Transferable Adversarial Examples against Vision Transformers via Adversarial Block Drop

Huipeng Zhou[1] , **Yu-an Tan**[2] , **Yajie Wang**[2] , **Shangbo Wu**[2] , **Ruinan Ma**[2] , **Quanxin Zhang**[1] and **Yuanzhang Li**[*,1]
[1]School of Computer Science and Technology, Beijing Institute of Technology
[2]School of Cyberspace Science and Technology, Beijing Institute of Technology
{zhouhuipeng, popular}@bit.edu.cn

Abstract

Vision Transformers (ViTs) have shown impressive performance in various vision tasks, which has aroused scholarly interest in studying adversarial example generation and transferability on ViTs. ViT has architecture with self-attention at its core, which is entirely different from traditional convolutional neural networks (CNNs). However, existing adversarial attacks have limited effect on ViTs due to neglecting these architectural features. To address this issue, we propose a self-attention oriented Adversarial Block Drop (ABD) method to generate transferable adversarial examples by skipping attention mechanism from partial blocks. The ViT encoder consists of multiple blocks that are consistent architectures consisting of a self-attentive layer and a feed-forward layer. Specifically, we tailor our approach to this architecture, enhancing self-attention uncertainty by dropping some of the blocks during inference and thus fooling the model decisions. This exploits a unique but widely used architectural feature in the transformer model that can be used as a general attack pattern. Extensive experiments using multiple popular transformers on ImageNet datasets show that the proposed ABD significantly outperforms other baseline methods. Our approach can greatly improve the transferability between ViTs and from ViTs to both CNNs and MLPs, demonstrating the true generalization potential of ViTs in the adversarial space.

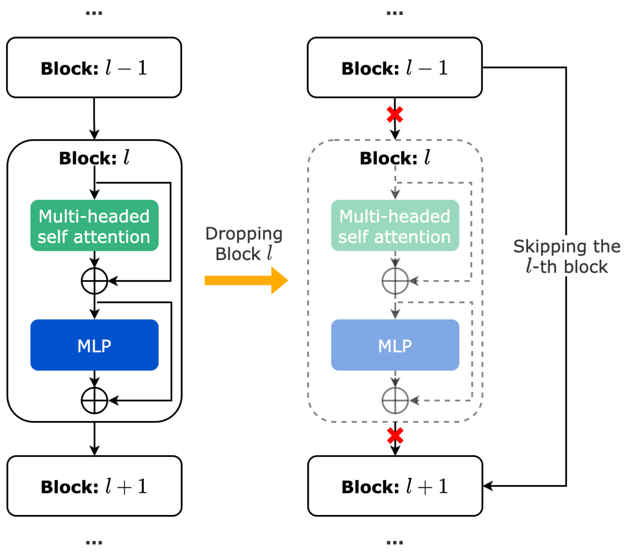

Figure 1: A different network structure is generated by dropping the blocks in the transformer encoder. The l-th block is dropped with probability $\mathcal{P}$ such that the $(l-1)$-th block is connected directly to the $(l+1)$-th block.

1 Introduction

Vision Transformers (ViTs) have not only excelled in natural language processing [Vaswani *et al.*, 2017], but also achieved significant performance gains in a wide range of computer vision tasks [Han *et al.*, 2020b]. This makes ViTs one of the standard architectures in computer vision. However, ViTs remain vulnerable to security threats from adversarial examples. Despite significant architectural differences from traditional convolutional neural networks (CNNs), it can still lead models to incorrect prediction by adding imperceptible perturbations to the input [Naseer *et al.*, 2022]. This has inspired research on adversarial attacks on ViTs [Bhojanapalli *et al.*, 2021; Naseer *et al.*, 2021; Mao *et al.*, 2022], and also presents novel security challenges for security-sensitive deep learning applications, such as face recognition and autonomous driving [Yuan *et al.*, 2019; Deng *et al.*, 2021].

Depending on the amount of information available to the adversary, adversarial attacks can be divided into white-box and black-box attacks [Papernot *et al.*, 2017]. Among them, black-box attacks need to be performed when the target model is entirely unknown. Transfer-based attacks are usually used to generate adversarial examples through local surrogate models and then transfer to unknown black-box target models [Dong *et al.*, 2018]. Transfer attacks are more challenging and of realistic value and have been well studied on CNNs [Xie *et al.*, 2019], but methods designed based on CNNs have been proved to be less transferable to ViTs [Shao *et al.*, 2021]. Some recent works have initially explored transfer-based attacks against ViTs [Aldahdooh *et al.*, 2021;

Mahmood *et al.*, 2021; Wei *et al.*, 2021], but the transferability they achieve still needs to be improved.

ViTs are fundamentally different from CNNs. A ViT consists of multiple encoder blocks, each of which can refine the input series of image patches to generate a self-attentive feature map, thus allowing the model to learn the relationship between any individual patches. The unique structure enables ViTs to generate an interesting set of features, but still exhibits low transferability in a black-box setting [Mahmood *et al.*, 2021; Bhojanapalli *et al.*, 2021; Shao *et al.*, 2021]. These encoder blocks, which play an essential role in the decision-making process of the transformer, are unique architectural features of ViTs, motivating us to exploit them during the attack. In this work, we introduce a highly transferable attack method by deliberately discarding some encoder blocks. Our approach enhances the transferability of adversarial attacks from ViT to unknown models. Our proposed transferable attack undermines the critical concept of ViT models — the self-attention mechanism.

Our approach is inspired by the modular nature of ViTs [Yuan *et al.*, 2021; Touvron *et al.*, 2021a]: the input series of image patches is repeatedly processed by multiple blocks, i.e., multi-headed self-attention [Vaswani *et al.*, 2017], which allows ViTs to better capture the image of key features, playing a crucial role for model prediction. With a thorough analysis of the key multi-headed self-attentive (MSA) mechanism of ViTs, we found that there is a very high correlation between the model's attention to spatial patches and blocks, which leads to overfitting the features of the model's attention to attention. Based on this, we explore whether some of the ViT blocks can be skipped by reusing the previous block output's latent vector to minimize each block's global impact, thus confusing the attention mechanism and image features. Proposed attack, called Adversarial Block Drop (ABD), exploits different blocks during each iteration and enriches the adversarial gradient in the forward pass to improve transferability. This process can simultaneously break the self-attention mechanism and avoid the influence of attention weights on the gradients. In other words, discarding some of the intermediate blocks allows the attention map to be continuously updated during each iteration, which prevents strong correlations between image patches. For a particular block, a diversity of input patterns can effectively improve the generalization of the output. This is a mechanism similar to Dropout [Srivastava *et al.*, 2014] that helps prevent adversarial examples from overfitting to the local model. We show that generating adversarial samples using all ViT blocks is redundant. The predictive features that make the model capture inaccuracies by interfering with some blocks can serve as more generalizable attack patterns, significantly improving the adversarial transferability of ViTs.

To evaluate the effectiveness of our proposed ABD attack, we attacked three families of neural architectures: ViTs, CNNs, and MLP-based models. We conducted extensive experiments on ImageNet datasets in both white-box and black-box settings and compared them with 7 state-of-the-art transfer attacks. The experimental results show that our ABD achieves a higher attack capability and significantly outperforms the baseline.

We highlight our main contributions as follows:

- We tailor the Adversarial Block Drop (ABD) method to enhance the transferability of adversarial examples for ViT, which is applicable to any attention-based network.
- We propose to drop some of the encoder blocks of ViTs to obfuscate the self-attentive features, which can significantly improve the transferability of the adversarial examples and thus achieve a true generalization of the ViT adversarial space.
- We conduct extensive experiments to comprehensively evaluate that ABD achieves superior performance over state-of-the-art transfer attack methods. Moreover, the proposed framework can be combined with existing methods to improve the attack performance further.

2 Related Work

Vision Transformers. Vision Transformer (ViT) is first proposed in [Dosovitskiy *et al.*, 2020]. ViTs takes image patches as input and pretrains them using a huge dataset. To overcome the issue of model pretraining, the massive dataset-based DeiT [Touvron *et al.*, 2021a] introduced a transformer-specific teacher-student strategy that uses a new distillation token to learn knowledge from CNNs. T2T-ViT [Yuan *et al.*, 2021] introduced the T2T module to model the local structure of an image and uses the deep-narrow structure as the backbone of the transformer. Swin [Liu *et al.*, 2021] allows the model to learn information across the window by introducing a sliding window mechanism.

Adversarial Attacks. Transfer-based attacks on CNNs have been well studied and have achieved satisfactory performances. The momentum iterative attack MI-FGSM (MIM) [Dong *et al.*, 2018] is designed to stabilize the update direction by integrating momentum terms and avoiding local optima. The diversity input attack DI-FGSM (DIM) [Xie *et al.*, 2019] randomly resizes and fills the input with a fixed probability at each iteration. The translation invariant attack TI-FGSM (TIM) [Dong *et al.*, 2019] optimizes a perturbation over an ensemble of translated images by convolving the gradient with a linear or Guassian kernel, motivated by the near translation-invariance of CNNs. The scale-invariant attack SI-FGSM (SIM) [Lin *et al.*, 2019] uses scaling invariance to scale the image i.e. the pixel values are multiplied by a factor $\left(\boldsymbol{S}\left(\boldsymbol{x}\right) = \boldsymbol{x}/2^{i}\right)$. The Nesterov iterative attack NI-FGSM (NIM) [Lin *et al.*, 2019] improves gradient momentum using the Nesterov Accelerated Gradient (NAG) method to improve the transferability of adversarial examples. The variance tuning gradient-based attack VMI-FGSM(VMI) stabilizes the update direction by reducing the variance of the gradient at each iteration, thus avoiding local optima in the search process.

Compared to transfer-based attacks on CNNs, less effort has been made to investigate the transferability of adversarial examples between white-box ViTs and different structural black-box models. [Naseer *et al.*, 2022] proposes a self-ensemble (SE) method, which uses class labelling at each layer and a shared classification head to construct a model

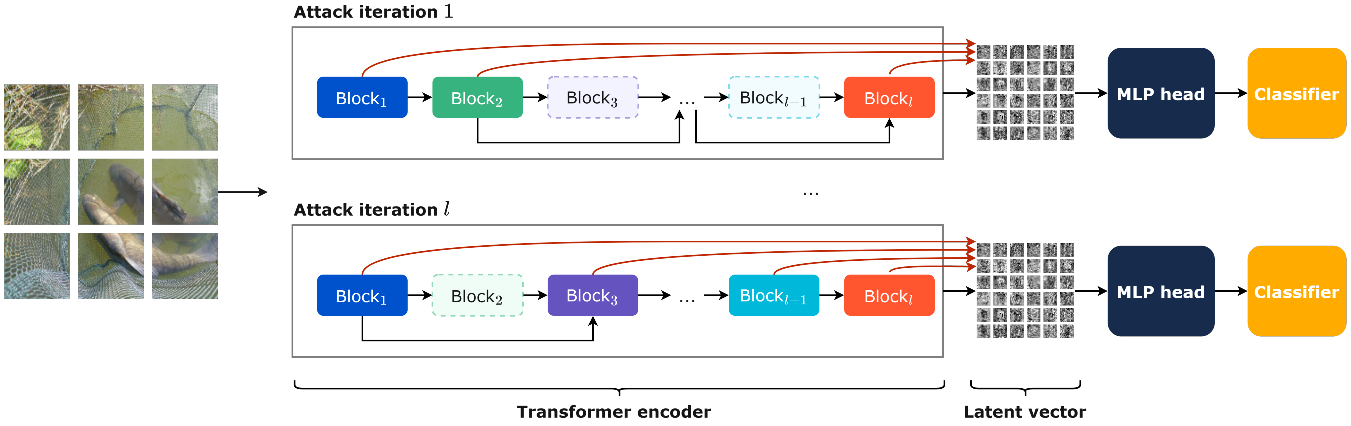

Figure 2: The ABD framework. ABD is applied to blocks in the transformer encoder during attack iterations. Direct block connection with the latent vector (red arrow) is cut-off when blocks are dropped. This allows the surrogate model to obtain a different network structure at each iteration. In this way, the problem of adversarial examples overfitting surrogate models can be effectively avoided, improving transferability.

ensemble. Transferability is improved by optimizing the perturbations on the model ensemble. [Wei *et al.*, 2021] proposes a Pay No Attention (PNA) method in which skipping the gradient of attention during back-propagation can generate adversarial examples with high transferability.

3 Methodology

3.1 Preliminary

Consider a vision transformer model $\mathcal{F}$ for image classification, given a clean input $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^{H\times W\times C}$ and its corresponding ground-truth label $y \in \mathcal{Y} = \{1, \ldots, K\}$, where H, W and C denote the height, width, and number of channels of input $\boldsymbol{x}$ and K denotes the number of classes. We use $\mathcal{F}(\boldsymbol{x}) : \boldsymbol{x} \rightarrow y$ to denote the prediction function of the ViT surrogate model. We use $\mathcal{M}$ to denote the black-box victim model, which can be either ViT or CNN. We focus on non-targeted adversarial attacks. The goal of adversarial attack is to generate an adversarial example $\boldsymbol{x}'$ with the internal information of $\mathcal{F}$. $\boldsymbol{x}'$ can mislead the prediction of the target model $(\arg\max \mathcal{M}(\boldsymbol{x}') \neq y)$. A set of distortion constraints is also imposed on the adversarial example to control the visual differences from the clean sample, usually set under an ℓ_∞-norm bounded constraint, i.e., $\|\boldsymbol{x} - \boldsymbol{x}'\|_\infty < \varepsilon$, where ε is the maximum perturbation budget.

3.2 Adversarial Block Drop

For a ViT classification model $\mathcal{F}$, the network structure consists of a transformer encoder and a classifier. To handle 2D images, the ViT first reshapes the input image $\boldsymbol{x}$ into a sequence of flattened 2D patches $\boldsymbol{x}_p \in \mathbb{R}^{N\times(P^2\cdot C)}$, where $N = H \times W/P^2$ is the resulting number of patches. N also serves as the effective input sequence length for the transformer. Similar to BERT's [class] token, the input image $\boldsymbol{x}$ is transformed into the latent vector $\boldsymbol{z}_0$ of size D as

$$\boldsymbol{z}_0 = \{\boldsymbol{x}_{\text{class}}; \boldsymbol{x}_p^1 E; \boldsymbol{x}_p^2 E; \ldots; \boldsymbol{x}_p^N E\} + E_{\text{pos}}, \\ \text{where } E \in \mathbb{R}^{(P^2\cdot C)\times D}, E_{\text{pos}} \in \mathbb{R}^{(N+1)\times D}. \quad (1)$$

The transformer encoder consists of L blocks. Each block consists of alternating layers of multi-headed self-attention (MSA) and MLP, with LayerNorm (LN) applied before each block. The latent vector $\boldsymbol{z}_l$ output by the l_{th} block is formulated as

$$\boldsymbol{z}_l^{'} = \text{MSA}(\text{LN}(\boldsymbol{z}_{l-1})) + \boldsymbol{z}_{l-1}, \quad l \in [1, L], \quad (2)$$

$$\boldsymbol{z}_l = \text{MLP}(\text{LN}(\boldsymbol{z}_l^{'})) + \boldsymbol{z}_l^{'}, \quad l \in [1, L]. \quad (3)$$

It is clear that the class token $\boldsymbol{z}_l^0$ used for the final prediction comes from a linear mapping of the L blocks, while maintaining the strong correlation of the global patches. Due to the single model structure and the strong global correlation of the self-attention mechanism, it is easy to overfit the local surrogate model during adversarial attacks, reducing the transferability of adversarial examples. To solve these problems, we introduce ABD — Adversarial Block Drop, which randomly attacks a subset of blocks at each iteration to mitigate overfitting. The method is detailed as follows.

The transformer architecture relies entirely on the attention mechanism. Each *block* models global information well, and even shallow *blocks* can learn structural information. In addition, the parallel computation of multiple blocks solves the input-output dependency problem. We take advantage of the behavior where the transformer uses a constant latent vector size D across all its layers, and treat the block in the transformer as a base unit for *drop* operations.

In ABD, each block will be dropped with probability $\mathcal{P}$ when generating adversarial examples, as shown in Figure 1. This not only allows obtaining surrogate models with different structures at each attack iteration, but also changes the linear mapping relation of $\boldsymbol{z}_l^0$, which in turn, breaks the strong correlation of the self-attention mechanism, allowing for higher transferability of adversarial examples created. Thus, the diversity is not only limited to the structure of ViT but also affects the attention mechanism. For the l-th block, which has a probability of being dropped at network forward prediction with $\mathcal{P}$, the latent vector $\boldsymbol{z}_{l+1}$ output from the

$(l+1)$-th block is formulated as

$$\boldsymbol{z}_{l+1} = \begin{cases} \mathrm{MLP}(\mathrm{LN}(\boldsymbol{z}_l^{'})) + \boldsymbol{z}_l^{'}, & \text{sampling } p^l \geq \mathcal{P}, \\ \mathrm{MLP}(\mathrm{LN}(\boldsymbol{z}_{l-1}^{'})) + \boldsymbol{z}_{l-1}^{'}, & \text{sampling } p^l < \mathcal{P}, \end{cases} \tag{4}$$

and the prediction y of the surrogate model is thus formulated as

$$y = \boldsymbol{z}_0 \circ M(\boldsymbol{z}_1) \circ M(\boldsymbol{z}_2) \circ \cdots \circ M(\boldsymbol{z}_l). \tag{5}$$

According to Equation 1, 2, 3, and 5, all blocks contain the MSA structure, causing each block to learn some image features. Finally, the image features learned by each block are mapped cohesively in the latent vector $\boldsymbol{z}_l$ output by the l-th layer block, thus affecting the class token $\boldsymbol{z}_l^0$. used to predict the outputs. We argue that the attention gradient of each block's output impacts the transferability of the adversarial examples, and we discard some blocks to suppress the interconnection between attention gradients and adversarial examples. Our proposed method ignores the attention gradient generated by certain blocks, i.e., leaving certain blocks without contribution to the self-attention mechanism. This makes the final attention region of the adversarial examples generated not fitting the model, and the adversarial perturbation is forced to generalize to regions that are not relevant to the model's attention, achieving a transferability boost.

We showcase our final framework for using ABD in adversarial attacks in Figure 2, and the algorithm in Algorithm 1. In our proposed attack, each iteration generates a different network structure and self-attention mechanism mapping rules. It is worth noting that our proposed method can easily make local models structurally diverse when learning transferable adversarial examples on ViTs without additional overhead, avoiding the problem of over-fitting of surrogate models.

4 Experiments

4.1 Settings

Dataset. We conduct all experiments on the 1000 images from the NeurIPS 2017 Adversarial Learning Challenge [Google, 2017], sampled from ImageNet.

Models. We use 4 state-of-the-art transformers as local white-box surrogate models when crafting adversarial examples: T2T-ViT-24, ViT-B/16, DeiT-B/16, and DeiT-small. We then perform black-box evaluations on a wide variety of models, namely another 5 ViTs: T2T-ViT-19, ViT-L/32, MobileViT [Mehta and Rastegari, 2021], PiT-B [Heo *et al.*, 2021], and CaiT-S-24 [Touvron *et al.*, 2021b], 5 CNNs: Inception-v4 [Szegedy *et al.*, 2016], ResNet50 [He *et al.*, 2015], VGG19 [Simonyan and Zisserman, 2014], RepVGG-A0 [Ding *et al.*, 2021], and ReXNetV1 [Han *et al.*, 2020a], and 2 MLPs: MixerMLP-B/16 [Tolstikhin *et al.*, 2021] and SwinMLP-B.

Attacks. The following adversarial attacks are used as baselines: MIM, TIM, DIM, SIM, NIM, and VMI. The combination of these attacks is also used, namely MITI-DISINIM, abbreviated as CombM. Our proposed method is named **ABD**, and annotated with * where applicable.

Algorithm 1 Adversarial Block Drop

Input: Original image $\boldsymbol{x}$ with ground-truth label y.
Parameter: Denote ViT model $\mathcal{F}$, the number of attack iterations T, step size α, ℓ_∞-norm constraint ε, probability of a block drop $\mathcal{P}$, number of transformer encoder blocks L, $\mathcal{J}$ is the loss function of a network with parameter θ.
Output: Adversarial example $\boldsymbol{x}^{'}$.

```
x' ← x  ($\boldsymbol{x}^{'} \leftarrow \boldsymbol{x}$)
```

```
$\boldsymbol{x}^{'} \leftarrow \boldsymbol{x}$
while $i < T$ do
    $\boldsymbol{z}_0 \leftarrow \boldsymbol{x}^{'}$
    $\boldsymbol{z}^* \leftarrow \boldsymbol{x}^{'}$
    while $l < L$ do
        $\boldsymbol{p}_l \leftarrow \boldsymbol{p}_l \sim \boldsymbol{N}(0,1)$    {for each forward block $\mathcal{B}_l$}
        if $\boldsymbol{p}_l < \mathcal{P}$ then
            $\boldsymbol{z}_l = \boldsymbol{z}^*$
        else
            $\boldsymbol{z}_l = \mathcal{B}_l(\boldsymbol{z}_{l-1})$
            $\boldsymbol{z}^* = \boldsymbol{z}_l$
        end if
    end while
    output $\leftarrow \mathcal{F}(\boldsymbol{x}^{'})$
    $g = \frac{\nabla \mathcal{J}(\text{output}, y)}{\|\nabla \mathcal{J}(\text{output}, y)\|_1}$
    $\boldsymbol{x}^{'} = \text{clip}_{\boldsymbol{x},\varepsilon}\left\{\boldsymbol{x}^{'} + \alpha \cdot \text{sign}(g)\right\}$
end while
return $\boldsymbol{x}^{'}$
```

Hyper-parameters. A maximum of $\varepsilon = 8$ is set under ℓ_∞-bounded constraint. Attacks are set with iteration $T = 10$. All other parameters not mentioned are kept same as their original implementations.

Probability threshold. The choice of probability threshold $\mathcal{P}$ is of paramount importance to our method. In our experiments, $\mathcal{P}$ is set to 0.15, 0.50, 0.25, and 0.20 for ViT-B/16, T2T-ViT-24, DeiT-B/16, and DeiT-small respectively. We further discuss this in Section 4.5.

4.2 Transferability evaluation

Transferring to ViTs. We first explore the effectiveness of our proposed attack against ViTs under black-box scenarios, where we attack local white-box surrogate models and then transfer the generated adversarial examples onto black-box ViTs. Our results are reported in Table 1. By leveraging ViT-specific architectural characteristics, our method drastically improves the transferability of all adversarial attacks not designed for ViTs by 15% on average when used in combination with said attacks.

Our evaluation is based on attacking various white-box surrogate ViTs that share the similar aforementioned self-attention mechanism, but hold different structural designs internally. We argue that our exceptional results stem directly from our approach's consideration of ViT-specific architectures, which is universal across ViTs, regardless of internal differences. By breaking the self-attention mechanism of ViTs, ABD successfully exploits ViT's architectural features, attaining generalized adversarial transferability. Our proposed ABD is universally applicable and can be easily

Surrogate Model	Transfer Model	Attack's fooling rate (%) (* are ours)													
		MIM		TIM		DIM		SIM		NIM		VMI		CombM	
T2T-ViT-24	T2T-ViT-19	71.80	**91.20***	70.70	**86.20***	86.90	**93.70***	74.50	**94.40***	80.80	**90.60***	94.50	**99.00***	97.00	**98.90***
	ViT-L/32	15.50	**17.10***	13.60	**16.80***	19.50	**22.40***	15.50	**18.60***	15.50	**16.20***	29.10	**30.00***	36.20	**39.60***
	MobileViT	34.80	**59.40***	30.90	**54.90***	61.50	**74.30***	37.00	**69.90***	37.30	**57.40***	57.20	**81.90***	76.20	**94.80***
	PiT-B	19.70	**27.90***	18.30	**24.80***	43.50	**53.50***	22.60	**33.80***	22.50	**26.00***	49.80	**55.10***	67.00	**69.00***
	CaiT-S24	14.60	**23.30***	12.20	**20.20***	36.10	**49.00***	17.90	**28.60***	17.10	**22.20***	46.80	**51.80***	62.50	**66.30***
ViT-B/16	T2T-ViT-19	11.60	**18.30***	9.10	**14.30***	20.50	**22.00***	15.30	**22.50***	13.40	**19.30***	23.50	**37.70***	35.50	**44.50***
	ViT-L/32	47.60	**79.10***	39.90	**70.40***	72.60	**74.80***	63.10	**88.70***	51.20	**77.90***	75.40	**94.00***	91.20	**96.70***
	MobileViT	29.60	**35.40***	24.90	**30.60***	35.50	**36.90***	33.40	**41.50***	30.10	**36.80***	36.90	**46.90***	51.20	**59.80***
	PiT-B	12.20	**16.10***	9.10	**12.30***	17.70	**19.50***	12.40	**16.70***	13.20	**14.40***	19.00	**27.90***	27.80	**33.30***
	CaiT-S24	14.70	**28.20***	10.10	**19.30***	25.10	**29.10***	19.50	**35.00***	17.50	**28.40***	34.50	**56.40***	42.20	**53.90***
DeiT-B/16	T2T-ViT-19	24.00	**55.30***	16.80	**42.50***	48.30	**50.90***	28.30	**66.80***	26.50	**54.50***	44.20	**78.30***	67.50	**84.30***
	ViT-L/32	27.10	**58.20***	22.80	**50.30***	42.70	**48.00***	35.50	**73.70***	29.50	**58.30***	45.70	**77.80***	59.50	**92.30***
	Mobile-ViT	36.10	**61.60***	29.00	**51.30***	49.10	**56.30***	40.90	**69.70***	39.10	**60.20***	48.70	**74.40***	65.50	**85.00***
	Pit-B	20.80	**45.70***	16.70	**37.20***	48.10	**48.80***	23.80	**54.60***	22.80	**44.20***	38.00	**68.60***	68.90	**81.50***
	CaiT-S24	47.20	**90.50***	36.00	**81.90***	73.20	**76.90***	53.30	**94.80***	52.90	**90.10***	73.80	**95.70***	85.20	**95.70***
DeiT-small	T2T-ViT-19	31.60	**54.30***	30.50	**48.90***	67.60	**75.50***	38.90	**63.90***	34.70	**52.00***	60.20	**77.40***	88.00	**90.20***
	ViT-L/32	36.20	**56.40***	33.70	**53.90***	58.90	**63.00***	45.30	**70.50***	37.90	**56.50***	59.30	**75.90***	80.70	**88.40***
	MobileViT	45.00	**64.80***	41.90	**62.60***	73.70	**75.40***	50.90	**74.10***	46.70	**65.90***	58.00	**76.20***	87.30	**93.20***
	Pit-B	22.10	**37.70***	21.60	**34.30***	52.80	**60.30***	26.50	**43.70***	23.80	**33.90***	44.00	**59.50***	75.90	**78.20***
	CaiT-S24	62.40	**88.10***	62.40	**87.20***	86.60	**91.10***	69.30	**93.30***	68.10	**87.50***	89.70	**94.90***	96.20	**97.40***

Table 1: Transferring to ViTs. ABD consistently improves transferability of adversarial examples created on ViTs by 15% on average.

Surrogate Model	Transfer Model	Attack's fooling rate (%) (* are ours)													
		MIM		TIM		DIM		SIM		NIM		VMI		CombM	
T2T-ViT-24	Inception-v4	12.40	**19.70***	9.30	**15.50***	32.20	**36.70***	12.80	**24.10***	14.40	**19.60***	30.90	**41.90***	45.90	**62.60***
	ResNet50	14.70	**23.20***	11.60	**19.40***	30.30	**34.10***	16.20	**25.50***	17.10	**22.40***	35.10	**47.80***	44.40	**60.90***
	VGG19	32.60	**51.70***	24.00	**43.80***	49.20	**60.00***	35.90	**58.00***	34.70	**51.00***	53.00	**75.00***	63.50	**85.90***
	RepVGG-A0	31.50	**49.90***	25.70	**43.60***	54.70	**62.70***	32.60	**55.40***	32.50	**47.90***	53.20	**74.70***	71.90	**90.40***
	ReXNetV1	31.30	**61.60***	25.40	**51.90***	62.10	**70.10***	36.00	**66.40***	36.00	**57.10***	61.90	**83.80***	78.70	**94.70***
	MixerMLP-B/16	26.20	**37.80***	23.50	**34.40***	41.50	**43.40***	26.10	**40.20***	26.20	**35.40***	46.70	**57.60***	51.80	**65.60***
	SwinMLP-B	24.40	**41.20***	20.20	**36.10***	49.40	**54.00***	27.50	**47.10***	28.10	**39.80***	53.00	**66.40***	64.90	**74.50***
ViT-B/16	Inception-v4	7.80	**11.90***	4.60	**7.10***	10.10	**14.80***	10.30	**14.30***	10.90	**11.60***	14.60	**21.60***	21.20	**26.00***
	ResNet50	11.70	**14.20***	8.40	**9.70***	13.10	**15.60***	13.00	**16.40***	12.40	**15.10***	16.00	**22.50***	21.10	**25.00***
	VGG19	26.00	**31.30***	17.90	**24.10***	25.60	**28.10***	28.80	**36.50***	28.50	**33.20***	34.80	**44.30***	42.10	**50.50***
	RepVGG-A0	24.00	**31.70***	17.70	**24.80***	28.70	**33.00***	28.40	**36.70***	26.40	**32.80***	32.60	**42.50***	45.30	**52.90***
	ReXNetV1	18.80	**28.40***	12.90	**19.10***	24.60	**27.00***	23.00	**32.20***	21.50	**29.70***	27.50	**42.60***	43.70	**53.80***
	MixerMLP-B/16	26.20	**38.90***	22.80	**32.10***	34.70	**38.20***	30.50	**44.50***	28.40	**37.00***	37.00	**56.90***	45.70	**56.00***
	SwinMLP-B	11.10	**15.90***	8.30	**11.60***	15.20	**15.50***	12.50	**18.10***	12.40	**15.30***	18.70	**28.50***	25.40	**30.20***
DeiT-B/16	Inception-v4	11.50	**24.10***	6.60	**16.70***	20.10	**23.00***	13.20	**30.70***	13.80	**23.60***	20.90	**38.90***	35.40	**50.90***
	ResNet50	15.40	**28.60***	11.20	**20.00***	22.50	**26.70***	16.60	**35.30***	16.60	**27.80***	24.70	**43.80***	36.30	**52.90***
	VGG19	32.10	**53.20***	22.60	**41.00***	37.10	**45.20***	36.40	**60.20***	34.80	**53.00***	44.70	**68.40***	56.20	**75.70***
	RepVGG-A0	30.30	**54.40***	23.10	**43.80***	40.60	**50.30***	34.90	**62.70***	34.90	**51.80***	43.70	**68.70***	59.50	**81.10***
	ReXNetV1	28.50	**59.60***	19.00	**46.40***	45.00	**52.70***	31.50	**70.60***	33.70	**59.10***	45.20	**76.80***	63.70	**86.10***
	MixerMLP-B/16	39.90	**73.90***	31.00	**64.50***	53.20	**63.00***	44.50	**82.90***	41.90	**73.10***	59.00	**88.10***	67.70	**86.20***
	SwinMLP-B	22.30	**48.40***	16.80	**38.90***	41.50	**48.10***	23.60	**57.70***	22.80	**47.90***	38.30	**68.70***	58.30	**74.60***
DeiT-small	Inception-v4	14.60	**25.50***	11.70	**21.20***	34.60	**36.80***	20.00	**34.30***	17.50	**23.20***	27.50	**40.90***	55.60	**62.80***
	ResNet50	20.60	**29.80***	15.90	**23.80***	36.60	**38.40***	23.00	**36.90***	20.10	**27.10***	30.30	**44.20***	57.40	**64.50***
	VGG19	42.00	**57.40***	34.70	**51.70***	60.90	**63.10***	46.00	**64.10***	43.50	**55.60***	54.10	**68.60***	78.40	**84.70***
	RepVGG-A0	39.10	**57.50***	34.10	**53.20***	65.90	**67.60***	45.10	**67.40***	40.50	**56.90***	52.60	**71.90***	83.50	**90.50***
	ReXNetV1	39.80	**65.20***	33.70	**56.50***	74.20	**74.90***	46.30	**73.50***	42.30	**63.60***	57.70	**79.10***	87.70	**94.50***
	MixerMLP-B/16	49.60	**73.90***	47.80	**70.60***	71.90	**74.30***	55.00	**80.70***	52.00	**72.90***	72.80	**87.50***	81.90	**89.50***
	SwinMLP-B	25.40	**44.60***	24.80	**40.20***	53.20	**58.40***	30.20	**51.20***	27.80	**42.20***	46.40	**63.10***	71.00	**76.70***

Table 2: Transferring to non-ViT models. ABD even considerably improves cross-architectural adversarial transferability by as high as 20%.

implemented on all ViTs that utilize self-attention, achieving state-of-the-art performances.

Transferring to non-ViTs. Next, we investigate our approach's transferability to CNNs and MLPs to prove ABD's universal effectiveness against even non-ViTs. We demonstrate our results in Table 2, where all attacks are launched under black-box scenarios. Original attacks fail miserably when trying to transfer adversarial examples created on ViTs to both CNNs and MLPs, simply because those attacks do not take ViT's unique architecture and self-attention mechanism into account. When combined with ABD, the transferability of ViT-crafted adversarial examples towards non-ViT models is dramatically improved by as high as 20%.

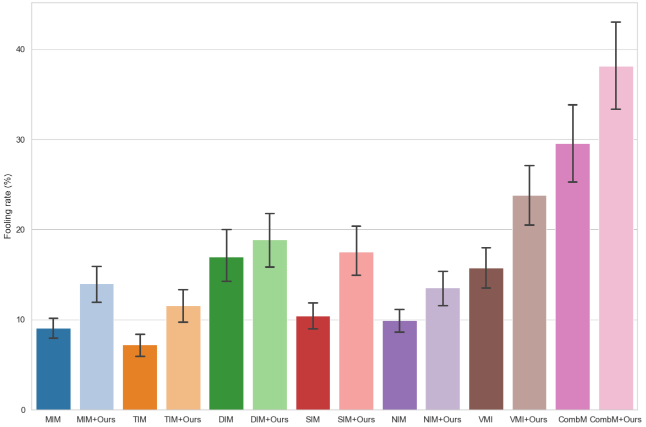

Figure 3: Defense evaluation against seven adversarially trained models. Our proposed approach successfully improves every attack's transferability, achieving as high as almost 40% transfer fooling rate on models with defenses (rightmost bar).

Being able to generalize across various model architectures is what especially makes ABD superior, which is only achievable as ABD successfully exploits the structural traits of ViTs. By dropping blocks of the transformer at a designated probability, our approach is able to structurally diversify local surrogate ViTs to a great extent, breaking structure or even architecture-wise connections between model gradients and adversarial perturbations, thereby avoiding overfitting to a specific model architecture, boosting cross-architectural transferability. Succinctly, we prove our proposed approach's comprehensive superiority and universality even against models with different architectures.

4.3 Evading defenses

We further evaluate ABD when attacking models with defenses. Considering the following adversarial training methods, namely [Tramèr *et al.*, 2017; Kurakin *et al.*, 2016; Kurakin *et al.*, 2018], we use the following models that are adversarially pretrained: Inc-v3, Inc-v4, IncRes-v2, Res-v2, Inc-v3$_{\text{ens3}}$, Inc-v3$_{\text{ens4}}$, IncRes-v2$_{\text{ens}}$. Our results are aggregated (averaged) over the permutation of adversarial examples crafted on all 4 local white-box surrogate ViTs and transferred onto all 7 black-box models with defenses, shown in Figure 3.

Among all baseline attacks, we observe a consistent improvement across every scenario when they are used in combination with our proposed approach. The most powerful attack (rightmost bar) – an ensemble of the selected baseline adversarial attacks, denoted as CombM – when integrated with ABD, managed to achieve an average of nearly 40% transfer fooling rate on adversarially defended models, a most significant improvement. Our results suggest potential future research on adversarial vulnerabilities that can generalize from one model architecture to another, regardless of adversarial defense.

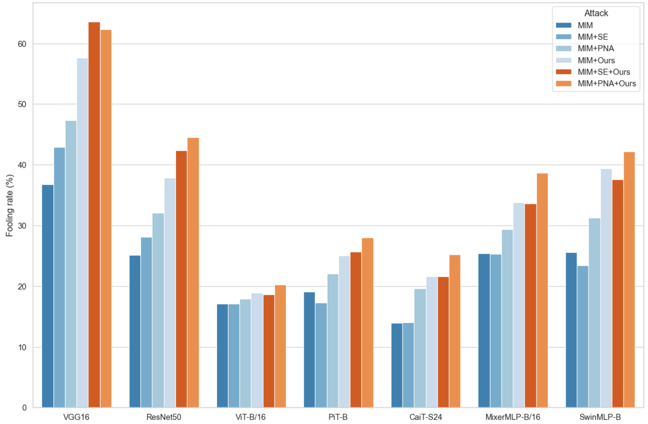

Figure 4: Comparing ABD with SE and PNA on various model architectures. Transferability of attacks that integrate our method is higher than others. ABD even improves SE and PNA themselves when used in combination (last two bars of each group in orange).

4.4 Comparison with attacks that target ViTs

The advent of ViTs brought up a flurry of research on adversarial attacks that specifically target ViTs, yet the number of methods that actually achieve state-of-the-art performance is very limited as we stated in Section 2. Nevertheless, we compare ABD with state-of-the-art attacks that are designed specifically for ViT, namely SE and PNA. These attacks are also designed to be used in combination with baseline attacks that we mentioned earlier. In our experiments, baseline attack MIM is used as the foundation for these methods (including our ABD).

We attack 2 CNNs, 3 ViTs, and 2 MLPs with these methods respectively. Results are shown in Figure 4. We find that ABD always manages to create more transferable adversarial examples than SE or PNA across every kind of model architecture (fourth bar in each bar group, light blue). Grad-CAM visualizations of the adversarial examples in Figure 5 also indicates that our proposed ABD successfully diverts ViT's attention, where SE and PNA both fail. Additionally, our proposed ABD is even able to be adapted to and used in combination with these attacks, further enhancing the cross-architectural transferability of ViT-crafted adversarial examples. We believe that ABD has proven itself an integral part within adversarial attacks on ViTs.

4.5 Discussion of the effect of $\mathcal{P}$

Finally, we would like to discuss the impact of probability threshold $\mathcal{P}$ towards both local ViT performance and fooling rates. The choice of probability threshold $\mathcal{P}$ is undoubtedly a crucial factor in ABD, which intuitively and directly affects the diversity of local surrogate ViT structures. We perform a grid search of $\mathcal{P}$ within range $[0, 1]$ with a step size of 0.05, and plot the correlation between — (1) the top-1 model classification accuracy, (2) the fooling rate of adversarial examples, and (3) the diversity of model structure created by ABD — and the magnitude of probability $\mathcal{P}$ respectively in Figure 6 (from top row to bottom row).

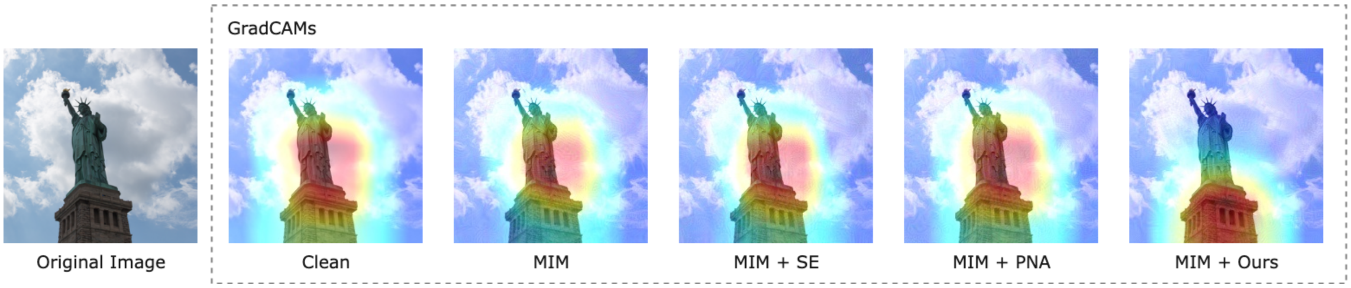

Figure 5: GradCAM visualizations of the model's attention on adversarial examples generated by compared methods (baseline attack MIM, comparing ABD with SE and PNA, transferring from T2T-ViT-24 to black-box ResNet50). Without exploiting ViT-specific architectural characteristics, the model's attention is not diverted, resulting in a failed transfer attack.

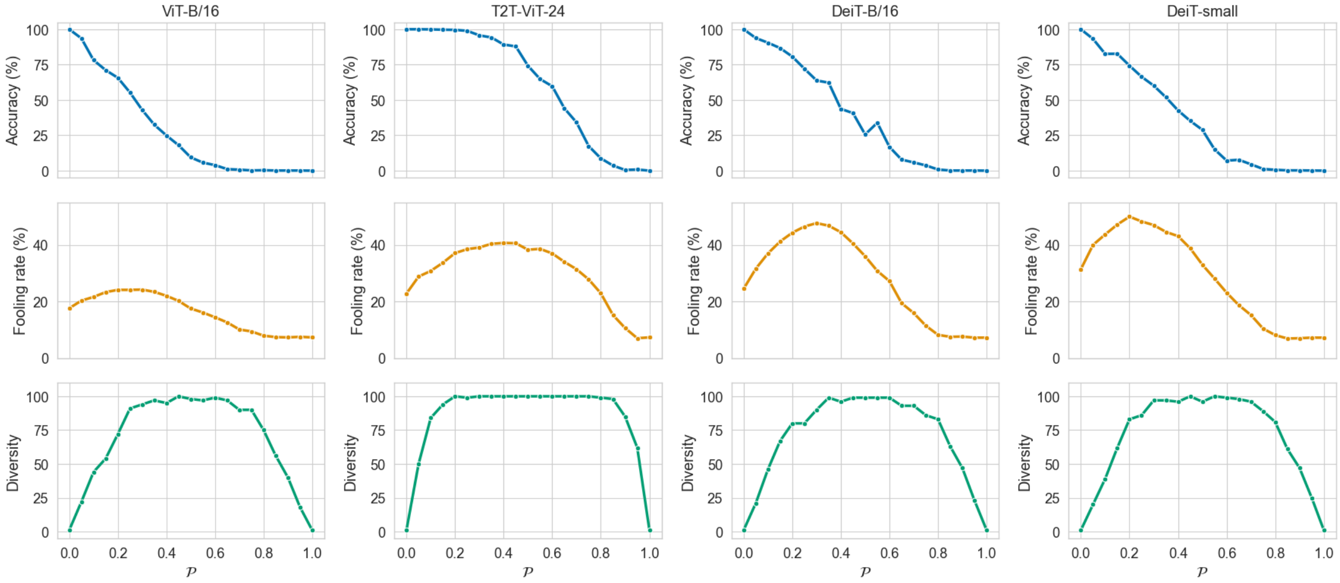

Figure 6: Relationship of $\mathcal{P}$ vs. (1) model accuracy (top row), (2) transfer fooling rate (middle row), and (3) model structure diversity (bottom row) across all four surrogate ViTs. $\mathcal{P}$ is grid searched within range $[0, 1]$ with a step size of 0.05.

Diversifying local models. It is as expected that the classification accuracy of different ViTs is almost always negatively correlated to probability threshold $\mathcal{P}$ (first row in Figure 6). As $\mathcal{P}$ increases, the number of dropped blocks in the transformer also increases, reducing the stability and accuracy of the ViT. However, this is the very act that enriches the diversity of surrogate models (bottom row), making ABD crafted adversarial examples highly transferable and ABD itself state-of-the-art.

Fooling rate. As $\mathcal{P}$ increases from 0 to 1, the fooling rates tend to first increase then decrease, as illustrated in Figure 6's middle row. We usually find $\mathcal{P}$ to be well-balanced when model accuracy decreases by a 30% threshold, where the diversity of local surrogate models is to a maximum and fooling rate to be highest.

5 Conclusion

In this paper, we explore the unique architecture of ViTs and demonstrate the potential of powerful transfer attacks that exploit these architectural features. Our proposed approach involving dropping partial blocks is novel and flexible, achieving significant performance gains. Specifically, we find that ignoring the attention gradient generated by partial blocks at each iteration prevents overfitting from interfering with the self-attention mechanism, thereby improving transferability. We propose Adversarial Block Drop (ABD), tailored for ViTs, that successfully transfers adversarial examples between target models of different architectures. We conducted evaluation on 3 architectures of target models (including 5 ViTs, 5 CNNs, and 2 MLPs). Results show that the ABD achieves significantly better transferability than other baseline approaches. Combining our attack with existing approaches can further enhance transferability. The results of our study show that transfer attacks between architectures with huge disparities still pose a significant threat to model robustness. The implicit bias between ViT models of different depths and even ViT and CNN architectures is not huge. Our work will inspire future work in designing more robust neural network architectures and stronger certifiable defenses.

References

[Aldahdooh *et al.*, 2021] Ahmed Aldahdooh, Wassim Hamidouche, and Olivier Déforges. Reveal of vision transformers robustness against adversarial attacks. *CoRR*, abs/2106.03734, 2021.

[Bhojanapalli *et al.*, 2021] Srinadh Bhojanapalli, Ayan Chakrabarti, Daniel Glasner, Daliang Li, Thomas Unterthiner, and Andreas Veit. Understanding robustness of transformers for image classification. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 10211–10221. IEEE, 2021.

[Deng *et al.*, 2021] Yao Deng, Tiehua Zhang, Guannan Lou, Xi Zheng, Jiong Jin, and Qing-Long Han. Deep learning-based autonomous driving systems: A survey of attacks and defenses. *IEEE Trans. Ind. Informatics*, 17(12):7897–7912, 2021.

[Ding *et al.*, 2021] Xiaohan Ding, X. Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13728–13737, 2021.

[Dong *et al.*, 2018] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 9185–9193. Computer Vision Foundation / IEEE Computer Society, 2018.

[Dong *et al.*, 2019] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4307–4316, 2019.

[Dosovitskiy *et al.*, 2020] 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. *ArXiv*, abs/2010.11929, 2020.

[Google, 2017] Brain Google. Nips 2017: Adversarial learning development set. https://www.kaggle.com/datasets/google-brain/nips-2017-adversarial-learning-development-set, Jul 2017. Accessed: 2022-07-15.

[Han *et al.*, 2020a] Dongyoon Han, Sangdoo Yun, Byeongho Heo, and Young Joon Yoo. Rethinking channel dimensions for efficient model design. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 732–741, 2020.

[Han *et al.*, 2020b] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, Zhaohui Yang, Yiman Zhang, and Dacheng Tao. A survey on visual transformer. *CoRR*, abs/2012.12556, 2020.

[He *et al.*, 2015] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015.

[Heo *et al.*, 2021] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11916–11925, 2021.

[Kurakin *et al.*, 2016] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *ArXiv*, abs/1611.01236, 2016.

[Kurakin *et al.*, 2018] Alexey Kurakin, Ian J. Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, Alan Loddon Yuille, Sangxia Huang, Yao Zhao, Yuzhe Zhao, Zhonglin Han, Junjiajia Long, Yerkebulan Berdibekov, Takuya Akiba, Seiya Tokui, and Motoki Abe. Adversarial attacks and defences competition. *ArXiv*, abs/1804.00097, 2018.

[Lin *et al.*, 2019] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *arXiv: Learning*, 2019.

[Liu *et al.*, 2021] 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. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021.

[Mahmood *et al.*, 2021] Kaleel Mahmood, Rigel Mahmood, and Marten van Dijk. On the robustness of vision transformers to adversarial examples. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 7818–7827. IEEE, 2021.

[Mao *et al.*, 2022] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 12032–12041. IEEE, 2022.

[Mehta and Rastegari, 2021] Sachin Mehta and Mohammad Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. *ArXiv*, abs/2110.02178, 2021.

[Naseer *et al.*, 2021] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 23296–23308, 2021.

[Naseer *et al.*, 2022] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Fahad Shahbaz Khan, and Fatih Porikli. On improving adversarial transferability of vision transformers. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[Papernot *et al.*, 2017] Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In Ramesh Karri, Ozgur Sinanoglu, Ahmad-Reza Sadeghi, and Xun Yi, editors, *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2017, Abu Dhabi, United Arab Emirates, April 2-6, 2017*, pages 506–519. ACM, 2017.

[Shao *et al.*, 2021] Rulin Shao, Zhouxing Shi, Jinfeng Yi, Pin-Yu Chen, and Cho-Jui Hsieh. On the adversarial robustness of visual transformers. *CoRR*, abs/2103.15670, 2021.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014.

[Szegedy *et al.*, 2016] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *ArXiv*, abs/1602.07261, 2016.

[Tolstikhin *et al.*, 2021] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *Neural Information Processing Systems*, 2021.

[Touvron *et al.*, 2021a] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 2021.

[Touvron *et al.*, 2021b] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Herv'e J'egou. Going deeper with image transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 32–42, 2021.

[Tramèr *et al.*, 2017] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Dan Boneh, and Patrick Mcdaniel. Ensemble adversarial training: Attacks and defenses. *ArXiv*, abs/1705.07204, 2017.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[Wei *et al.*, 2021] Zhipeng Wei, Jingjing Chen, Micah Goldblum, Zuxuan Wu, Tom Goldstein, and Yu-Gang Jiang. Towards transferable adversarial attacks on vision transformers. *ArXiv*, abs/2109.04176, 2021.

[Xie *et al.*, 2019] 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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2730–2739. Computer Vision Foundation / IEEE, 2019.

[Yuan *et al.*, 2019] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE Trans. Neural Networks Learn. Syst.*, 30(9):2805–2824, 2019.

[Yuan *et al.*, 2021] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis E. H. Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 538–547. IEEE, 2021.