

# ViT-EnsembleAttack: Augmenting Ensemble Models for Stronger Adversarial Transferability in Vision Transformers

Hanwen Cao\*, Haobo Lu\*, Xiaosen Wang, Kun He<sup>†</sup>

School of Computer Science and Technology

Huazhong University of Science and Technology

{hanwen, haobo, brooklet60}@hust.edu.cn, xswanghuster@gmail.com

## Abstract

*Ensemble-based attacks have been proven to be effective in enhancing adversarial transferability by aggregating the outputs of models with various architectures. However, existing research primarily focuses on refining ensemble weights or optimizing the ensemble path, overlooking the exploration of ensemble models to enhance the transferability of adversarial attacks. To address this gap, we propose applying adversarial augmentation to the surrogate models, aiming to boost overall generalization of ensemble models and reduce the risk of adversarial overfitting. Meanwhile, observing that ensemble Vision Transformers (ViTs) gain less attention, we propose ViT-EnsembleAttack based on the idea of model adversarial augmentation, the first ensemble-based attack method tailored for ViTs to the best of our knowledge. Our approach generates augmented models for each surrogate ViT using three strategies: Multi-head dropping, Attention score scaling, and MLP feature mixing, with the associated parameters optimized by Bayesian optimization. These adversarially augmented models are ensembled to generate adversarial examples. Furthermore, we introduce Automatic Reweighting and Step Size Enlargement modules to boost transferability. Extensive experiments demonstrate that ViT-EnsembleAttack significantly enhances the adversarial transferability of ensemble-based attacks on ViTs, outperforming existing methods by a substantial margin. Code is available at <https://github.com/Trustworthy-AI-Group/TransferAttack>.*

## 1. Introduction

Deep Neural Networks (DNNs), including Convolutional Neural Networks (CNNs) [14] and Vision Transformers (ViTs) [5], are inherently vulnerable to adversar-

ial attacks [10, 41], despite their impressive performance in solving various computer vision tasks. Adversarial examples, carefully designed to deceive DNNs, can be transferred between different models [22, 38], which means that a perturbation generated on a surrogate model can also mislead other models, even those with different architectures. This transferability enables a type of adversarial attack known as transfer-based attacks. Transfer-based adversarial examples are trained on surrogate models and can effectively attack unknown target models. To mitigate the gap between surrogate models and target models, recent researches [18, 21, 36, 37, 48] have introduced various techniques to improve transferability, such as input transformations [9, 21, 39] and advanced objective functions [18, 46].

Ensemble-based attacks [22] combine the outputs of multiple surrogate models to generate adversarial examples. These attacks can be easily integrated with existing transfer-based methods, such as gradient-based MI-FGSM [3] or NI-FGSM [20], and input transformation methods like TI-FGSM [4], to further enhance attack performance. Earlier approaches [22] simply average the outputs of ensemble models, yielding modest transferability. Subsequent work has focused on reducing discrepancies among surrogate models and adjusting ensemble weights. For instance, Stochastic Variance Reduced Ensemble adversarial attack (SVRE) [45] utilizes the idea of Stochastic Variance Reduced Gradient (SVRG) [16] to reduce the variances of gradient updates; Adaptive Model Ensemble Adversarial Attack (AdaEA) [1] and Stochastic Mini-batch black-box attack with Ensemble Reweighting (SMER) [32] dynamically adjust model weights based on adversarial contribution.

These methods have enhanced transferability by optimizing the combination of fixed surrogate models. However, we think it is not enough to merely focus on how to optimize the combination. Prior works don't investigate the potential contributions of surrogate models themselves in enhancing attack transferability. In other words, original surrogate models may not be the most effective surrogates for ensemble-based attacks. This gap motivates

\*The first two authors contributed equally.

<sup>†</sup>Corresponding author.

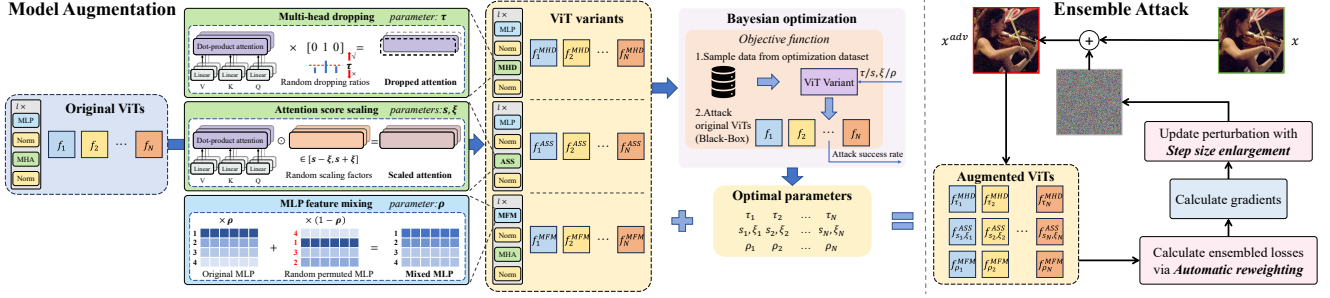


Figure 1. Overview of the proposed ViT-EnsembleAttack framework. The models  $f_1, \dots, f_N$  represent the  $N$  original surrogate ViTs. Unlike traditional ensemble-based attacks, ViT-EnsembleAttack generates a set of augmented models using three strategies with parameters optimized by Bayesian optimization, and ensembles these augmented models to produce adversarial examples.

our approach of augmenting ensemble models adversarially. It is noteworthy that model augmentation can be achieved through various approaches. Our approach focuses on increasing model diversity by introducing randomness into the model inference process. This method requires designing randomization strategies tailored to the characteristics of the models and, more importantly, confirming the optimal degree of randomness. In ensemble-based attacks, where multiple surrogate models are available, we can apply this augmentation to each individual surrogate. We then treat the others as black-box models to evaluate the transferability of the augmented model. Higher transferability indicates a more suitable degree of randomness. By doing this, all of the augmented surrogates can generate more diverse backpropagation paths for the same input than original surrogates, guiding the update of perturbations and thereby reducing the risk of adversarial overfitting.

Given the superior performance of ViTs over CNNs in many tasks, we focus on designing an attack framework specifically for ViTs, which is less explored in existing works. We propose a novel ensemble-based attack, termed ViT-EnsembleAttack, against ViTs from the perspective of adversarially augmenting the ensemble models. Specifically, we draw inspiration from three data augmentation strategies—masking, scaling, and mixup—and propose three corresponding augmentation strategies for ViTs: Multi-head dropping (MHD), Attention score scaling (ASS), and MLP feature mixing (MFM). Each original surrogate ViT will be modified through these strategies and generate three variants. These variants are parameterized and will be optimized by Bayesian optimization to become augmented ViTs, which will be used as new surrogate models. Additionally, we propose Automatic Reweighting to adjust the ensemble weights dynamically and Step Size Enlargement to accelerate convergence during the attack. The overview of ViT-EnsembleAttack is illustrated in Figure 1.

The main contributions of this work are as follows:

- We introduce a novel perspective to improve ensemble-based attack transferability by adversarially augmenting

the surrogate models and propose, to the best of our knowledge, the first ensemble-based attack tailored for ViTs.

- We design three augmentation strategies tailored to the structure of ViTs and utilize Bayesian optimization to fine-tune the optimal parameters. We further introduce Automatic Reweighting and Step Size Enlargement to improve the attack’s efficiency.
- Comprehensive experiments validate the superior performance of ViT-EnsembleAttack in enhancing the adversarial transferability. Notably, our approach outperforms the state-of-the-art baseline by a clear margin of 15.3% attack success rate on average when attacking CNNs.

## 2. Related Work

### 2.1. Adversarial Attacks

**Gradient-based attacks.** Adversarial attacks differ from standard gradient descent, as they typically employ gradient ascent to reverse the optimization effect. Goodfellow *et al.* [10] introduced the Fast Gradient Sign Method (FGSM), which generates adversarial perturbation in a single step. Based on this, Kurakin *et al.* [17] and Dong *et al.* [3] proposed iterative versions of FGSM, the latter introducing momentum to stabilize the update direction. Although these methods achieve high performance in white-box settings, they struggle to maintain the same transferability in black-box settings, where information about the target model is typically unavailable.

**Transfer-based attacks.** Several approaches have been explored to improve adversarial transferability [3, 8, 22, 47]. Xie *et al.* [44] and Lin *et al.* [20] combined the gradients of the augmented examples using resizing and scaling techniques to create diverse input patterns for higher transferability. Ganeshan *et al.* [7] disrupted the deep features within DNNs, while Zhang *et al.* [46] extended this idea by calculating feature importance for each neuron. Li *et al.* [19] targets ghost networks generated through aggressive dropout applied to intermediate features, and Wang *et*

al. [42] mitigated gradient truncation by recovering gradients lost due to non-linear activation functions. Although transfer-based attacks show promising performance in enhancing adversarial transferability between CNNs, their attack success rate diminishes when transferring to ViTs, which are known to exhibit greater robustness [41].

**Ensemble-based attacks.** Ensemble-based methods fuse outputs of multiple models to enhance the effectiveness of transfer-based attacks. Among the three common ensemble approaches, *i.e.* ensemble on predictions, ensemble on losses, and ensemble on logits, Dong *et al.* [4] showed that the latter is the most effective. Xiong *et al.* [45] proposed the SVRE method to reduce the variance among the ensemble models utilizing the idea of SVRG [16] method. Chen *et al.* [1] introduced AdaEA, which adaptively adjusts the contribution of each model in the ensemble and synchronizes update directions through a disparity-reduced filter, aiming to bridge the gap between CNNs and ViTs. Tang *et al.* [32] proposed SMER, which generates stochastic mini-batch perturbations to enhance ensemble diversity and utilizes reinforcement learning to adjust ensemble weights. In contrast, ViT-EnsembleAttack focuses on optimizing the surrogate models themselves rather than the ensemble path, by exploiting unique augmentations specific to ViTs.

## 2.2. Adversarial Defenses

Various approaches have been proposed to defend against adversarial attacks and improve the robustness of DNNs. Adversarial training [35] is one of the most effective techniques, where clean images and their corresponding adversarial examples are incorporated into the training process. Another category of adversarial defense focuses on input transformation techniques, which disrupt the adversarial pattern by preprocessing the input data. Popular methods in this category include reversing adversarial features [28], randomly resizing [43], utilizing compression techniques [12], and purifying inputs with GANs [28] or diffusion models [40]. In this work, we select some defensive models as target models to assess the effectiveness of the proposed ViT-EnsembleAttack compared to existing SOTA baselines.

## 3. Methodology

### 3.1. Preliminaries

Given a clean image  $x$  with the ground-truth label  $y$ , a surrogate ViT model  $f$ , the goal of the adversarial attack is to generate an adversarial image  $x^{adv} = x + \delta$  to mislead the model  $f$ , *i.e.*,  $f(x^{adv}) \neq f(x) = y$ , where  $\delta$  is the additive perturbation. A set of boundary conditions are imposed on the perturbation to make it imperceptible in relation to the clean example, *i.e.*,  $\|\delta\|_p < \epsilon$ , where  $\|\cdot\|_p$  represents the  $L_p$  norm. To align with previous works, we employ  $p = \infty$  for

the following comparisons. Therefore, the iterative attack process on a single surrogate model can be described as:

$$x_{t+1}^{adv} = x_t^{adv} + \alpha \cdot \text{sign}(\nabla_{x_t^{adv}} J(f(x_t^{adv}), y)), \quad (1)$$

where  $\alpha$  is step size,  $J$  is the loss function,  $\text{sign}(\cdot)$  denotes the sign function,  $x_t^{adv}$  denotes the adversarial example in  $t^{th}$  iteration and  $\nabla_{x_t^{adv}} J(f(x_t^{adv}), y)$  is the gradient of the loss function *w.r.t.*  $x_t^{adv}$ .

Ensemble-based attacks utilize the output of multiple surrogate models and usually average them to obtain loss. Assuming that there are  $N$  surrogate models, the generation process of adversarial examples can be described as:

$$x_{t+1}^{adv} = x_t^{adv} + \alpha \cdot \text{sign}\left(\sum_{i=1}^N w_i \cdot \nabla_{x_t^{adv}} J(f_i(x_t^{adv}), y)\right), \quad (2)$$

where  $w_i \geq 0$  is the ensemble weight of each ensemble model  $f_i$  and satisfies  $\sum_{i=1}^N w_i = 1$ .

### 3.2. Motivation

Since the effectiveness of transferable adversarial attacks has been shown to be highly correlated with the diversity of the model [1, 19], we argue that ensemble models can be adversarially augmented to be more diverse, thus further enhancing their adversarial transferability. This inspires us to treat the ensemble models as tunable components, rather than fixed components as assumed in other studies. Following this principle, we introduce ViT-EnsembleAttack, the first ensemble-based attack method tailored for ViTs to the best of our knowledge.

### 3.3. The ViT-EnsembleAttack Method

The ViT-EnsembleAttack method consists of three modules: Model Augmentation, Automatic Reweighting, and Step Size Enlargement. Detailed descriptions of these modules are provided below.

**Model Augmentation.** A typical ViT model consists of alternating layers of multi-head self-attention (MSA) and multi-layer perceptron (MLP) blocks. To augment surrogate ViTs, we adapt three data-augmentation-inspired strategies on these special modules, namely **Multi-head dropping**, **Attention score scaling**, and **MLP feature mixing**. We also design **Parameter optimization** process to identify the optimal parameters. Detailed descriptions are provided below.

**Multi-head dropping (MHD)** means randomly abandoning some heads in each MSA. In practice, we set a threshold  $\tau \in [0, 1]$  to determine whether to drop the head. Each head in each MSA of the surrogate ViTs will be independently assigned a random probability from 0 to 1 following a uniform distribution. Heads with lower probabilities than  $\tau$  will be dropped, *i.e.*, the attention score matrix in this head

**Algorithm 1** Objective function for Bayesian optimization

**Input:** Parameter(s)  $p$ , augmentation strategy  $c$ , surrogate model  $f$ , test models set  $F = \{f_1, \dots, f_{N-1}\}$ , images for Bayesian optimization  $X^B$  with corresponding ground-truth label  $Y^B$ , the number of randomly sampled images  $M$ .

**Output:** Average attack success rate.

- 1: Random choose  $M$  images from  $X^B$  and their corresponding labels to compose the attack datasets.
- 2: Modify  $f$  to  $f_p^c$  according to  $c$  and  $p$ .
- 3: Using MI-FGSM algorithm generate adversarial examples  $\{x_1^{adv}, \dots, x_M^{adv}\}$  on  $f_p^c$ .
- 4: Calculate the average attack success rate of  $\{x_1^{adv}, \dots, x_M^{adv}\}$  on test models  $F$ .
- 5: **return** Average attack success rate.

becomes an all-zero matrix. Here  $\tau$  is the corresponding parameter to be optimized.

*Attention score scaling (ASS)* means that for each attention score matrix, we generate a matrix with random scaling factors  $\in [s-\xi, s+\xi]$  following a uniform contribution. The scaling matrix has the same shape with the attention score matrix to make element-wise multiplication. Here  $s, \xi$  are the corresponding parameters to be optimized.

*MLP feature mixing (MFM)* randomly permutes the feature representations of MLP to form a new matrix. Then mix the vanilla MLP matrix with  $(1 - \rho)$  and the new matrix with  $\rho$  as the final output. Here  $\rho$  is the parameter to be optimized.

*Parameter optimization.* Each surrogate model  $f_i$  can generate three variants  $f_{i,p_i}^c$  with the above strategies, where  $c \in \{MHD, ASS, MFM\}$  means the augment strategy,  $p_i \in \{\tau_i, (s_i, \xi_i), \rho_i\}$  means the corresponding parameter(s). For simplicity, we use  $f_{p_i}^c$  in place of  $f_{i,p_i}^c$ . We employ Bayesian optimization to optimize parameters for these variants. The most important aspect of Bayesian optimization is a well-designed objective function that guides the search process. In our method, we generate adversarial examples on  $f_{p_i}^c$  and attack the other original surrogates  $\{f_1, \dots, f_{i-1}, f_{i+1}, \dots, f_N\}$ . The average attack success rate on target models is set as the output of objective function, with the purpose of enhancing the transferability of the select model  $f_{p_i}^c$ . Details of the objective function are listed in Algorithm 1. For convenience, we use *gp\_minimize* function in Python library *skopt* to build this Bayesian optimization process. We denote the number of calls to the objective function as  $n_{calls}$ , the parameter selection space as  $P$ , and the remaining parameters of *gp\_minimize* are set as default.

**Automatic Reweighting.** Due to the difference in inner architecture between surrogate models, the loss calculated on each model will exhibit different magnitudes. It is more

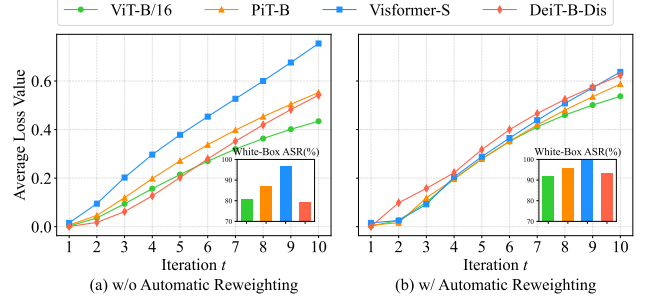


Figure 2. Comparison of average loss values during the attack process for ViT-B/16, PiT-B, Visformer-S, and DeiT-B-Dis over 10 iterations, (a) without and (b) with Automatic Reweighting, with embedded bar charts showing the final white-box attack success rate (ASR) for each surrogate model.

likely that adversarial examples will overfit to the models with larger loss values because they play a more important role in the backpropagation of gradients. Figure 2 (a) shows when averaging the ensemble weights, Visformer-S has the largest loss value and it also achieves the highest attack success rate of nearly 100%. However, models with low loss values, such as ViT-B/16 and DeiT-B-Dis, achieve less than 80% attack success rate.

To mitigate this issue, we propose an Automatic Reweighting module to balance the contribution of each model to the loss calculation. Specifically, we record the loss values of all surrogate models at each iteration and assign weights to each model according to the following equation:

$$w_i = \frac{\left(\frac{L_{max}}{L_i}\right)^b}{\sum_{j=1}^N \left(\frac{L_{max}}{L_j}\right)^b}, \quad (3)$$

where  $L_{max} = \max\{L_1, \dots, L_N\}$  is the maximum loss among all surrogate models,  $L_i$  denotes the loss of the  $i$ -th model  $f_i$ , and  $b$  is the hyper-parameter. Figure 2 (b) provides the loss value and attack performance with Automatic Reweighting. The results demonstrate that it effectively reduces discrepancy in loss magnitudes across surrogate models and enhances the white-box attack success rate, especially for those with low loss values originally.

**Step Size Enlargement.** Traditionally, the step size  $\alpha$  in each iteration is set to  $\frac{\epsilon}{T}$ , where  $\epsilon$  is the maximum perturbation and  $T$  is the number of attack iterations. However, as shown in Figure 2 (a), we find that while using the basic ensemble attack setting (Ens), ensemble models retain a large margin to 100% white-box attack success rate, indicating that the attack process has not converged yet. Hence, we propose Step Size Enlargement to enhance the attack strength and accelerate the convergence process. Specifically, we set the step size as  $\alpha = \frac{q \cdot \epsilon}{T}$ , and  $q$  is the hyper-parameter. We do comprehensive ablation studies to test the attack performance under different  $q$  and validate that a



---

**Algorithm 2** ViT-EnsembleAttack

---

**Input:** Loss function  $J$ , surrogate models  $\{f_1, \dots, f_N\}$ , a clean image  $x$  with ground-truth label  $y$ , the maximum perturbation  $\epsilon$ , number of iterations  $T$ , inference times  $loop$ , step size enlargement times  $q$ , momentum decay factor  $\mu$ , objective function  $OF$ , Bayesian optimization function  $gp\_minimize$ , parameter selection space  $P$ , the number of calls to the objective function  $n_{calls}$ .

**Output:** Adversarial images  $x^{adv}$ .

```
1: # Phase 1: Model Augmentation
2: for  $i=0$  to  $N-1$  do
3:   Set  $F = \{f_1, \dots, f_{i-1}, f_{i+1}, \dots, f_N\}$ .
4:   Build Bayesian optimization process
      $gp\_minimize(n_{calls}, P, OF(p \in P, c, f_i, F))$ 
5:    $\tau_i = gp\_minimize(c = MHD)$ 
6:    $s_i, \xi_i = gp\_minimize(c = ASS)$ 
7:    $\rho_i = gp\_minimize(c = MFM)$ 
8: end for
9: # Phase 2: Ensemble Attack
10: Set step size  $\alpha = \frac{q\epsilon}{T}$ ,  $g_0 = 0$ ,  $x_0^{adv} = x$ .
11: for  $t = 0$  to  $T - 1$  do
12:   for  $i = 0$  to  $N - 1$  do
13:     for  $j = 0$  in  $loop - 1$  do
14:        $L_i = J(f_{\tau_i}^{MHD}(x_t^{adv}), y) + J(f_{s_i, \xi_i}^{ASS}(x_t^{adv}), y)$ 
15:        $+ J(f_{\rho_i}^{MFM}(x_t^{adv}), y)$ 
16:     end for
17:   end for
18:   Calculate  $\{w_1, \dots, w_N\}$  using Eq. (3).
19:    $g_{t+1} = \nabla_{x_t^{adv}}(\sum_{i=1}^N w_i \cdot L_i)$ 
20:    $g_{t+1} = \mu \cdot g_t + \frac{g_{t+1}}{\|g_{t+1}\|_1}$ 
21:    $x_{t+1}^{adv} = x_t^{adv} + \alpha \cdot \text{sign}(g_{t+1})$ 
22: end for
23: return  $x^{adv}$ 
```

---

large step size leads to high transferability.

**Overall attack framework.** We present the details of ViT-EnsembleAttack in Algorithm 2, and there are two aspects that should be highlighted. First, to take full advantage of the randomness of our method and improve the diversity of ensemble models, we perform inference  $loop$  times for the augmented models. Second, model augmentation and ensemble attack are two independent processes. Note that the model augmentation is a pre-process that takes only once. When generating adversarial examples, most of the time consumption depends on the number of ensemble models and the inference times.

## 4. Experiments

In this section, we begin by detailing our experimental setup, then compare our method with the latest adversarial ensemble attacks against ViTs and CNNs. This com-

parison highlights the effectiveness of our method in enhancing ensemble transferability between ViTs as well as cross-structure transferability. We also do ablation studies on the modules of ViT-EnsembleAttack, hyperparameters  $q$ ,  $b$ ,  $loop$ , and resource consumption. Finally, we further analyze the effect of each augmentation strategy on the transferability of adversarial examples.

### 4.1. Experimental Setup

We compare the performance of ViT-EnsembleAttack with existing state-of-the-art methods against the normally trained ViTs, robust ViTs, adversarially trained ViTs, normally trained CNNs, adversarially trained CNNs, and a hybrid model, respectively. Our experiments concentrate on the image classification task.

**Dataset.** We randomly sample 1000 images from the ILSVRC 2012 validation set [29] as the clean images to be attacked, then randomly sample another 4000 different images used for Bayesian optimization. We check that all of the surrogate and target models achieve almost 100% classification success rate on the two sampled datasets.

**Models.** We choose four representative ViT models as the surrogate models to generate adversarial examples, including ViT-B/16 [5], PiT-B [15], DeiT-B-Dis [33], and Visformer-S [2]. We evaluate the transferability of adversarial examples of ViTs under two attacking scenarios. One is that the surrogate and target models are both ViTs to validate the transferability across different ViTs. The other is that the surrogate models are ViTs, but the target models are CNNs to examine the cross-model structure transferability. For the first setting, the target ViT models contain four normally trained ViTs: CaiT-S/24 [34], TNT-S [13], LeViT-256 [11], ConViT-B [6], three robust ViTs: RVT-S\* [25], Drvit [23], Vit+Dat [24], and an adversarially trained ViT: ViT-B/16<sub>AT</sub> [27]. For the second setting, we select normally trained CNNs: Inception-v3 (Inc-v3) [30], Inception-v4 (Inc-v4) [31], Inception-Resnet-v2 (IncRes-v2) [31], Resnet-v2-152 (Res-v2) [14], adversarially trained models: an ensemble of three adversarial trained Inceptionv3 models (Inc-v3<sub>ens3</sub>) [35], an ensemble of four adversarial trained Inception-v3 models (Inc-v3<sub>ens4</sub>) [35], adversarial trained Inception-Resnet-v2 (IncRes-v2<sub>adv</sub>) [35] and a hybrid model MobileViTv2 (MViTv2) [26] which has both convolutional layers and ViT blocks as the target models.

**Comparisons and baselines.** We choose the ensemble attack (Ens), which updates adversarial examples using Eq (2) and average weights, and three SOTA methods, SVRE [45], AdaEA [1] and SMER [32], as the competitive baselines. All methods are integrated into four attack settings, including I-FGSM [17], MI-FGSM [3], DI-FGSM [44], and TI-FGSM [4].

**Evaluation metric.** The evaluation metric is the attack

Attack	Model	CaiT-S/24	TNT-S	LeViT-256	ConViT-B	RVT-S*	Drvit	Vit+DAT	ViT-B/16 <sub>AT</sub>
I-FGSM	Ens	63.6	60.9	48.6	61.4	47.8	59.1	50.4	97.6
	SVRE	94.1	90.2	74.3	92.9	75.9	88.6	84.3	97.7
	AdaEA	86.8	78.9	61.0	84.8	60.0	76.5	70.5	97.6
	SMER	95.3	90.4	78.6	94.1	79.7	90.0	86.4	97.8
	Ours	<b>99.1</b>	<b>98.1</b>	<b>95.4</b>	<b>99.0</b>	<b>92.7</b>	<b>97.4</b>	<b>97.1</b>	<b>97.9</b>
MI-FGSM	Ens	76.1	74.8	69.0	74.6	69.4	72.5	69.8	97.7
	SVRE	99.5	97.9	95.1	99.4	94.3	97.6	97.8	97.8
	AdaEA	96.4	93.7	86.3	95.9	86.8	93.8	92.4	97.8
	SMER	<b>99.7</b>	98.1	95.0	<b>99.5</b>	94.2	97.8	97.4	97.8
	Ours	99.5	<b>99.0</b>	<b>98.5</b>	99.3	<b>97.3</b>	<b>99.3</b>	<b>99.1</b>	<b>97.9</b>
DI-FGSM	Ens	78.0	78.5	73.7	76.5	72.5	74.6	69.0	97.4
	SVRE	98.9	98.5	96.7	98.5	95.1	97.8	96.0	97.8
	AdaEA	92.1	89.9	81.0	91.2	81.5	88.6	84.9	97.5
	SMER	99.0	98.0	96.9	98.6	96.0	98.3	96.4	97.8
	Ours	<b>99.9</b>	<b>100.0</b>	<b>99.8</b>	<b>100.0</b>	<b>99.7</b>	<b>100.0</b>	<b>99.4</b>	<b>98.0</b>
TI-FGSM	Ens	70.9	68.9	55.2	68.5	55.0	67.7	58.2	97.6
	SVRE	94.8	92.5	79.1	93.9	81.2	92.8	87.8	97.7
	AdaEA	90.2	84.9	67.1	88.5	68.1	84.4	77.5	97.8
	SMER	95.9	93.6	81.4	94.9	83.1	93.9	89.4	97.8
	Ours	<b>99.5</b>	<b>99.1</b>	<b>97.8</b>	<b>99.4</b>	<b>95.4</b>	<b>99.2</b>	<b>98.3</b>	<b>97.9</b>

Table 1. The attack success rates (%) against eight ViTs by various transfer-based ensemble attacks. The best results appear in bold.

Attack	Model	Inc-v3	Inc-v4	IncRes-v2	Res-v2	Inc-v3 <sub>ens3</sub>	Inc-v3 <sub>ens4</sub>	IncRes-v2 <sub>adv</sub>	MViT-v2
I-FGSM	Ens	38.8	38.4	32.6	34.6	26.1	23.1	17.9	30.2
	SVRE	63.3	61.9	55.1	54.9	46.3	41.6	32.9	50.9
	AdaEA	47.4	44.8	38.4	40.8	29.2	26.7	20.2	35.4
	SMER	64.9	62.5	57.5	58.4	48.7	46.0	37.6	53.7
	Ours	<b>90.3</b>	<b>88.1</b>	<b>84.5</b>	<b>84.6</b>	<b>76.8</b>	<b>70.7</b>	<b>61.8</b>	<b>79.5</b>
MI-FGSM	Ens	66.3	64.3	60.4	63.3	54.2	50.3	46.5	57.9
	SVRE	88.4	87.2	87.4	84.6	78.0	72.5	68.5	80.9
	AdaEA	76.5	77.3	73.4	73.0	66.9	62.4	59.0	69.8
	SMER	88.2	87.7	85.8	84.7	77.6	74.1	68.8	81.0
	Ours	<b>95.7</b>	<b>95.3</b>	<b>94.0</b>	<b>93.3</b>	<b>89.0</b>	<b>84.4</b>	<b>80.0</b>	<b>90.2</b>
DI-FGSM	Ens	67.3	67.1	60.9	62.1	54.1	50.9	45.9	58.3
	SVRE	91.9	92.2	90.9	87.1	82.9	80.4	76.7	86.0
	AdaEA	70.9	70.4	64.7	63.6	57.6	53.6	47.5	61.2
	SMER	93.4	92.7	91.1	87.7	84.1	82.0	76.8	86.8
	Ours	<b>99.0</b>	<b>99.2</b>	<b>98.3</b>	<b>97.0</b>	<b>97.2</b>	<b>96.1</b>	<b>93.8</b>	<b>97.2</b>
TI-FGSM	Ens	46.4	45.6	39.9	40.2	31.6	29.2	23.7	36.7
	SVRE	68.9	70.6	62.6	61.2	56.8	54.5	47.0	60.2
	AdaEA	55.1	53.0	47.0	47.7	38.2	35.4	29.4	43.7
	SMER	73.8	71.9	64.6	63.5	59.2	56.4	50.4	62.8
	Ours	<b>93.9</b>	<b>94.8</b>	<b>90.5</b>	<b>88.9</b>	<b>84.8</b>	<b>82.2</b>	<b>76.2</b>	<b>87.3</b>

Table 2. The attack success rates (%) against eight CNNs by various transfer-based ensemble attacks. The best results appear in bold.

success rate (ASR), the ratio of the adversarial examples that successfully mislead the target model among all samples.

**Hyper-parameters.** For a fair comparison, we follow the hyper-parameters setting in [32] to set the maximum

perturbation to  $\epsilon = 16$  and the number of iterations to  $T = 10$ , so the step size in other methods is  $\alpha = \frac{\epsilon}{T} = 1.6$ . Hyper-parameters of other methods follow their default settings. For the decay factor  $\mu$  in MI-FGSM, we set  $\mu$  to 1.0. For the translation kernel in TI-FGSM, we use the Gaus-

sian kernel, the size is  $5 \times 5$ . For transformation operation  $T(\cdot; p)$  in DI-FGSM, we set  $p = 0.5$  and the range of  $rnd$  is  $[224, 248)$ . We set  $n_{calls} = 50$ ,  $P = (0, 1)$  for  $gp\_minimize$  function. For the other hyper-parameters in ViT-EnsembleAttack, we set  $loop = 2$ ,  $q = 3$  and  $b = 2$ . All images are resized to  $224 \times 224$  to conduct experiments and set the patch size to 16 for the inputs of ViTs.

## 4.2. Transferability

Here we analyze the performance of our approach against ViTs and CNNs, respectively. Specifically, we generate adversarial examples on four given surrogate models and directly attack various target models to show the generalization of the proposed method.

**Performance on ViTs.** We first compare the general attack performance of ViT-EnsembleAttack with existing ensemble methods on the normally trained, robust and adversarially trained ViTs. As shown in Table 1, in the black-box setting, our method outperforms the state-of-the-art baselines by a large average margin of 4.6% attack success rate on average. Specifically, our method improves the attack success rate from 78.6% to 95.4% on LeViT-256 when integrating with I-FGSM. For DI-FGSM, our method achieves an attack success rate of nearly 100%, further demonstrating its effectiveness.

**Performance on CNNs.** We then attempt to evaluate the cross-structure transferability by attacking normally trained and adversarially trained CNNs. The results are summarized in Table 2. It can be seen that the attack success rate decreases a lot compared to attacking ViTs, illustrating the difficulty of cross-model structure transfer attack. Nevertheless, our method still achieves nearly 88.3% attack success rate on average, outperforming SMER by a significant margin of 15.3% on average, which represents a substantial advancement over prior methods, demonstrating the superior cross-structure transferability performance of our proposed ViT-EnsembleAttack.

## 4.3. Ablation Study

In this subsection, we analyze the contribution of each module and study the effects of several key hyper-parameters to justify our choices.

**On the modules of ViT-EnsembleAttack.** We integrate our method with all attack algorithms, utilizing various modules to craft adversarial examples, and report their transferability on ViTs and CNNs. As shown in Table 3, Model Augmentation module improves the attack success rate mostly, indicating its effectiveness in ViT-based ensemble attacks. Automatic Reweighting and Step Size Enlargement each surpass the baseline individually, and their combination outperforms either alone. When paired with augmentation, both techniques improve upon augmentation alone, with the best results achieved by combining all

Augmentation	Reweighting	Enlargement	ViTs	CNNs
-	-	-	70.5	48.1
✓	-	-	93.4	78.8
-	✓	-	78.6	52.6
-	-	✓	87.1	65.5
✓	✓	-	95.7	81.0
✓	-	✓	98.0	88.1
-	✓	✓	89.2	66.9
✓	✓	✓	98.4	88.3

Table 3. The average attack success rates (%) against ViTs and CNNs by various settings of modules. ✓ indicates that the module is applied. For simplicity, we only retain the last word of each module.

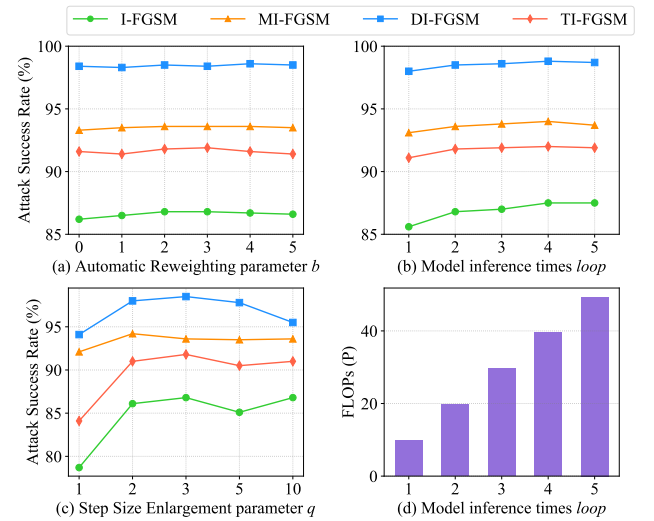


Figure 3. Average attack success rate against ViTs and CNNs under three varying parameters: (a) automatic reweighting parameter  $b$ , (b) model inference times  $loop$ , and (c) step size enlargement parameter  $q$ . (d) Computational cost (FLOPs) for different model inference times  $loop$ .

three, exceeding any single or pairwise setup. This outcome demonstrates that the three modules in ViT-EnsembleAttack are complement and combine each other could achieve the improvement of transferability.

**On hyper-parameter sensitivity.** We conduct a detailed analysis of the key hyper-parameters  $b$ ,  $q$ , and  $loop$  to explain the optimal configuration. As shown in Figure 3 (a), the variation in attack success rate with changes in  $b$ , except for  $b = 0$ , is not significant. We set  $b = 2$  as the final choice because it maintains high attack success rates across all algorithms, making it a balanced option. Figure 3 (c) illustrates that a moderate increase in  $q$  enhances attack success, with the peak performance observed at  $q = 3$  for most algorithms. However, beyond this point (e.g.,  $q = 5$  and  $q = 10$ ), the attack success rate declines, likely due

to instability caused by excessively large step sizes. Based on this observation, we select  $q = 3$  as the optimal value. Figure 3 (b) exhibits that increasing *loop* improves the attack success rate, but the gains become marginal beyond *loop* = 2. Meanwhile, Figure 3 (d) indicates that the computational cost grows exponentially with larger *loop* values. Given the trade-off between attack effectiveness and computational efficiency, we choose *loop* = 2 to balance performance and resource consumption.

Increasing the number of *loop* iterations improves attack success because our method uses model augmentation to inject randomness during inference, resulting in varied gradient estimates at each back-propagation. Accumulating these diverse directions over multiple rounds enhances transferability. Without model augmentation, repeated inference yields identical gradients. Thus, *loop* is designed to amplify the effect of model augmentation.

Table 4. Computational resource consumption of different methods. We report the result of our method into two phases, as described in Algorithm 2.

	Ens	SVRE	AdaEA	SMER	Ours	
					Phase1	Phase2
FLOPs (P)	3.290	29.623	18.653	61.309	54.069	19.738
Time (s)	395.2	3460.2	2176.0	7573.9	2394.7	2669.9

**On resource consumption.** In Table 4, we report both floating-point operations per second (FLOPs) and time to compare computational resource consumption of all methods. Since our method includes two phases, we calculate the resource consumption on the two phases separately. Our method consumes 54.069P FLOPs and takes 2394.7 seconds in Phase 1. Although the resource consumption in Phase 1 is relatively high, it is worth noting that Phase 1 only needs to be executed once. In Phase 2, our method consumes 19.738P FLOPs and takes 2669.9 seconds. Compared to Phase 1, the resource consumption in Phase 2 is significantly reduced. Compared to other methods, such as SMER and SVRE, our method consumes fewer resources in general during the attack process.

#### 4.4. Further Analysis

Since we design three strategies for model augmentation, we further analyze the effect of each strategy on the transferability of adversarial examples.

**Whether each strategy contributes to the improvement of transferability?** We first conduct experiments to test the attack performance when using the three strategies separately. From Figure 4(a), it can be observed that all three strategies significantly improve the attack success rate over the Ens setting, demonstrating their effectiveness in augmenting the surrogate models.

**Is each strategy indispensable to the overall attack performance?** We further conduct experiments to test the

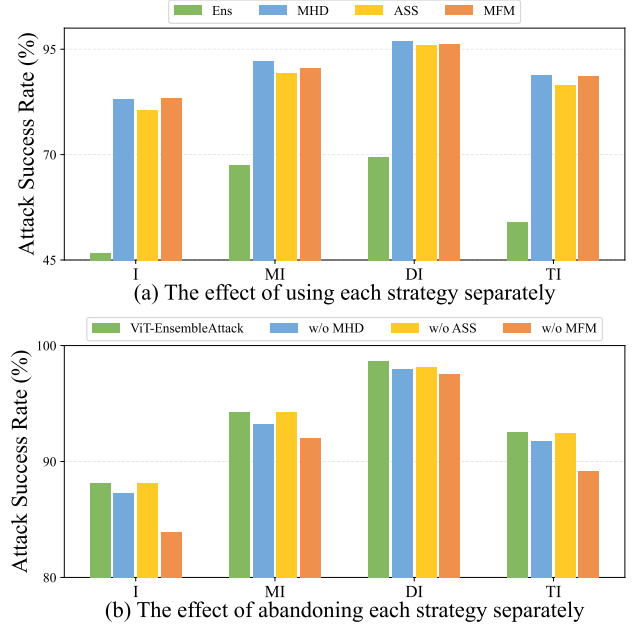


Figure 4. The average attack success rates (%) against ViTs and CNNs with different settings of augment strategies: (a) the effect of using each strategy separately, (b) the effect of abandoning each strategy separately.

effect of abandoning each strategy on the overall attack success rate. It can be seen from Figure 4 (b) that when abandoning one strategy, the attack success rate declines in most cases, demonstrating that each strategy is indispensable in our model augmentation. We also observe an interesting phenomenon: when abandoning MFM, the attack success rate declines the most. We believe this is because MHD and ASS are both designed for the multi-head attention module, restricting the diversity of augmented models. In contrast, when abandoning MHD or ASS, the remaining two strategies are for multi-head attention and multi-layer perception, ensuring diversity and achieving higher performance.

## 5. Conclusion

In this work, we propose ViT-EnsembleAttack, a novel ensemble-based adversarial attack designed for ViTs. Different from prior ensemble-based attacks, we propose to augment surrogate models by increasing diversity to enhance the transferability of adversarial examples. Extensive experimental results show that our method outperforms state-of-the-art methods by a substantial margin across various transfer settings. The core innovation of our method lies in the adversarial augmentation of the surrogate models. Future work could explore new augmentation techniques on ViTs and other kinds of models to enhance the ensemble-based adversarial transferability.



## Acknowledgments

This work is supported by the National Natural Science Foundation (U22B2017) and the International Cooperation Foundation of Hubei Province, China (2024EHA032).

## References

- [1] Bin Chen, Jiali Yin, Shukai Chen, Bohao Chen, and Ximeng Liu. An adaptive model ensemble adversarial attack for boosting adversarial transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4489–4498, 2023. 1, 3, 5
- [2] 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
- [3] 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, 5
- [4] 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, 3, 5
- [5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. 1, 5
- [6] Stéphane d’Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International conference on machine learning*, pages 2286–2296. PMLR, 2021. 5
- [7] Aditya Ganeshan, Vivek BS, and R Venkatesh Babu. Fda: Feature disruptive attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8069–8079, 2019. 2
- [8] Zhijin Ge, Hongying Liu, Xiaosen Wang, Fanhua Shang, and Yuanyuan Liu. Boosting Adversarial Transferability by Achieving Flat Local Maxima. In *Proceedings of the Advances in Neural Information Processing Systems*, 2023. 2
- [9] 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. In *Proceedings of the ACM International Conference on Multimedia*, page 4440–4449, 2023. 1
- [10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2
- [11] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet’s clothing for faster inference. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12259–12269, 2021. 5
- [12] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017. 3
- [13] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in neural information processing systems*, 34:15908–15919, 2021. 5
- [14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 5
- [15] 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
- [16] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013. 1, 3
- [17] 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, 5
- [18] Qizhang Li, Yiwen Guo, Wangmeng Zuo, and Hao Chen. Improving adversarial transferability via intermediate-level perturbation decay. *Advances in Neural Information Processing Systems*, 36, 2024. 1
- [19] Yingwei Li, Song Bai, Yuyin Zhou, Cihang Xie, Zhishuai Zhang, and Alan Yuille. Learning transferable adversarial examples via ghost networks. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11458–11465, 2020. 2, 3
- [20] 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
- [21] Qinliang Lin, Cheng Luo, Zenghao Niu, Xilin He, Weicheng Xie, Yuanbo Hou, Linlin Shen, and Siyang Song. Boosting adversarial transferability across model genus by deformation-constrained warping. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3459–3467, 2024. 1
- [22] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016. 1, 2
- [23] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete representations strengthen vision transformer robustness. *arXiv preprint arXiv:2111.10493*, 2021. 5
- [24] Xiaofeng Mao, Yuefeng Chen, Ranjie Duan, Yao Zhu, Gege Qi, Xiaodan Li, Rong Zhang, Hui Xue, et al. Enhance the visual representation via discrete adversarial training. *Advances in Neural Information Processing Systems*, 35:7520–7533, 2022. 5

- [25] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 12042–12051, 2022. 5
- [26] Sachin Mehta and Mohammad Rastegari. Separable self-attention for mobile vision transformers. *arXiv preprint arXiv:2206.02680*, 2022. 5
- [27] Yichuan Mo, Dongxian Wu, Yifei Wang, Yiwen Guo, and Yisen Wang. When adversarial training meets vision transformers: Recipes from training to architecture. *Advances in Neural Information Processing Systems*, 35:18599–18611, 2022. 5
- [28] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 262–271, 2020. 3
- [29] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 5
- [30] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 5
- [31] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence*, 2017. 5
- [32] Bowen Tang, Zheng Wang, Yi Bin, Qi Dou, Yang Yang, and Heng Tao Shen. Ensemble diversity facilitates adversarial transferability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24377–24386, 2024. 1, 3, 5, 6
- [33] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 5
- [34] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 32–42, 2021. 5
- [35] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017. 3, 5
- [36] Kunyu Wang, Xuanran He, Wenxuan Wang, and Xiaosen Wang. Boosting adversarial transferability by block shuffle and rotation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24336–24346, 2024. 1
- [37] 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
- [38] 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*, 2021. 1
- [39] 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*, pages 4607–4619, 2023. 1
- [40] Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning*, pages 36246–36263. PMLR, 2023. 3
- [41] Zhipeng Wei, Jingjing Chen, Micah Goldblum, Zuxuan Wu, Tom Goldstein, and Yu-Gang Jiang. Towards transferable adversarial attacks on vision transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2668–2676, 2022. 1, 3
- [42] Wang Xiaosen, Kangheng Tong, and Kun He. Rethinking the backward propagation for adversarial transferability. *Advances in Neural Information Processing Systems*, 36:1905–1922, 2023. 3
- [43] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. *arXiv preprint arXiv:1711.01991*, 2017. 3
- [44] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2730–2739, 2019. 2, 5
- [45] 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, 3, 5
- [46] Jianping Zhang, Weibin Wu, Jen-tse Huang, Yizhan Huang, Wenxuan Wang, Yuxin Su, and Michael R Lyu. Improving adversarial transferability via neuron attribution-based attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14993–15002, 2022. 1, 2
- [47] Jianping Zhang, Jen tse Huang, Wenxuan Wang, Yichen Li, Weibin Wu, Xiaosen Wang, Yuxin Su, and Michael R. Lyu. Improving the Transferability of Adversarial Samples by Path-Augmented Method. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8173–8182, 2023. 2
- [48] Zeliang Zhang, Rongyi Zhu, Wei Yao, Xiaosen Wang, and Chenliang Xu. Bag of Tricks to Boost Adversarial Transferability. 2024. 1