	39.5	24	21.4	12.9
	SI-NI-TIM	100*	77.2	75.8	66.5	51.8	45.9	33.5
	DA-TIM(Ours)	99.8*	**80.9**	**77.9**	**71.8**	**66.9**	**65.2**	**51.2**
Inc-V4	TIM	58.5	99.6*	47.5	43.2	25.7	23.3	17.3
	SI-NI-TIM	83.5	100*	76.6	68.9	57.8	54.3	42.9
	DA-TIM(Ours)	**84.2**	98.4*	**77.7**	**69.3**	**66.8**	**65.9**	**56.4**
IncRes-V2	TIM	62	56.2	97.5*	51.3	32.8	27.9	21.9
	SI-NI-TIM	**86.4**	**83.2**	99.5*	**77.2**	66.1	60.2	57.1
	DA-TIM(Ours)	80	78.5	94*	74	**69.5**	**66.4**	**66**
Res-101	TIM	59	53.6	51.8	99.3*	36.8	32.2	23.5
	SI-NI-TIM	78.3	74.1	73	99.8*	58.9	53.9	43.1
	DA-TIM(Ours)	**78.6**	**74.7**	**76**	99.2*	**72.1**	**69.7**	**62.7**

4.3 Ensemble-based Attacks

In this section, we further evaluate the performance of our method under ensemble-based attacks. Liu et al. [19] have shown that attacking multiple models simultaneously can generate more transferable adversarial examples. It is because if an adversarial example can attack multiple models successfully, it more likely can attack another model successfully.

We follow the ensemble-based attack strategy proposed in [3], which fuses the logit activations of multiple models to generate adversarial examples. In this experiment, we generate adversarial examples by attacking Inc-V3, Inc-V4, IncRes-V2 and Res-101 models simultaneously with equal ensemble weights. In Table 5, we show the attack success rates for DA-DIM, DA-TIM, DA-TI-DIM attacks and baselines. It shows that our method outperforms these baselines across all experiments. The highest attack success rates is achieved by our DA-TI-DIM attack and the average attack success rates against the three robust models reaches 94.6%.

4.4 Attacking Other Defense Models

To further show the performance of our method on defense models. We test it against HGD [17], R&P [28], NIPS-r3, FD [20] and ComDefend [11] defense methods. HGD, R&P and NIPS-r3 are the

Table 3. Comparison with DIM, SI-NI-DIM and the DA-DIM extension. The numbers in table denote attack success rates (%). Adversarial examples are generated based on Inc-V3, Inc-V4, IncRes-V2 and Res-101 respectively using DIM, SI-NI-DIM and DA-DIM methods. $*$ denotes the success rates under white-box attacks.

Model	Attack	Inc-V3	Inc-V4	IncRes-V2	Res-101	Inc-V3$_{ens3}$	Inc-V3$_{ens4}$	IncRes-V2$_{ens}$
Inc-V3	DIM	98.7*	67.7	62.9	54	20.5	18.4	9.7
	SI-NI-DIM	99.6*	84.7	81.7	75.4	36.9	34.6	20.2
	DA-DIM(Ours)	99.5*	**89**	**87.3**	**81.2**	**57.1**	**56.6**	**38.8**
Inc-V4	DIM	70.7	98.0*	63.2	55.9	21.9	22.3	11.9
	SI-NI-DIM	89.7	99.3*	84.5	78.5	47.6	45	28.9
	DA-DIM(Ours)	**90.8**	98.1*	**87.1**	**80.9**	**62.1**	**62.9**	**49.7**
IncRes-V2	DIM	69.1	63.9	93.6*	47.4	29.4	24	17.3
	SI-NI-DIM	**89.7**	**86.4**	99.1*	**81.2**	55	48.2	38.1
	DA-DIM(Ours)	86.1	85.8	95*	80.2	**64.6**	**59.7**	**57.1**
Res-101	DIM	75.9	70	71	98.3*	36	32.4	19.3
	SI-NI-DIM	88.7	84.2	84.4	99.3*	53.4	48	33.2
	DA-DIM(Ours)	**90.9**	**87.7**	**89.4**	99.2*	**75.3**	**72.6**	**62.9**

Table 4. Comparison with TI-DIM, SI-NI-TI-DIM and the DA-TI-DIM extension. The numbers in table denote attack success rates (%). Adversarial examples are generated based on Inc-V3, Inc-V4, IncRes-V2 and Res-101 respectively using TI-DIM, SI-NI-TI-DIM and DA-TI-DIM methods. $*$ denotes the success rates under white-box attacks.

Model	Attack	Inc-V3	Inc-V4	IncRes-V2	Res-101	Inc-V3$_{ens3}$	Inc-V3$_{ens4}$	IncRes-V2$_{ens}$
Inc-V3	TI-DIM	98.5*	66.1	63	56.1	38.6	34.9	22.5
	SI-NI-TI-DIM	99.6*	85.5	80.9	75.7	61.5	56.9	40.7
	DA-TI-DIM(Ours)	99.6*	**88.3**	**85.1**	**80.3**	**77.4**	**76.8**	**62.9**
Inc-V4	TI-DIM	72.5	97.8*	63.4	54.5	38.1	35.2	25.3
	SI-NI-TI-DIM	88.1	99.3*	83.7	77	65	63.1	49.4
	DA-TI-DIM(Ours)	**88.8**	97.8*	**83.9**	**78.3**	**75.7**	**75.7**	**68.1**
IncRes-V2	TI-DIM	73.2	67.5	92.4*	61.3	46.4	40.2	35.8
	SI-NI-TI-DIM	**89.6**	**87**	99.1*	**83.9**	74	67.9	63.7
	NS-TI-DIM(Ours)	84.2	83.5	94.5*	78.3	**76.1**	**73.1**	**72.8**
Res-101	TI-DIM	74.9	69.8	70.5	98.7*	52.6	49.1	37.8
	SI-NI-TI-DIM	86.4	82.6	84.6	99*	72.6	66.8	56.4
	DA-TI-DIM(Ours)	**88.1**	**83.8**	**86.2**	99.3*	**82.6**	**82.2**	**76.2**

top 3 defense methods in NIPS 2017 defense competition. FD and ComDefend are recently published defense methods for purifying adversarial perturbations. TI-DIM [4] and SI-NI-TI-DIM attacks [18] are presented as baselines. Adversarial examples are generated based on the ensemble of Inc-V3, Inc-V4, IncRes-V2 and Res-101 models. The attack success rates against FD and ComDefend defense are based on IncRes-V2$_{ens}$ model.

As shown in Table 6, our model achieves state-of-the-art results and reaches 94.8% for averaged attack success rates, which also indicates current defense methods are not safe to transferable adversarial attacks.

4.5 Parameter Analysis

In this section, we conduct a series of experiments to study the impact of different hyper-parameters on the transferability of adversarial examples.

Sampling Times N We explore the influence of sampling times N upon the transferability of adversarial examples. Fig. 3 shows the attack success rates (%) against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models under black-box settings. The generation

Fig. 2. Visualization of randomly selected clean images and their corresponding adversarial examples. All examples are generated by TIM, DIM, SI-NI-FGSM and M-DAI-FGSM attacks respectively.

Table 5. The attack success rates (%) against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models. Adversarial examples are generated based on the ensemble of Inc-V3, Inc-V4, IncRes-V2 and Res-101 models using DIM, SI-NI-DIM, TIM, SI-NI-TIM, TI-DIM, SI-NI-TI-DIM, DA-DIM, DA-TIM and DA-TI-DIM attacks respectively. *Average* column denotes averaged attack success rates against the three robust models. $*$ denotes the white-box model being attacked.

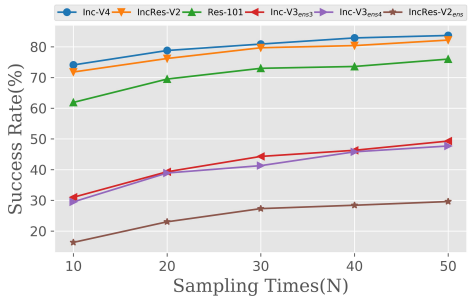
Attack	Inc-V3*	Inc-V4*	IncRes-V2*	Res-101*	Inc-V3$_{ens3}$	Inc-V3$_{ens4}$	IncRes-V2$_{ens}$	Average
DIM	99.7	99.2	98.9	98.9	66.4	60.9	41.6	56.3
SI-NI-DIM	100	100	100	99.9	88.2	85.1	69.7	81
DA-DIM(Ours)	99.9	99.8	99.7	99.8	**91**	**90.1**	**85.5**	**88.9**
TIM	99.9	99.3	99.3	99.8	71.6	67	53.2	63.9
SI-NI-TIM	100	100	100	100	93.2	90.1	84.5	89.2
DA-TIM(Ours)	99.8	99.8	99.2	99.6	**93.4**	**92.1**	**89.3**	**91.6**
TI-DIM	99.6	98.8	98.8	98.9	85.2	80.2	73.3	79.5
SI-NI-TI-DIM	99.9	99.9	99.9	99.9	96	94.3	90.3	93.5
DA-TI-DIM(Ours)	99.8	99.8	99.6	99.6	**96.2**	**94.7**	**93**	**94.6**

of adversarial examples is based on Inc-V3, Inc-V4, IncRes-V2 and Res-101 models respectively with standard deviation σ setting as 0.05.

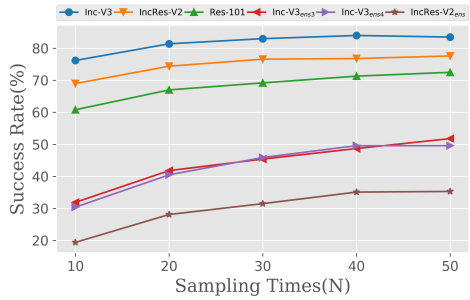
From Fig. 3, we can see that the attack success rates are growing with the increase of sampling times. In detail, the curve is growing fast when sampling times N is less than 30 and the trend of growth tends to be flattening when sampling times N is greater than 30. Besides, the growing trends of Fig. 3a, Fig. 3b, Fig. 3c and Fig. 3d are similar, which indicates that the influence of sampling times N on the transferability is little sensitive to the white-box model.

Table 6. The attack success rates against the five advanced defense models.

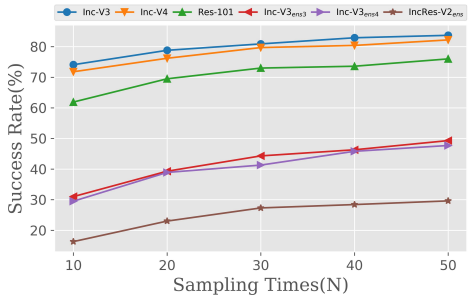
Attack	HGD	R&P	NIPS-r3	FD	ComDefend	Average
TI-DIM	84.8	75.3	80.7	84.2	79.6	80.9
SI-NI-TI-DIM	96.1	91.3	94.4	93.7	91.9	93.5
DA-TI-DIM(Ours)	**96.1**	**93.6**	**94.8**	**94.4**	**94.3**	**94.8**



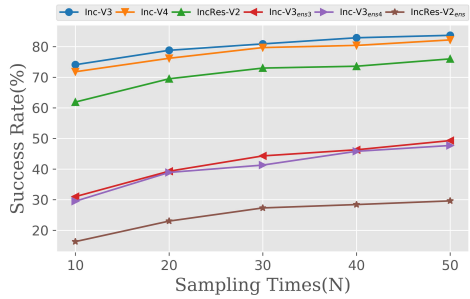
(a) Inc-V3



(b) Inc-v4



(c) IncRes-V2



(d) Res-101

Fig. 3. The attack success rates (%) of black-box attack against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models when varying sampling times N ranging from 10 to 50. The adversarial examples are generated based on Inc-V3 (Fig. 3a), Inc-V4 (Fig. 3b), IncRes-V2 (Fig. 3c) and Res-101 (Fig. 3d) models respectively by M-DAI-FGSM attack.

The σ for Gaussian Distribution Standard deviation σ controls the shape of Gaussian distribution and plays an important role in Gaussian noise generation. We study the influence of σ upon the transferability of adversarial examples. Fig. 4 shows the attack success rates against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models under black-box attack. Adversarial examples in this experiment are crafted based on Inc-V3, Inc-V4, IncRes-V2 and Res-101 models respectively with sampling times $N = 30$.

From Fig. 4, we can see that the attack success rates have a surge increasing at first, then the growing trends tend to be flattening. The surge increasing of the attack success rates indicates that the parameter σ plays an important role in our method. Besides, the similar trends among Fig. 4a,

Fig. 4b, Fig. 4c and Fig. 4d indicate that the influence of σ on achieving transferability is insensitive to the white-box model.

It is deserved to note that a very large σ is not encouraged for our method for the two reasons: 1) a larger σ indicates larger noise size will be generated (Fig. 1), thus more sampling times are needed to cover the sampling region. 2) noise sampling from a very large σ might already be too large to flip the prediction and consequently disturb the attack direction.

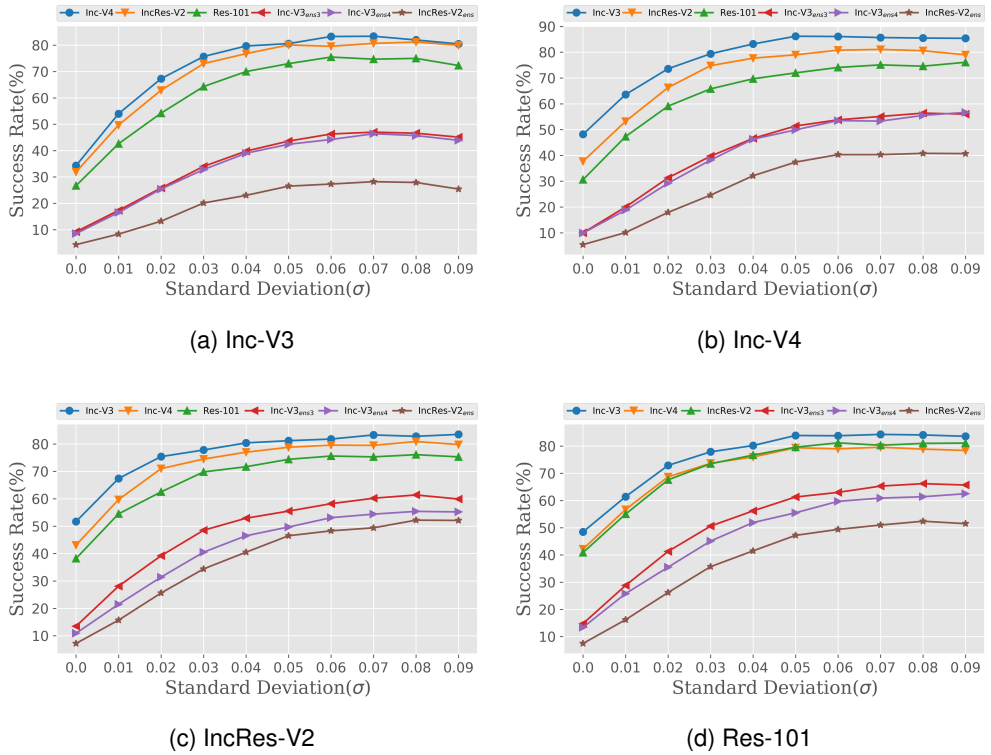


Fig. 4. The attack success rates (%) of black-box attack against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models when varying σ from 0 to 0.09. The adversarial examples are generated based on Inc-V3 (Fig. 4a), Inc-V4 (Fig. 4b), IncRes-V2 (Fig. 4c) and Res-101 (Fig. 4d) models respectively using M-DAI-FGSM attack.

Perturbation Size ϵ We study the impact of perturbation size ϵ on the success rates. We set sampling times N and standard deviation δ to 30 and 0.05 respectively. The attack success rates (%) against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models are achieved under black-box settings. The ϵ varies from 10 to 16 and the results are showed in Fig. 5.

From Fig. 5, we observe that the attack success rates increase as perturbation size ϵ increase on both adversarial trained models and normal trained models. Besides, the increasing trends of the attack success rates are similar even when adversarial examples are generated by different models .

Iterations T We study the impact of iterations T on the transferability of adversarial examples. Similarly, we set sampling times N and standard deviation δ to 30 and 0.05 respectively and generate adversarial examples based on normal trained models. Then these adversarial examples are tested on

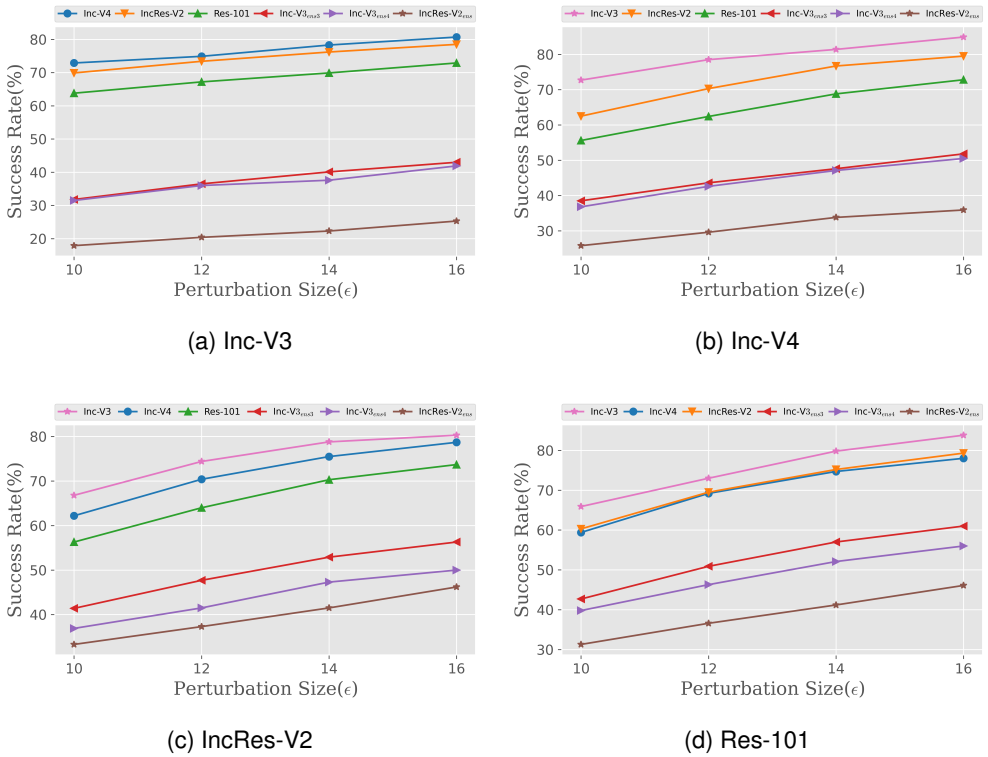


Fig. 5. The attack success rates (%) of black-box attack against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models when varying ϵ from 10 to 16. The adversarial examples are generated based on Inc-V3 (Fig. 5a), Inc-V4 (Fig. 5b), IncRes-V2 (Fig. 5c) and Res-101 (Fig. 5d) models respectively using M-DAI-FGSM attack.

the other models under black-box settings. Iterations T varies 5 to 22 and the results are showed in Fig. 6.

From Fig. 6, we can see that the attack success rates are improved with the increase of iterations T. It also indicates a smaller step size is beneficial to the transferability of adversarial examples since the step size $\alpha = \epsilon/T$, which is consist with the state in [29]. Besides, the increasing trend of the attack success rates is insensitive to the white-box model.

5 DISCUSSIONS

In this section, we provide further discussions for better understanding our proposed method. Our method mitigates the overfitting problem by aggregating the attack directions of a set of examples perturbed with Gaussian noise, which is different with DIM, TIM and SI-NI-FGSM attacks. DIM [29] refers the idea that input augmentation can effectively prevent models from overfitting during training process and generates adversarial examples by adding transformed inputs. TIM [4] generates adversarial examples by a set of translated images so that the resultant adversarial examples are translation-invariant. SI-NI-FGSM [18] utilizes Nesterov accelerated gradient and a set of scaled inputs to generate the scale-invariant adversarial examples. Essentially, these methods are based on geometric transformations of the inputs, e.g. scale, translation. Besides, the big boosting of

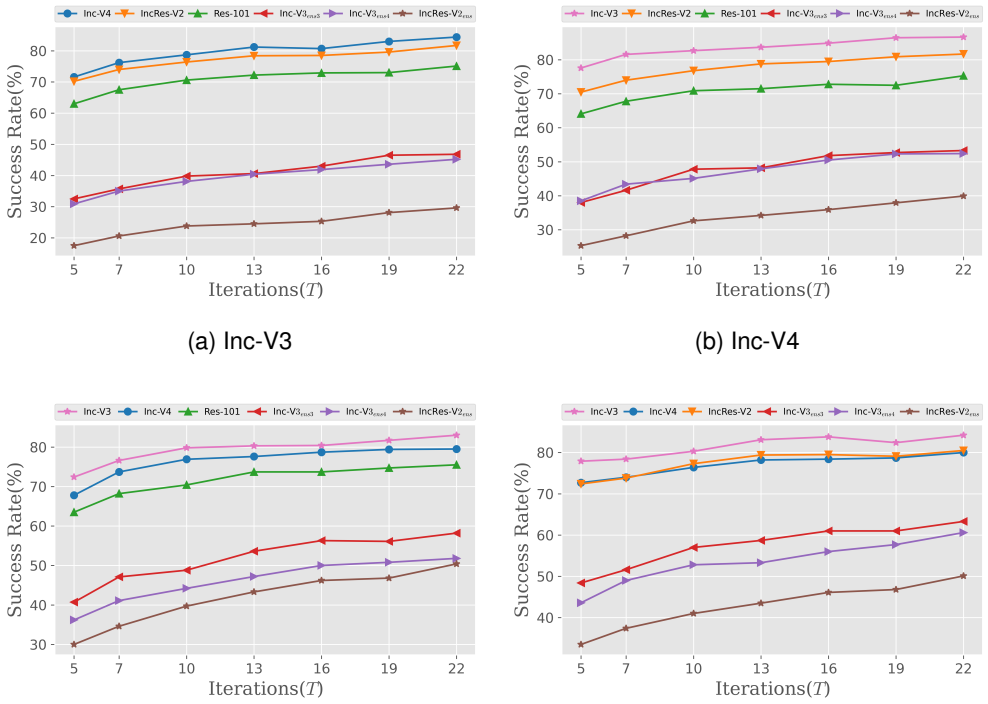


Fig. 6. The attack success rates (%) of black-box attack against Inc-V3, Inc-V4, IncRes-V2, Res-101, Inc-V3$_{ens3}$, Inc-V3$_{ens4}$ and IncRes-V2$_{ens}$ models when varying T from 5 to 22. The adversarial examples are generated based on Inc-V3 (Fig. 6a), Inc-V4 (Fig. 6b), IncRes-V2 (Fig. 6c) and Res-101 (Fig. 6d) models respectively using M-DAI-FGSM attack.

the combinations of our method with DIM,TIM on performance (Table 2, Table 3, Table 4) also provides the evidence that our method is perpendicular with these attacks. From the perspective of loss landscape, our method can be approximately equivalent to generate adversarial examples by Gaussian noise smoothed classifier:

$$\Phi(f)(\boldsymbol{x}) = \int_{R^n} g(\boldsymbol{y}-\boldsymbol{x}) f(\boldsymbol{y}) d\boldsymbol{y} \tag{14}$$

$$g(\boldsymbol{y}-\boldsymbol{x}) = \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(\boldsymbol{y}-\boldsymbol{x})^2}{2\sigma^2}. \tag{15}$$

Smoothing classifier by adding noise has been shown to be beneficial to improve the adversarial robustness of the model [2]. Therefore, accordingly, our method is approximately equivalent to generate adversarial examples by a robust model. This perspective provides an explanation for why our method improves the attack success rates more on adversarial trained model than normal trained model.

6 CONCLUSION

In this paper, we propose to improve the transferability by aggregating attack directions of a set of examples perturbed with Gaussian noise. Based on aggregated direction, we propose Direction-aggregated attack. We make extensive experiments on ImageNet under single model attack and ensemble-based attack to show the performance of our method. We show that our method achieves better results and outperform state-of-the-art attacks among all experiments except for the experiments made on IncRes-V2 model. The best averaged attack success rates of our method reaches 94.6% against three adversarial trained models and 94.8% against five defense methods under black-box attacks. The results also indicate current defense models are not safe to transferable adversarial attacks. Besides, we analyze the influence of sampling times N, standard deviation σ, perturbation size ϵ and iterations T on achieving the transferability of adversarial examples. We show that sampling times N, standard deviation σ, perturbation size ϵ and iterations T play an important role in our method and the influence of these parameters on achieving the transferability is insensitive to the white-box model.

REFERENCES

[1] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 39–57.

[2] Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. 2019. Certified adversarial robustness via randomized smoothing. *arXiv preprint arXiv:1902.02918* (2019).

[3] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9185–9193.

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

[5] Gamaleldin F. Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. 2018. Large margin deep networks for classification. In *Advances in Neural Information Processing Systems*, Vol. 2018-December. Neural information processing systems foundation, 842–852.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2014.81 arXiv:1311.2524

[7] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and Harnessing Adversarial Examples. (dec 2014). arXiv:1412.6572 http://arxiv.org/abs/1412.6572

[8] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. 2017. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117* (2017).

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[10] Nathan Inkawhich, Wei Wen, Hai Helen Li, and Yiran Chen. 2019. Feature space perturbations yield more transferable adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7066–7074.

[11] Xiaojun Jia, Xingxing Wei, Xiaochun Cao, and Hassan Foroosh. 2019. Comdefend: An efficient image compression model to defend adversarial examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6084–6092.

[12] Alex Krizhevsky and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. *Neural Information Processing Systems* (2012). https://doi.org/10.1016/j.protcy.2014.09.007 arXiv:1102.0183

[13] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533* (2016).

[14] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* (2016).

[15] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. 2018. Certified adversarial robustness with additive gaussian noise. *arXiv preprint arXiv:1809.03113* (2018).

[16] Yingwei Li, Song Bai, Yuyin Zhou, Cihang Xie, Zhishuai Zhang, and Alan Yuille. 2018. Learning transferable adversarial examples via ghost networks. *arXiv preprint arXiv:1812.03413* (2018).

[17] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. 2018. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE Conference on Computer Vision and*

Pattern Recognition. 1778–1787.

[18] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. 2020. Nesterov Accelerated Gradient and Scale Invariance for Adversarial Attacks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SJlHwkBYDH

[19] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. 2016. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770* (2016).

[20] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. 2019. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 860–868.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully Convolutional Networks for Semantic Segmentation ppt. In *CVPR 2015 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. https://doi.org/10.1109/CVPR.2015.7298965 arXiv:1411.4038

[22] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards Deep Learning Models Resistant to Adversarial Attacks. (jun 2017). arXiv:1706.06083 http://arxiv.org/abs/1706.06083

[23] Seyed Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2016-Decem. IEEE Computer Society, 2574–2582. https://doi.org/10.1109/CVPR.2016.282

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

[25] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

[26] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. (dec 2013). arXiv:1312.6199 http://arxiv.org/abs/1312.6199

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

[28] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. 2017. Mitigating adversarial effects through randomization. *arXiv preprint arXiv:1711.01991* (2017).

[29] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. 2019. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2730–2739.

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