This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Learning to Transform Dynamically for Better Adversarial Transferability

‡ Rongyi Zhu *† Zeliang Zhang Susan Liang Zhuo Liu ‡ Chenliang Xu
University of Rochester
{rongyi.zhu, zeliang.zhang, susan.liang, zhuo.liu, chenliang.xu}@rochester.edu

Abstract

Adversarial examples, crafted by adding perturbations imperceptible to humans, can deceive neural networks. Recent studies identify the adversarial transferability across various models, i.e., the cross-model attack ability of adversarial samples. To enhance such adversarial transferability, existing input transformation-based methods diversify input data with transformation augmentation. However, their effectiveness is limited by the finite number of available transformations. In our study, we introduce a novel approach named Learning to Transform (L2T). L2T increases the diversity of transformed images by selecting the optimal combination of operations from a pool of candidates, consequently improving adversarial transferability. We conceptualize the selection of optimal transformation combinations as a trajectory optimization problem and employ a reinforcement learning strategy to effectively solve the problem. Comprehensive experiments on the ImageNet dataset, as well as practical tests with Google Vision and GPT-4V, reveal that L2T surpasses current methodologies in enhancing adversarial transferability, thereby confirming its effectiveness and practical significance. The code is available at https://github.com/ZhangAIPI/TransferAttack.

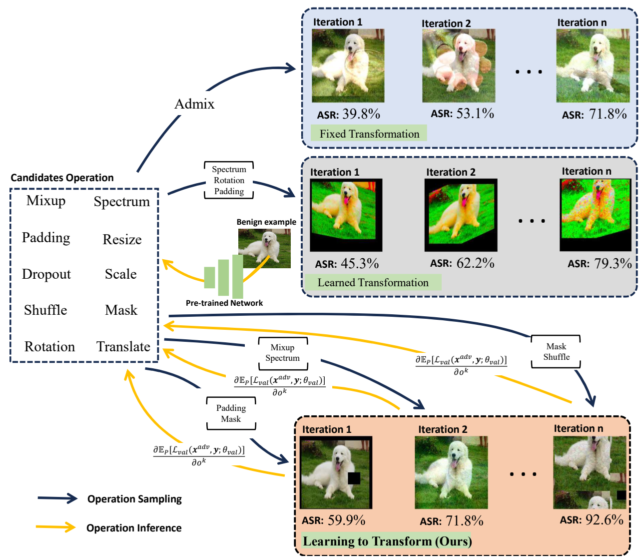

Figure 1. For input transformation-based attacks, most works design a fixed transformation and use it to craft the adversarial perturbation. The learning-based methods preliminarily predict augmentation strategies for current images for better adversarial transferability. These methods cannot respond to the distribution shifts between benign images and adversarial examples. We propose Learning to Transform (L2T), which uses the dynamic of the optimal transformation in each iteration to further boost the adversarial transferability.

1. Introduction

Neural networks have been adopted as the building block for various real-world applications, such as face detection [37, 41], autonomous driving [12, 25], and medical diagnosis [1, 35]. However, neural networks are vulnerable to adversarial examples, which contain human imperceptible adversarial perturbations on the benign input. This issue is increasingly concerning researchers, as it is essential for ensuring the trustworthy use of neural networks [3, 19, 65, 66].

In real-world scenarios of adversarial attacks, the target model is usually inaccessible. To attack these inaccessible models, many studies instead rely on surrogate models to generate adversarial examples [7, 57] and use generated samples to mislead the target model. This cross-model attack ability of samples generated on the surrogate models is called "adversarial transferability." Numerous research studies are dedicated to enhancing adversarial transferability, which can be classified into four categories: gradient-based methods [7, 26, 44, 47], input transformation-based methods [8, 26, 46, 57], architecture-based methods [23, 52], and ensemble-based methods [28, 60]. Among these attack methodologies, input transformation-based methods gain much popularity because of their plug-n-play advantage, which can be seamlessly integrated into other attack techniques [7, 44]. However, we discover that existing input

*Equal contribution
†Project lead
‡Corresponding author

transformation-based methods adopt the same transformation when crafting adversarial examples, limiting the flexibility of transformation operations. We hypothesize that we should select the optimal transformation dynamically in each iteration to enhance the adversarial transferability.

As shown in Fig. 1, prior input transformation-based methods often revolve around designing fixed augmentation strategies like resizing inputs [57], block masking [10], or mix-up [46]. A more dynamic approach is presented by [63], advocating the precomputation of various sequences of augmentation strategies to apply to each iteration to enhance the attack performance. Complementing this, Wu et al. [53] proposes the use of generative models for image augmentation to boost the adversarial transferability. Some studies go further, combining multiple augmentation strategies to amplify input diversity to improve the performance. For example, Yuan et al. [64] introduces a neural network that generates a prediction of the optimal transformation strategy and applies the strategy to improve performance. A further improvement is hindered by the limited number of transformations.

To fully utilize the limited number of transformations, a natural idea is to use a combination of operations. However, it is not always efficient to combine different transformations together for attack, as reported in [50]. We expect to find an optimal combination of transformations to achieve a trade-off between operation diversity and adversarial transferability. Nonetheless, the enormity of the search space presents a significant challenge, impeding the identification of the most efficacious combination of transformations during an attack for optimal adversarial transferability. To surmount this hurdle, we conceptualize the search process of the optimal combination of transformations as a problem of optimal trajectory search. Each node within this trajectory represents an individual transformation, and each directed edge means a transfer of the optimal transformation from the current step to the next step. To effectively obtain the optimal trajectory in such a large search space, we design a reinforcement learning-based approach, capitalizing on its demonstrated efficacy in navigating expansive search domains.

In this paper, we introduce a novel framework called Learning to Transform (L2T) to improve the adversarial transferability of generated adversarial examples. L2T dynamically learns and applies the optimal input transformation in each iteration. Instead of exhaustively enumerating all possible input transformation methods, we employ a reinforcement learning-based approach to reduce the search space and better utilize the transformations to improve the diversity. In each iteration of the adversarial attack, we sample a subset of transformations and apply them to the adversarial examples. Subsequently, we update the sampling probabilities by conducting gradient ascent to maximize the loss. Our method effectively learns the dynamics of optimal transformations in attacks, leading to a significant enhancement in adversarial transferability. Additionally, compared to other learn-based adversarial attack methods, our approach is more efficient for adversarial example generation, as it obviates the need for additional training modules.

We summarize our contributions as follows,

- We formulate the problem of optimal transformation in adversarial attacks, which studies finding the optimal combination of transformations to increase the input diversities, thus improving the adversarial transferability.
- We propose Learning to Transform (L2T) that exploits the optimal transformation in each iteration and dynamically adjusts transformations to boost adversarial transferability.
- Extensive experiments on the ImageNet dataset demonstrate that L2T outperforms other baselines. We also validate L2T's superiority in real-world scenarios, such as Google Vision and GPT-4V.

2. Related Work

2.1. Adversarial Attack

Various adversarial attacks have been proposed, *e.g.*, gradient-based attack [13, 20, 32], transfer-based attack [7, 31, 51, 57], score-based attack [4, 18, 22], decision-based attack [2, 21, 49], generation-based attack [45, 54]. Among these, transfer-based attacks do not require the information of the victim models, making it popular to attack the deep models in the real world and raise more research interests. To improve adversarial transferability, various momentum-based attacks have been proposed, such as MI-FGSM [7], NI-FGSM [26], VMI-FGSM [44], EMI-FGSM [47], *etc*. Several input transformation methods are also proposed, such as DIM [57], TIM [8], SIM [26], Admix [46], SIA [50], STM [11], BSR [43], *etc*., which augment images used for adversarial perturbation computation to boost transferability. The input transformation-based methods can be integrated into the gradient-based attacks for better performance.

Delving into the input transformation-based methods, most works are limited to designing a fixed transformation to augment the images, which limits the diversity of transformed images and the adversarial transferability. To address this issue, some researchers [53, 63, 64] propose to augment the images with a set of multiple transformations predicted by a pre-trained network. Automatic Model Augmentation (AutoMA) [63] adopts a Proximal Policy Optimization (PPO) algorithm in search of a strong augmentation policy. Adversarial Transformation-enhanced Transfer Attack (ATTA) [53] proposes to employ an adversarial transformation network in modeling the most harmful distortions. Adaptive Image Transformation Learner (AITL) [64] incorporates different image transformations into a unified framework to learn adaptive transformations for each benign sample to boost adversarial transferability. By applying optimal multiple transformations, the adversarial attack performance is largely improved.

2.2. Adversarial Defense

Various defense approaches have been proposed to mitigate the threat of adversarial attacks, such as adversarial training [32, 40, 48], input preprocessing [33, 55], feature denoising [24, 56, 62], certified defense [6, 14, 34], *etc*. Liao et al. [24] train a denoising autoencoder, namely the High-level representation guided denoiser (HGD), to purify the adversarial perturbations. Xie et al. [55] propose to randomly resize the image and add padding to mitigate the adversarial effect, namely the Randomized resizing and padding (R&P). Xu et al. [61] propose the Bit depth reduction (Bit-Red) method, which reduces the number of bits for each pixel to squeeze the perturbation. Liu et al. [29] defend against adversarial attacks by applying a JPEG-based compression method to adversarial images. Cohen *et al*. [6] adopt randomized smoothing (RS) to train a certifiably robust classifier. Naseer et al. [33] propose a Neural Representation Purifier (NRP) to eliminate perturbation.

3. Learning to Transform

3.1. Task definition

The crafting of adversarial examples usually takes an iterative framework to update the adversarial perturbation. Given a benign sample $\boldsymbol{x}$ and the corresponding label y, a transferable attack takes a surrogate classifier $f_{\boldsymbol{\theta}}$ and iteratively updates the adversarial example $\boldsymbol{x}^{adv}$ to maximize the loss of classifying $f_\theta(\boldsymbol{x}^{adv})$ to y. Take I-FGSM [38] as an example. The adversarial example $\boldsymbol{x}_t^{adv}$ at the t-th iteration can be formulated as follows:

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

where we denote α as the step size, $J(\cdot,\cdot)$ as the classification loss function. As identified by previous studies, the adversarial example exhibits a characteristic of transferability, where the adversarial examples generated by the surrogate model can fool other neural networks.

Input transformation-based methods are one of the most effective methods to boost adversarial transferability. With these methods, the adversarial samples are firstly transformed by a set of image transformations and then proceeded to gradient calculation. Let φ denote a set of image transformations operation o, where $\varphi = \{o^i | i \in \{1, 2, ..., k\}\}$. At the t-th iteration, the adversarial example $\boldsymbol{x}_t^{adv}$ is transformed sequentially by o^i as follows,

$$\varphi(\boldsymbol{x}_t^{adv}) = o^k \oplus o^{k-1} \oplus \cdots \oplus o^1(\boldsymbol{x}_t^{adv}), \qquad (2)$$

where $o^2 \oplus o^1(\boldsymbol{x})$ denotes the operation $o^2(o^1(\boldsymbol{x}))$, $o^1, o^2 \in \varphi$. We use the gradient of $\varphi(\boldsymbol{x}_t^{adv})$ with respect to the loss function to update the adversarial perturbation as Eq. (1).

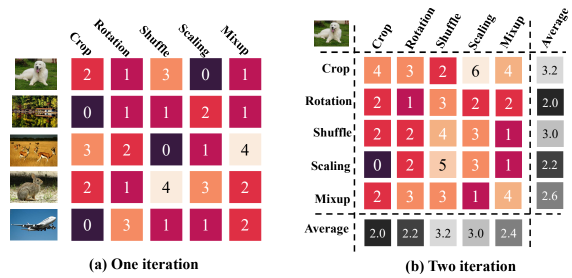

Figure 2. Comparsion for different operations in boosting the adversarial transferability. The number in the box denotes the number of fooled models (Maximum: 9). In (a), the horizontal axis denotes different transformation operations and the vertical axis denotes different benign examples. In (b), the vertical axis denotes the transformation used in the first iteration and the horizontal axis denotes the transformation used in the second iteration

There are two categories for selecting the operation set φ in the previous study. One line of research focuses on designing fixed transformation-based methods, which use a pre-defined transformation φ. For example, Admix chooses mixup and scaling for transformation φ. The other line of research proposes the learning-based transformation methods, which usually use a generative model to directly generate the transformed $\varphi(x)$. Compared with the fixed transformation-based methods, learning-based methods enjoy more diversity of transformed images, leading to a better performance in adversarial transferability. In our work, we study the learning-based transformation methods.

3.2. Motivation

Previous research designs lots of transformations to improve the diversity of images, thus guiding the adversarial attacks to focus more on the invariant robust features. However, it does not always work by increasing the number of transformed images for attacks to boost the adversarial transferability. Because some combination of transformations can cause damage to original examples, losing massive amounts of information used for transferable attacks. A natural question occurs to us, *for one image, does there exist the optimal combination of transformations for the best adversarial transferability?*

To answer this question, we start by generating adversarial examples in one iteration. We take an example of crafting adversarial examples using ResNet-18 to attack other 9 models[1]. We denote 5 operations for input transformation methods, namely the crop, rotation, shuffle, scaling, and mix-up. We use these operations on five images for attacks and report the number of models fooled. We report the results in Fig. 2. It can be seen that by shuffle, we can achieve the maximum

[1]ResNet-101, DenseNet-121, ResNext-50, Inception-v3, Inception-v4, ViT, PiT, Visformer, Swin

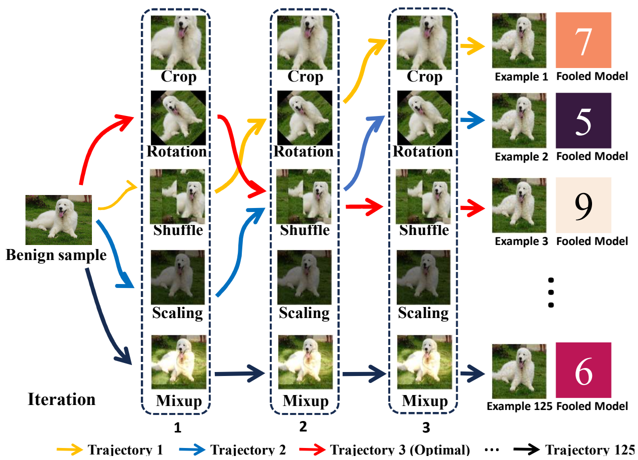

Figure 3. There exists an optimal transformation trajectory for boosting adversarial transferability. However, the search space increases exponentially with iteration number and operation number.

transferable attack success rates on a dog image, indicating the optimal transformations in all possible 5 operations.

We continue our discussion in the two-iteration scenario. Following the same setting in one iteration, we report the number of fooled models. It can be seen that by choosing crop in the first iteration and scaling for the second iteration, which successfully fooled 6 models out of 9. We also notice that shuffle, the optimal transformation in one iteration, can not maintain the optimal performance. The average fooled model for shuffle is less than crop in 0.2.

Following the aforementioned discussion, we move on to generating adversarial examples in 3 iterations, where we only take one operation as the image transformation to attack the image. As exemplified in Fig. 3, there are $5 \times 5 \times 5$ possible trajectories to transform the image for attacks. Among these trajectories, it can achieve the best performance by first shuffling, then rotating, and last shuffling the image. It should be noted it cannot consistently achieve the best performance by increasing the number of transformations for a higher diversity. As shown in Fig. 3, we respectively take the scaling, shuffle, and rotation operations at each iteration in trajectory 2. However, it has the worst attack success rate among the presented results.

Generalizing the previous problem to common cases, we are motivated to identify an optimal transformation trajectory $\mathcal{T}$, which is defined as the sequence of transformation used in each iteration as $(\varphi_1, \varphi_2, \ldots, \varphi_T)$, for the best adversarial transferability. Each element φ_T denotes the transformation used in iteration t. It can be formulated as follows:

$$\mathcal{T}^* = \underset{\mathcal{T}}{\operatorname{argmax}}(\mathbb{E}[\mathcal{L}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{\mathcal{T}}^{adv}), y)]), \tag{3}$$

where we denote $\boldsymbol{x}_{\mathcal{T}}^{adv}$ as the adversarial example generated by the surrogate model under transformation trajectory $\mathcal{T}$.

However, finding $\mathcal{T}^*$ is hard. First, the search space is large. For example, supposing five candidate transformations, even if we only take one operation in one iteration to transform the image, we will still have an enormous search space for ten iterations that will be 5^{10}. The number of possible transformation trajectories grows exponentially with increasing the number of iterations and candidate transformations. Second, we can not access the black-box model f, making it hard to optimize the Eq. (3) directly. Besides, as identified in the previous work [64], each image has a different optimal transformation to boost the adversarial transferability. There is no optimal transformation trajectory shared for all images.

3.3. Methodology

The problem of Eq. (3) can be transformed into an optimal trajectory search problem, on which reinforcement learning has shown great compatibility. We are inspired to take a reinforcement learning-based approach in solving this optimization problem to enhance adversarial transferability.

Supposing we have M operations $\{o^1, o^2, \ldots, o^M\}$ in total, the optimal transformation trajectory $\mathcal{T}$ is a temporal sequence of the combination of different operations. The probability $\boldsymbol{p}$ contains M possibilities $\{p_{o^1}, p_{o^2}, ..., p_{o^M}\}$ for each iteration. Each element p_{o^m} denotes the possibility of sampling operation $o^m, m \in \{1, 2, ..., M\}$. And p_{o^m} follows $\sum_{m=1}^{M} p_{o^m} = 1$. A transformation φ consists K operations $o^k, k \in \{1, 2, ..., M\}$. We sampled K operations from $\boldsymbol{p}$. We have the possibility of a transformation φ by $P(\varphi) = \prod_{k=1}^{K} p_{o^k}$.

For each iteration t, we sample a combination of transformation φ_t. Each transformation in φ_t is sampled from candidates depending on $\boldsymbol{p}$. To get an optimal trajectory $\mathcal{T} = (\varphi_1, ..., \varphi_T)$, we need to dynamically optimize the sampling distribution $\boldsymbol{p}$ in each iteration t. We formulate the problem of searching optima $\boldsymbol{p}^*$ in each iteration as follows,

$$\begin{aligned} \boldsymbol{p}^* &= \arg\max_{\boldsymbol{p}} \mathbb{E}_{\varphi \sim \mathbf{p}}[\mathcal{L}(f_{\boldsymbol{\theta}}(\varphi(\tilde{\boldsymbol{x}}^{adv})), y)] \\ \textit{s.t.} \quad \tilde{\boldsymbol{x}}^{adv} &= \arg\max_{\boldsymbol{x}^{adv}} \mathbb{E}_{\varphi \sim \mathbf{p}}[\mathcal{L}(f_{\boldsymbol{\theta}}(\varphi(\boldsymbol{x}^{adv})), y)], \end{aligned} \tag{4}$$

which is a bi-level optimization. The inner optimization targets to optimize the adversarial example, and the outer optimization tries to find the optimal sampling probability. Following [27], we adopt an one-step optimization strategy to derive the approximated $\boldsymbol{p}^*$:

$$\boldsymbol{p}^* \approx \boldsymbol{p} + \rho \cdot \boldsymbol{g}_{\boldsymbol{p}}, \tag{5}$$

where the ρ is the learning rate and $\boldsymbol{g}_{\boldsymbol{p}}$ is the gradient for $\boldsymbol{p}$.

Implementation details. We present the overview of our method in Fig. 4. First, we sample L sequences of transformation $\varphi_t^l, l \in [1, 2, ..., L]$, depending on the sampling

Algorithm 1 Gradient policy for optimal augmentation search.

Input: Classifier $f(\cdot)$;The benign sample $\boldsymbol{x}$ with ground-truth label y; Loss function $\mathcal{L}(\cdot,\cdot)$; candidate operation pool Γ, the number of iterations T, perturbation scale ϵ, policy learning rate ρ, number of operations K, number of transformations L, decay factor μ;

$\alpha = \epsilon/T$, $\boldsymbol{g}_0 = 0$, $\boldsymbol{x}_0^{adv} = \boldsymbol{x}, \boldsymbol{p} \sim \mathcal{N}(0,1)$

while $t = 1 \leftarrow T$ **do**

1. Under the distribution $\boldsymbol{p}$, sample L transformation φ_t, each consisting of K operations.
2. Transform adversarial examples:
$$\varphi_t^l(x_t^{adv}) = o^K \oplus o^{K-1} \oplus \cdots \oplus o^1(\boldsymbol{x}_t^{adv}).$$
3. Calculate the average gradient:
$$\bar{\boldsymbol{g}} = \frac{1}{L}\sum_{l=1}^{L} \nabla_{\boldsymbol{x}_{t-1}^{adv}} \mathcal{L}(\varphi_t^l(\boldsymbol{x}_{t-1}^{adv}), y).$$
4. Update the momentum:
$$\boldsymbol{g}_t = \mu \boldsymbol{g}_{t-1} + \frac{\bar{\boldsymbol{g}}}{||\bar{\boldsymbol{g}}||_1}.$$
5. Update the adversarial example:
$$\boldsymbol{x}_t^{adv} = \text{clip}(\boldsymbol{x}_{t-1}^{adv} + \alpha \cdot \text{sign}(\boldsymbol{g}_t), 0, 1).$$
6. Calculate the probability gradient:
$$\boldsymbol{g}_{\boldsymbol{p}} = \frac{\partial\left(\frac{1}{L}\sum_{l=1}^{L} \mathbf{P}(\varphi_t^l)\mathcal{L}(f_{\boldsymbol{\theta}}(\varphi_t^l(\mathbf{x}_t^{adv})), y)]\right)}{\partial \mathbf{P}(\varphi_t^l)}.$$
7. Update the probability:
$$\boldsymbol{p} = \boldsymbol{p} + \rho \cdot \boldsymbol{g}_{\boldsymbol{p}}.$$

end while

Output: $\boldsymbol{x}_T^{adv}$.

distribution $\boldsymbol{p}$. Next, we get the transformed examples denoted as $\varphi_t^l(x_t^{adv})$. The probability of each sequence φ_t^l is $\boldsymbol{P}(\varphi_t^l)$. We use φ_t to denotes all L transformation, $\varphi_t = \{\varphi_t^1, \varphi_t^2, ..., \varphi_t^L\}$. Then, we use Eq. (1) to update the adversarial examples for each iteration. The gradient is calculated by loss between L transformed examples and their corresponding labels. Last, after updating the adversarial example, we recompute the approximate $\boldsymbol{p}$. Specifically, we compute the gradient g_{o^k} of each sampled operation o^k as:

$$\begin{aligned} g_{o^k} &= \frac{\partial \mathbb{E}_{\varphi_t \sim \mathbf{p}}[\mathcal{L}(f_{\boldsymbol{\theta}}(\varphi_t(\mathbf{x}_t^{adv})), y)]}{\partial \mathbf{P}(\varphi_t)} \cdot \frac{\partial \mathbf{P}(\varphi_l)}{\partial p_{o^k}} \\ &= \frac{\partial \sum_{l=1}^{L} \mathbf{P}(\varphi_t^l)\mathcal{L}(f_{\boldsymbol{\theta}}(\varphi_t^l(\mathbf{x}_t^{adv})), y)])}{\partial \mathbf{P}(\varphi_t^l)} \cdot \frac{\partial \mathbf{P}(\varphi_l)}{\partial p_{o^k}} \\ &= \sum_{l=1}^{L} \mathcal{L}(f_{\boldsymbol{\theta}}(\varphi_t^l(\mathbf{x}_t^{adv}), y)) \cdot \frac{\partial \mathbf{P}(\varphi_t^l)}{\partial p_{o^k}}. \end{aligned} \tag{6}$$

We concat the gradients for each operation as $[g_{o^1}, g_{o^2}, \ldots, g_{o^K}]$, which is denoted as $\boldsymbol{g}_{\boldsymbol{p}}$. We use gradient ascent to update $\boldsymbol{p}$ by $\boldsymbol{g}_{\boldsymbol{p}}$ with the learning rate ρ.

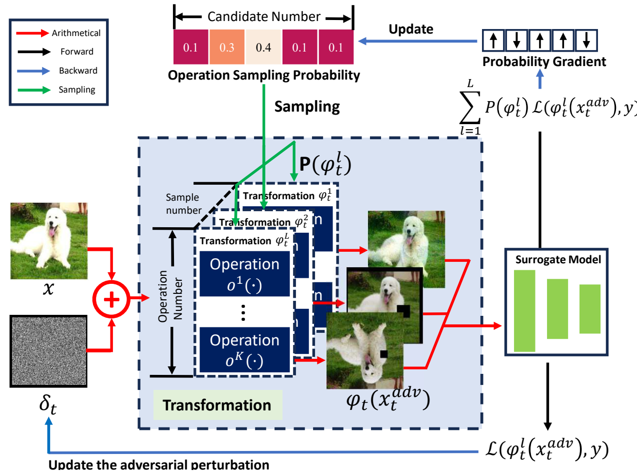

Figure 4. Overview of the pipeline in L2T. We use probability in sampling L transformations and update this probability through gradient ascent.

4. Experiments

4.1. Setup

Models. We evaluate the proposed method in three categories of target models. (1) Normally trained model: We select ten well-known models for experiments. ResNet-18 [15], ResNet-101 [15], ResNext-50 [59], DenseNet-121 [17], Inception-v3 [39], and Inception-v4 [39], ViT-B [9], PiT [16], Visformer [5], and Swin [30]. All of these models are pre-trained on the ImageNet dataset. (2) Adversarial trained models: we select four defense methods in our experiments. They are adversarial training (AT) [40], high-level representation guided denoiser (HGD) [24], neural representation purifier (NRP) [33], and randomized smoothing (RS) [6]. (3) Vision API: to imitate a practical scenario, we compare the attack performance on popular vision API. We chose Google Vision, Azure AI, GPT-4V, and Bard. For categories (2) and (3), we use ensemble-based attack. We choose two CNN-based models, ResNet18 and Inception-v4, and two transformer-based models, Visformer and Swin, to construct the ensemble surrogate model.

Dataset. Following previous works [46, 50, 57], we randomly choose 1,000 images from ILSVRC 2012 validation set [36]. All images are classified correctly by the models.

Baseline. We compare L2T with other input transformation adversarial methods. There are two categories of previous methods. The fixed transformation attack followed a fixed transformation scheme. We select TIM [8], SIM [26], Admix [46], DEM [67], IDE [58], Mask [10], S^{2}IM [31], BSR [42], and SIA [50] for comparison. The learned transformation attack followed a set of transformations predicted by a trained network to generate adversarial examples. We also compare our method with learned transformation at-

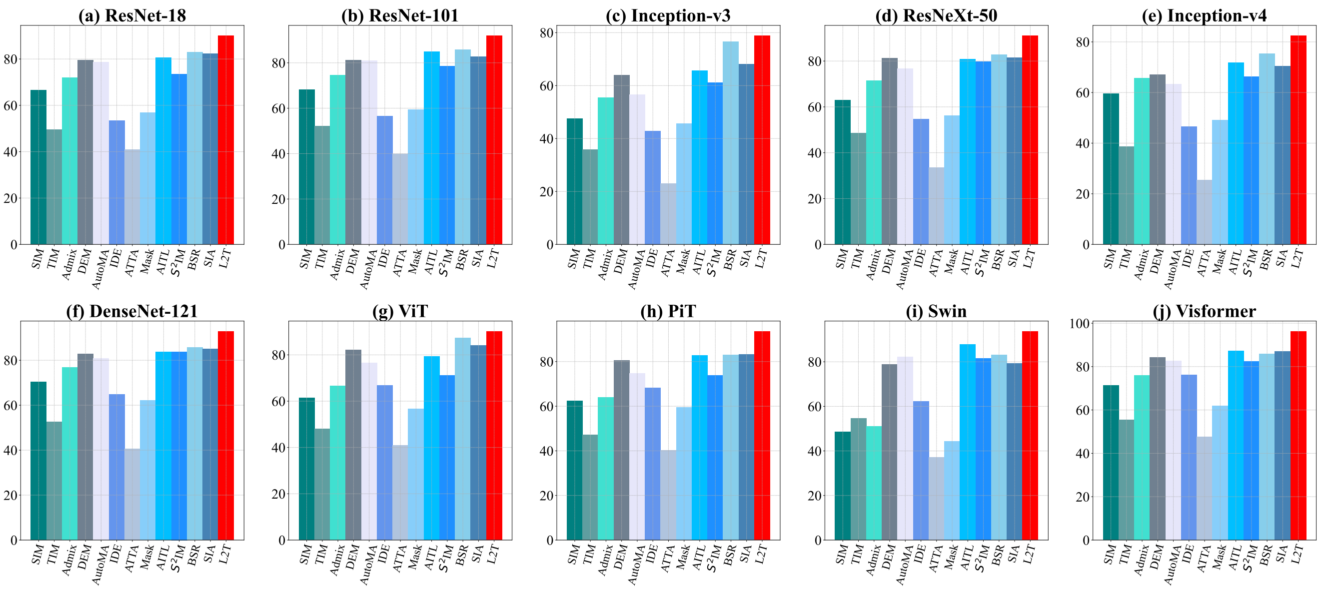

Figure 5. Average attack success rates (%) of ten models on the adversarial examples crafted on each model. The x-axis of each sub-figure denotes different attack methods. We include the detail number in our supplementary material.

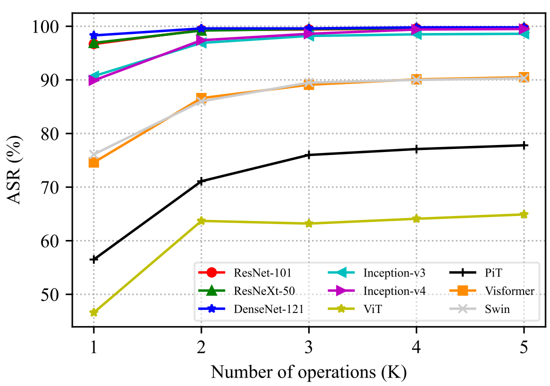

Figure 6. Attack success rates (ASR) (%) of adversarial examples generated by L2T with various number of operations K. We include the detail number in our supplementary material.

tacks, such as AutoMA [63], ATTA [53], and AITL [64]. All these methods are integrated with MI-FGSM [7] to generate adversarial examples.

Evaluation Settings. We follow the hyper-parameter setting of MI-FGSM and set the perturbation budget $\epsilon = 16$, number of iteration $T = 10$, step size $\alpha = \epsilon/T = 1.6$ and decay factor $\mu = 1$. For our method, we adopt the number of operations as 2, the number of samples as 10, and the learning rate ρ as 0.01. For the candidate operation, we chose ten categories of transformations. Each category contains ten specific operations with different parameters. We will discuss the detailed settings of our method and other baselines in the supplementary materials.

4.2. Evaluations on single models

Our proposed L2T exhibits better adversarial transferability to various input transformation based attacks. We take a single model as the surrogate model and evaluate the average attack success rate (ASR), *i.e.*, the average misclassification rates across ten models. We summarized our results in Figure 5. Each subfigure denotes the attacker generates the adversarial examples on the corresponding models and its x-axis denotes the attack algorithm used.

First, we observe that L2T consistently outperforms all other attackers, regardless of the surrogate model. Other baseline methods have various adversarial transferability according to the surrogate models. For example, the BSR performs to be the strongest baseline on ResNet-18. However, the BSR cannot remain efficient when the surrogate model is changed to Swin or PiT. In contrast, our proposed L2T is suitable for all the surrogate models being tested. These results also strengthen our argument that we should dynamically choose the transformation to fit the surrogate models. Specifically, in the worst case (subfig. c), our proposed L2T still outperforms the strongest baseline (S^2IM) by 2.1%. Overall, L2T outperforms the other baseline by 22.9% on average ASR.

4.3. Evaluations on defense methods

L2T is also capable of adversarial robust mechanisms. We test the attack performance of L2T against several defense mechanisms, including AT, HGD, NRP, and RS. We choose the ensemble setting to attack these defense approaches. We use the ensemble of four models, ResNet-18, Inception-v4,

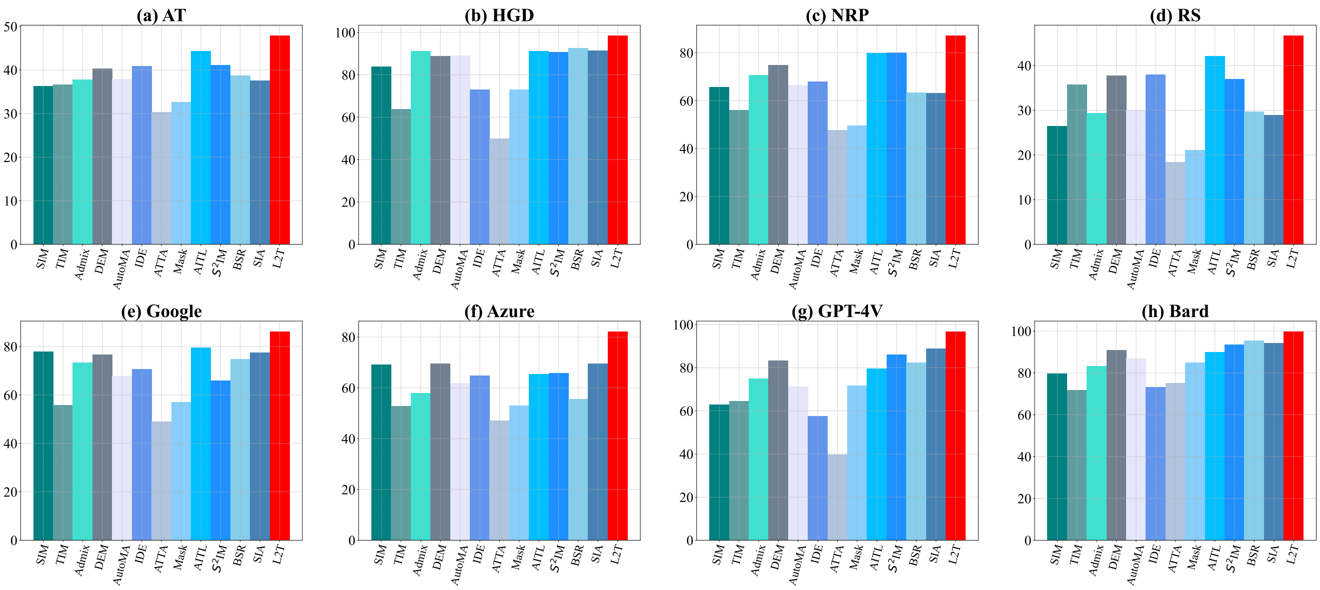

Figure 7. We integrate the ensemble-based attack with input transformation and evaluate the performance on defense methods and popular vision APIs. We include the detail number in our supplementary material.

Visformer, and Swin, as the surrogate model. We summarized our results in Figure 7 (a), (b), (c), and (d). Each subfigure denotes the model to be attacked and its x-axis denotes the attack algorithm used.

From Fig. 7, it is clear that L2T remains efficient. L2T consistently outperforms other methods against various defense methods. Notably, it achieves the attack success rate of 47.9%, 98.5%, 87.2%, and 46.7% on AT, HGD, NRP, and RS, respectively. Even on the certified defense RS, the strongest defense among the four, L2T achieves the attack success rate of 46.7%, which exceeds the best baseline (AITL) by 4.6%. This is also the biggest improvement L2T made compared to other defenses. This indicates that the dynamic of iteration also exists in the adversarial robust mechanism, which can be used to dimish the its performance.

4.4. Evaluations on vision API

Our proposed L2T can also perform well in realistic scenarios. To imitate the real-world application, we test the performance of L2T on Vision API. We use the same setting in sec. 4.3 to craft adversarial examples. We choose Google Vision (Figure 7 (e)) and Azure AI (Figure 7 (f)) to evaluate attacks on vision-only API. We also choose ChatGPT-4V (Figure 7 (g)) and Gemini (Figure 7 (f)) to evaluate attacks on the foundation model API.

As shown in Fig. 7, L2T is generally the best attacker to the real-world API. All attacks perform better on foundation model API than vision-only API. For vision-only API, L2T outperforms the strongest baseline by 8.7% and 12.6%, respectively. For foundation model API, L2T achieves nearly

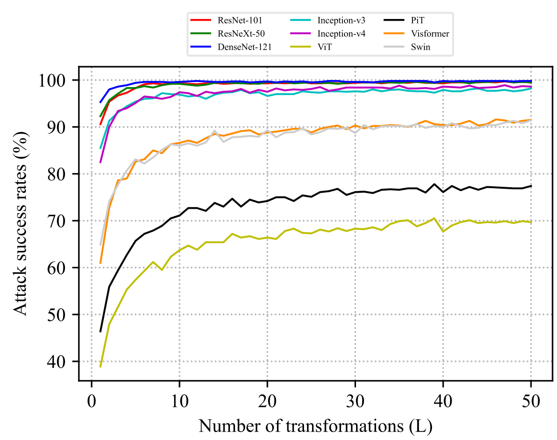

Figure 8. Attack success rates (ASR) (%) of adversarial examples generated by L2T with various number of transformations L. We include the detail number in our supplementary material.

100% attack success rate on both GPT-4V and Gemini.

4.5. Ablation study

On the numbers of operation K. As shown in Fig. 6, we study the impact of K on adversarial transferability. We craft the adversarial example on ResNet-18 and evaluate them on the other nine models. There is a clear difference between one operation and two operations. The average attack success rate increases by 8.09%, from 80.89% to 88.98%. However, when the $K \geq 3$, the improvement becomes marginal. The average attack success rate only

Table 1. Attack success rates (%) of adversarial examples by L2T and Rand (randomly choose transformation in each iteration).

ResNet-18	ResNet-101	ResNeXt-50	DenseNet-121	Inception-v3	Inception-v4	ViT	PiT	Visformer	swin
Rand	52.35	59.06	53.19	56.64	43.01	44.41	58.41	54.48	65.08
L2T (Ours)	90.00	91.90	91.00	92.80	78.80	82.40	90.10	93.50	96.20

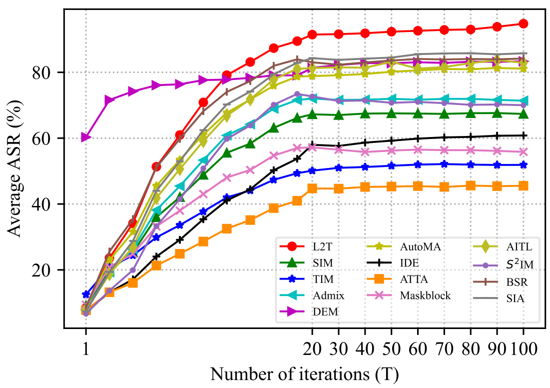

Figure 9. Average attack success rates (ASR) (%) of adversarial examples generated by L2T with various number of steps T. We include the detail number in our supplementary material.

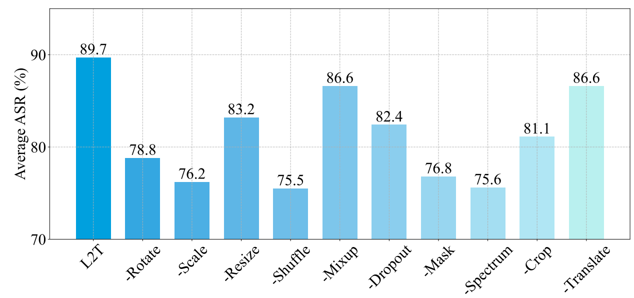

Figure 10. The average attack success rates (%) of adversarial examples crafted by L2T and L2T without a single transformation. $-$ indicates removing such transformation.

increases by 2.29% when K is increased from 2 to 5. Thus, K should be moderately settled as 2.

On the number of transformations L. We conducted experiments on the number of transformations L. We craft the adversarial example on ResNet-18 and evaluate them on the other nine models. We choose L from 1 to 50. From Fig. 8, we observe that the adversarial transferability improves steadily with the number of transformations. The increase is significant when the number of transformations grows from 1 to 20, which improves from an average attack success rate of 75.7% to an average attack success rate of 91.1%. However, transferability does not increase significantly after the number exceeds 20, where the average attack success rate only increases 1.5%. To keep the balance between computation efficiency and adversarial transferability, we suggest the number of samples set to 20.

On the number of iterations T. We discuss the number of iterations among different attack approaches. We craft the adversarial example on ResNet-18 and compare the average attack success rate of 10 models. As shown in Fig. 9, for all the attack methods, the attack success rate increased steadily for the first 10 iterations. L2T achieves the fastest speed of increase, which reaches 89.47% at iteration 10. After 10 iterations, most of the methods struggled to make improvements. For example, the Admix goes around 71%. The performance of S^2IM even decreases from 73% to 70%. Meanwhile, L2T still maintains a stable increase, from 89.47% to 94.77%.

Comparison with random sampling. We compare the learnable strategy with random sampling. As shown in Tab. 1, there is a clear gap of the attack success rate between random sampling and gradient-guided sampling. The minimum difference is 31.12% with setting Visformer as the surrogate model. For other surrogate models, the gap is even larger. This experiment indicates random sampling cannot effectively sample the best transformation trajectory, and the transformation in each iteration needs to be chosen carefully.

Operation candidates analysis. We conducted an ablation study for the operation candidates. We subtract each operation in the candidates and conduct L2T on the updated operation candidates. From Fig. 10, we observe that subtracting any operations will lead to a performance decrease. For example, by subtracting the scale operation, the performance decreases for 23.5%. Meanwhile, subtracting mixup and translation only results in a 3.1% decrease.

5. Conclusion

In this paper, we study the dynamic property for input transformation. Utilizing this property, we propose L2T to optimize the input transformation in each iteration. By updating a sampling probability, our method provides an approximate solution to input transformation optimization. Our experiments further study the effectiveness of our methods. Our method performs consistently well among different targeted models. This paper provides a new perspective to understand the transferability of adversarial examples.

Acknowledgement. This work was supported by NSF under grant 2202124 and the Center of Excellence in Data Science, an Empire State Development-designated Center of Excellence. The content of the information does not necessarily reflect the position of the Government, and no official endorsement should be inferred.

References

[1] Mihalj Bakator and Dragica Radosav. Deep learning and medical diagnosis: A review of literature. *Multimodal Technologies and Interaction*, 2(3):47, 2018. 1

[2] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models. In *Proceedings of the International Conference on Learning Representations*, 2018. 2

[3] Raja Chatila, Virginia Dignum, Michael Fisher, Fosca Giannotti, Katharina Morik, Stuart Russell, and Karen Yeung. Trustworthy ai. *Reflections on Artificial Intelligence for Humanity*, pages 13–39, 2021. 1

[4] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26, 2017. 2

[5] Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, and Qi Tian. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 589–598, 2021. 5

[6] Jeremy Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified Adversarial Robustness via Randomized Smoothing. In *Proceedings of the International Conference on Machine Learning*, pages 1310–1320, 2019. 3, 5

[7] 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

[8] 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. 1, 2, 5

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[10] Mingyuan Fan, Cen Chen, Ximeng Liu, and Wenzhong Guo. Maskblock: Transferable adversarial examples with bayes approach. *arXiv preprint arXiv:2208.06538*, 2022. 2, 5

[11] Zhijin Ge, Fanhua Shang, Hongying Liu, Yuanyuan Liu, Liang Wan, Wei Feng, and Xiaosen Wang. Improving the Transferability of Adversarial Examples with Arbitrary Style Transfer. *arXiv preprint arXiv:2308.10601*, 2023. 2

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 1

[13] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 2

[14] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Arthur Mann, and Pushmeet Kohli. Scalable Verified Training for Provably Robust Image Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4841–4850, 2019. 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[16] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11936–11945, 2021. 5

[17] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely Connected Convolutional Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2261–2269, 2017. 5

[18] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box Adversarial Attacks with Limited Queries and Information. In *Proceedings of the International Conference on Machine Learning*, pages 2142–2151, 2018. 2

[19] Jinyang Jiang, Zeliang Zhang, Chenliang Xu, Zhaofei Yu, and Yijie Peng. One forward is enough for neural network training via likelihood ratio method. In *The Twelfth International Conference on Learning Representations*, 2023. 1

[20] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 2

[21] Huichen Li, Xiaojun Xu, Xiaolu Zhang, Shuang Yang, and Bo Li. QEBA: Query-Efficient Boundary-Based Blackbox Attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1218–1227, 2020. 2

[22] 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 International Conference on Machine Learning*, pages 3866–3876, 2019. 2

[23] Yingwei Li, Song Bai, Yuyin Zhou, Cihang Xie, Zhishuai Zhang, and Alan L. Yuille. Learning Transferable Adversarial Examples via Ghost Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11458–11465, 2020. 1

[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/CVF Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018. 3, 5

[25] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and

Daan Wierstra. Continuous Control with Deep Reinforcement Learning. In *Proceedings of the International Conference on Learning Representations*, 2016. 1

[26] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *arXiv preprint arXiv:1908.06281*, 2019. 1, 2, 5

[27] Aoming Liu, Zehao Huang, Zhiwu Huang, and Naiyan Wang. Direct differentiable augmentation search. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12219–12228, 2021. 4

[28] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into Transferable Adversarial Examples and Black-box Attacks. In *Proceedings of the International Conference on Learning Representations*, 2017. 1

[29] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature Distillation: DNN-Oriented JPEG Compression Against Adversarial Examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 860–868, 2019. 3

[30] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 5

[31] Yuyang Long, Qilong Zhang, Boheng Zeng, Lianli Gao, Xianglong Liu, Jian Zhang, and Jingkuan Song. Frequency domain model augmentation for adversarial attack. In *European Conference on Computer Vision*, pages 549–566. Springer, 2022. 2, 5

[32] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Proceedings of the International Conference on Learning Representations*, 2018. 2, 3

[33] Muzammal Naseer, Salman H. 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 259–268, 2020. 3, 5

[34] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified Defenses against Adversarial Examples. In *Proceedings of the International Conference on Learning Representations*, 2018. 3

[35] Jonathan G Richens, Ciarán M Lee, and Saurabh Johri. Improving the accuracy of medical diagnosis with causal machine learning. *Nature communications*, 11(1):3923, 2020. 1

[36] 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. 5, 12

[37] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015. 1

[38] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. Physical adversarial examples for object detectors. In *12th USENIX workshop on offensive technologies (WOOT 18)*, 2018. 3

[39] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4278–4284, 2017. 5

[40] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick D. McDaniel. Ensemble Adversarial Training: Attacks and Defenses. In *Proceedings of the International Conference on Learning Representations*, 2018. 3, 5

[41] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018. 1

[42] Kunyu Wang, Xuanran He, Wenxuan Wang, and Xiaosen Wang. Boosting adversarial transferability by block shuffle and rotation. *arXiv preprint arXiv:2308.10299*, 2023. 5

[43] Kunyu Wang, Xuanran He, Wenxuan Wang, and Xiaosen Wang. Boosting Adversarial Transferability by Block Shuffle and Rotation. *arXiv preprint arXiv:2308.10299*, 2023. 2

[44] 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. 1, 2

[45] Xiaosen Wang, Kun He, Chuanbiao Song, Liwei Wang, and John E Hopcroft. AT-GAN: An adversarial generator model for non-constrained adversarial examples. *arXiv preprint arXiv:1904.07793*, 2019. 2

[46] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the Transferability of Adversarial Attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16138–16147, 2021. 1, 2, 5

[47] Xiaosen Wang, Jiadong Lin, Han Hu, Jingdong Wang, and Kun He. Boosting Adversarial Transferability through Enhanced Momentum. In *Proceedings of the British Machine Vision Conference*, page 272, 2021. 1, 2

[48] Xiaosen Wang, Chuanbiao Song, Liwei Wang, and Kun He. Multi-stage Optimization Based Adversarial Training. *arXiv preprint arXiv:2106.15357*, 2021. 3

[49] Xiaosen Wang, Zeliang Zhang, Kangheng Tong, Dihong Gong, Kun He, Zhifeng Li, and Wei Liu. Triangle Attack: A Query-Efficient Decision-Based Adversarial Attack. In *Proceedings of the European Conference on Computer Vision*, pages 156–174, 2022. 2

[50] Xiaosen Wang, Zeliang Zhang, and Jianping Zhang. Structure Invariant Transformation for better Adversarial Transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 2, 5

[51] Xingxing Wei, Siyuan Liang, Ning Chen, and Xiaochun Cao. Transferable Adversarial Attacks for Image and Video Object

Detection. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 954–960, 2019. 2

[52] 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 *Proceedings of the International Conference on Learning Representations*, 2020. 1

[53] Weibin Wu, Yuxin Su, Michael R Lyu, and Irwin King. Improving the transferability of adversarial samples with adversarial transformations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9024–9033, 2021. 2, 6

[54] Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating Adversarial Examples with Adversarial Networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 3905–3911, 2018. 2

[55] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan L. Yuille. Mitigating Adversarial Effects Through Randomization. In *Proceedings of the International Conference on Learning Representations*, 2018. 3

[56] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He. Feature Denoising for Improving Adversarial Robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 501–509, 2019. 3

[57] 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. 1, 2, 5

[58] Pengfei Xie, Linyuan Wang, Ruoxi Qin, Kai Qiao, Shuhao Shi, Guoen Hu, and Bin Yan. Improving the transferability of adversarial examples with new iteration framework and input dropout. *arXiv preprint arXiv:2106.01617*, 2021. 5

[59] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5987–5995, 2017. 5

[60] Yifeng Xiong, Jiadong Lin, Min Zhang, John E Hopcroft, and Kun He. Stochastic variance reduced ensemble adversarial attack for boosting the adversarial transferability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14983–14992, 2022. 1

[61] Weilin Xu, David Evans, and Yanjun Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *Proceedings of the Network and Distributed System Security Symposium*, 2018. 3

[62] Yichen Yang, Xiaosen Wang, and Kun He. Robust Textual Embedding against Word-level Adversarial Attacks. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, page 2214–2224, 2022. 3

[63] Haojie Yuan, Qi Chu, Feng Zhu, Rui Zhao, Bin Liu, and Neng-Hai Yu. Automa: towards automatic model augmentation for transferable adversarial attacks. *IEEE Transactions on Multimedia*, 2021. 2, 6

[64] Zheng Yuan, Jie Zhang, and Shiguang Shan. Adaptive image transformations for transfer-based adversarial attack. In *European Conference on Computer Vision*, pages 1–17. Springer, 2022. 2, 4, 6

[65] Zeliang Zhang, Jinyang Jiang, Minjie Chen, Zhiyuan Wang, Yijie Peng, and Zhaofei Yu. A novel noise injection-based training scheme for better model robustness. *arXiv preprint arXiv:2302.10802*, 2023. 1

[66] Zeliang Zhang, Wei Yao, Susan Liang, and Chenliang Xu. Random smooth-based certified defense against text adversarial attack. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1251–1265, 2024. 1

[67] Junhua Zou, Zhisong Pan, Junyang Qiu, Xin Liu, Ting Rui, and Wei Li. Improving the transferability of adversarial examples with resized-diverse-inputs, diversity-ensemble and region fitting. In *European Conference on Computer Vision*, pages 563–579. Springer, 2020. 5