Focus on Generalization: Improving Adversarial Transferability via Bi-Level Bias Mitigation

Yiqiang Guo
Fuzhou University
Fuzhou, China
231027026@fzu.edu.cn

Lei Zhong
Fuzhou University
Fuzhou, China
231027220@fzu.edu.cn

Bin Chen
Fuzhou University
Fuzhou, China
c_chenbin@foxmail.com

Jia-Li Yin*
Fuzhou University
Fuzhou, China
jlyin@fzu.edu.cn

Xiaolei Liu
China Academy of Engineering Physics
Mianyang, China
luxaole@gmail.com

Shouling Ji
Zhejiang University
Hangzhou, China
sji@zju.edu.cn

Abstract

Transfer-based adversarial attacks have endowed adversarial examples with the ability to transfer from a source model to an unknown target model, which poses a more realistic threat to security-critical applications. Existing transferable adversarial attacks generally suffer from overfitting to the source model, *i.e.*, the perturbations are locally optimal in the source model and focus on the model-specific information. We demand the adversarial perturbation to contain more generalized knowledge, which reveals the intrinsic general properties and can introduce model-general optimum into adversarial examples, for improving transferability. To this end, we devise a Bi-level Bias Mitigated Attack (BBMA), which empowers the transferability of adversarial examples by exploring generalization in two levels: 1) Progressive filtering of high-frequency sample components. We first propose to remove the sample-specific high-frequency components of samples to explore model-level generation. To simulate how a model evaluates feature importance at different stages, we devise a stride-wise step-tuning strategy to progressively produce multiple samples for aggregating the gradients. 2) Accumulated gradient-guided model attention shift. To facilitate the sample-level bias mitigation, we employ an accumulated gradient-guided attention map to distort the more generalized features during perturbation generation. Comprehensive experiments on several benchmarks demonstrate the superiority of our method in attack transferability over state-of-the-art attacks.

CCS Concepts

• **Computing methodologies** → **Feature selection**; *Learning latent representations*; • **Security and privacy** → **Domain-specific security and privacy architectures**.

*Corresponding author.

MM '25, Dublin, Ireland

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755611

Keywords

Adversarial Attack, Adversarial Example, Model Generalization, Frequency Filtering, Attention Shift

ACM Reference Format:
Yiqiang Guo, Lei Zhong, Bin Chen, Jia-Li Yin, Xiaolei Liu, and Shouling Ji. 2025. Focus on Generalization: Improving Adversarial Transferability via Bi-Level Bias Mitigation. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3746027.3755611

1 Introduction

As the Deep Neural Networks (DNNs) have become the de facto tools for various practical applications, their vulnerability to adversarial attacks, *i.e.*, adversarial examples (AEs) that involve imperceptible malicious perturbations, has attracted wide attention [8, 13, 26, 37]. An important property of AEs is their transferability, which empowers the ability to disturb other unknown models in the black-box settings, posing a more realistic threat to security-critical applications.

Recently, various methods have been proposed to improve the transferability of AEs [28, 33, 46, 48]. However, the performance generally falls inferior compared with white-box attacks. The main bottleneck in transfer-based attacks is the overfitting of AEs to the source model, *i.e.*, the adversarial perturbations generated from the source model are model-specific optimal and would not disturb the target model since different models behave differently on the same sample, as shown in Fig. 1 (a). Existing methods single-mindedly focus on mitigating the model-specific knowledge to reduce the reasoning gap between the source model and the target model, yet they fail to explicitly identify and address the structural biases rooted in frequency and attention mechanisms. In contrast, we decompose the transferability challenge into two critical biases: High-Frequency Overfitting Bias(model-level bias) and Focus Discrepancy Bias(sample-level bias). High-Frequency Overfitting Bias, concretely, neural networks tend to first fit the low frequency and as the training process goes deeper, they might capture high-frequency, sample-specific features that represent highly variable patterns in the input data [19, 29]. Intuitively, adversarial examples that primarily manipulate high-frequency, sample-specific features are unlikely to transfer well among different models due to the limited generalization. This overlooked issue motivates us to work

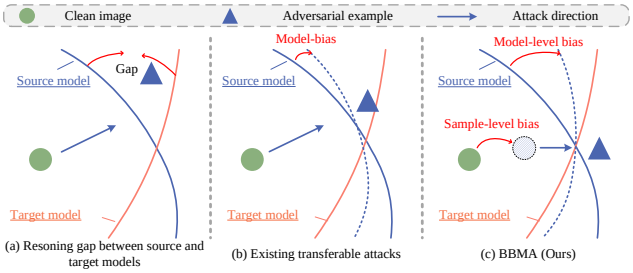

Figure 1: Comparison between existing transferable attacks and the proposed BBMA method. Existing methods single-mindedly focus on escaping the model-local optimum. Our method provides bi-level bias mitigation and can better improve the generalization of adversarial examples.

towards perturbations that contain more generalized knowledge and introduce model-general optimum into AEs. Focus Discrepancy Bias refers to the phenomenon where different neural network models exhibit significant differences in the regions or features of the input data they focus on when processing the same sample. In this paper, we focus on the generalization of adversarial examples from both model-level and sample-level, and propose a new transfer-based adversarial attack dubbed Bi-level Bias Mitigated Attack (BBMA), as shown in Fig. 1 (c). BBMA enriches the generalized information in adversarial perturbations in two levels: **1) Progressive filtering of high-frequency sample components (PFF).** BBMA first explores the model-level generalization by removing the high-frequency components of samples. We propose a stride-wise step-tuning strategy to perform progressive frequency removal where a frequency mask with step-stride is applied to produce multiple samples for aggregating the final attention map with proper frequency remaining. The process of gradually removing features can be viewed as an effective way to simulate how a model evaluates feature importance at different stages. By progressively removing the high-frequency components of an image, we are essentially simulating how the model gradually "ignores" or "focuses" on different details and features. This approach helps to uncover which features are more critical for the model's decision-making process, and which features can be downplayed or disregarded. It allows us to assess the model's ability to prioritize key information, providing deeper insights into its feature importance evaluation. In addition, to facilitate the low-frequency component strengthening, we introduce a sample jitter module to randomly transform inputs via low-level adjustments for improving sample diversity. This progressive high-frequency components removal contributes to the general promotion of samples, and spawns BBMA to excel in reducing model-specific bias and encouraging more general information extraction in feature processing. **2) Accumulated gradient-guided model-specific attention shift (AAS).** Alongside the model-level generalization promotion, we also consider improving the generalization from sample-level bias mitigation. Specifically, we devise an accumulated gradient-guided attention shift method to guide the distortion on more model-agnostic and class-relevant features. Instead of aggregating gradients from multiple transformed images in previous methods [15, 41], we propose to utilize the adversarial perturbations to gradually shift the model attention, and the gradients are accumulated to better evaluate the feature importance.

Our contributions can be summarized as follows:

- We first reveal that the single-minded focus of transfer-based attacks on model bias mitigation overlooks the structural biases rooted in frequency and attention mechanisms, which we categorize as High-Frequency Overfitting Bias (model-level) and Focus Discrepancy Bias (sample-level).
- We propose a generalized transfer-based adversarial attack called Bi-level Bias Mitigated Attack (BBMA). BBMA introduces a progressive frequency filtering module to remove the high-frequency components of samples, and an accumulated gradient-guided attention shift module to mitigate the model-specific feature distortion.
- We demonstrate the effectiveness of our method based on extensive experiments, and reveal that BBMA can effectively mitigate both model-level and sample-level bias.

2 Related works

2.1 Adversarial Attacks

Since Szegedy et al. [37] first demonstrated the existence of adversarial examples, numerous attacks have been proposed to explore the vulnerabilities of neural networks. These attacks are typically classified into two categories: white-box and black-box attacks.

White-Box Attacks. White-box attacks utilize the gradient information of the target model to craft adversarial examples. For instance, the Fast Gradient Sign Method (FGSM) [13] adds perturbations in the direction of the gradient to the benign sample. Iterative Fast Gradient Sign Method (I-FGSM) [23] extends FGSM into an iterative version. Projected Gradient Descent (PGD) [30] extends I-FGSM with a random start. Carlini and Wagner Attack (C&W) [2] designs a loss function that balances between achieving adversarial effectiveness and keeping the perturbation small. However, white-box attacks require full access to the target model, *i.e.*, model structures and parameters, which is often unrealistic in real-world applications where models are protected or black-boxed.

Black-Box Attacks. Black-box attacks generate adversarial examples using a source model and exploit the transferability of these examples to successfully deceive victim models. Some works improve the transferability of AEs with advanced gradient optimization methods. For instance, Momentum Iterative Fast Gradient Sign Method (MI) [8] introduces momentum into I-FGSM to stabilize the optimization direction and help the attack escape local maxima. Similarly, the Nesterov Iterative Fast Gradient Sign Method (NI) [26] incorporates Nesterov Accelerated Gradient to accumulate momentum, resulting in improved transferability. Other methods focus on input transformation techniques to enhance transferability. For example, Spectrum Simulation Attack (SSA) [28] adds Gaussian noise and randomly masks the image in the frequency domain to transform the input image. Likewise, Admix [40] calculates the gradient on the input image admixed with a small portion of each add-in image from other categories while using the original label of the input. In contrast to the aforementioned methods that perturb the output layer, some studies focus on disrupting internal

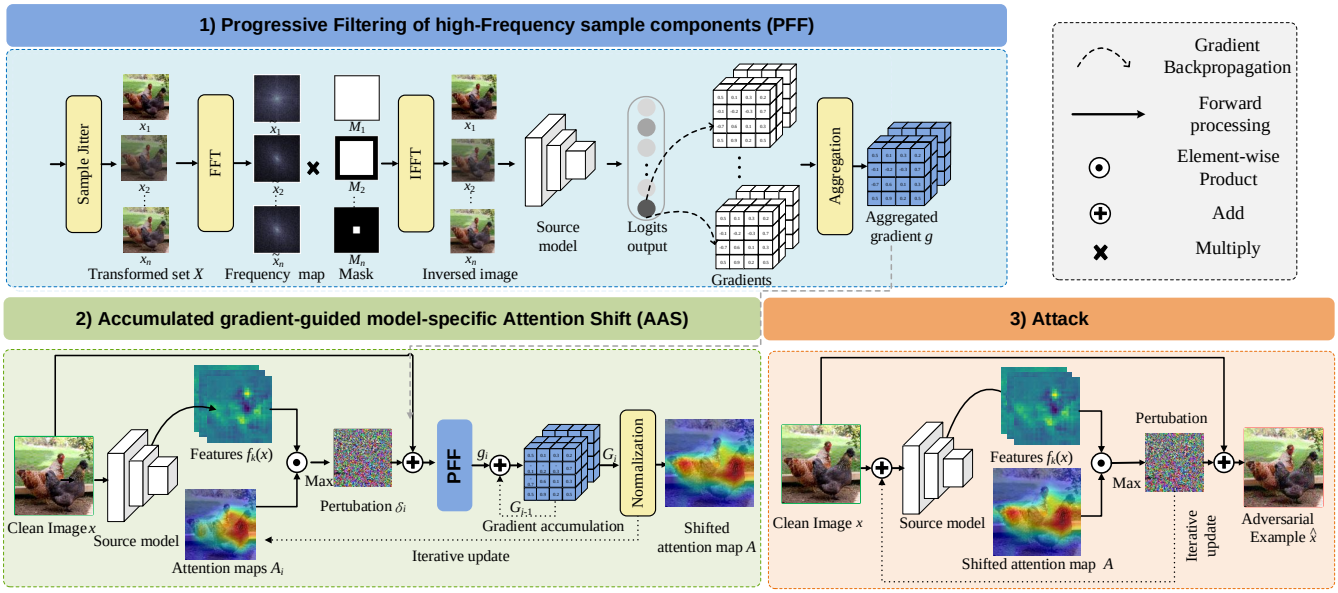

Figure 2: An overview of our BBMA. Given an input clean image x, the adversarial example is generated by iteratively maximizing the distortion on features with an attention map A. To reduce the model-level and sample-level bias, we propose PFF to progressively remove the high-frequency components in samples, and then use AAS to shift the model attention to avoid bias by the guidance of accumulated gradients.

features to enhance transferability, known as feature-level attacks. As an example, Feature Disruptive Attack(FDA) [12] introduces an attack method motivated by corrupting features at the targeted layer. The Neural Representation Distortion Method(NDRM) [33] maximizes the distance of internal features to disrupt features. Intermediate Level Attack (ILA) [18] fine-tunes existing adversarial examples by increasing the perturbation on a target layer. Feature Importance-aware (FIA) [41] measures the importance of internal features by aggregated gradients. Random Patch Attack (RPA) [46] applies patch-wise random transformations to get aggregate grad. Neuron Attribution-Based Attack (NAA) [48] uses neuron attribution methods to measure feature importance. Intermediate-Level Perturbation Decay (ILPD) [24] encourages the intermediate-level perturbation to be in an effective adversarial direction. Our proposed method also falls into this category.

2.2 Adversarial Defenses

In response to the adversarial attacks, various adversarial defenses have been proposed to defend against such attacks. One promising way is adversarial training [1, 13, 20, 31, 34, 47], which leverage the online generated adversarial examples into the training dataset so that the model can prefer more robust features during learning. Ensemble adversarial training [39, 44] explores combining multiple models to enhance robustness against adversarial attacks and improve defense strategies. While adversarial training effectively enhances model robustness, it incurs computational costs, making it impractical for large-scale datasets. To avoid this issue, many pre-processing methods have been proposed. State-of-art methods include the use of a High-level Representation Guided Denoiser (HGD) [25], randomly resizing [42], Randomized Smoothing (RS) [5], compressing input image [7, 11] and Neural Representation Purifier (NRP) [32]. In this paper, we employ these state-of-the-art defenses to evaluate the effectiveness of our attack method.

3 Methodology

3.1 Preliminaries

Given a target model f_θ with parameters θ and a clean image x with ground-truth label y, adversarial attacks generate an adversarial example by injecting imperceptible perturbation δ on the input image x and optimize the following maximization:

$$\arg\max_{\delta} \mathcal{L}_\theta(x+\delta, y), \text{ s.t. } \|\delta\|_p \leq \epsilon, \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ is the task loss, *e.g.*, cross-entropy loss in classification task. $\|\cdot\|_p$ denotes the l_p-norm constraint and ϵ is the radius of the l_p-norm ball. Such optimization aims to find the perturbation δ that can most deviate the model predictions from the true labels. Typically, under the white-box setting, *i.e.*, θ is accessible, the maximization can be generally achieved by inversing the gradient as:

$$\delta = \text{sign}\left(\nabla_x \mathcal{L}_\theta(x, y)\right), \tag{2}$$

where $\text{sign}(\cdot)$ is the sign function. Intuitively, adding the inversed gradients on inputs can maximize the model loss so that the gradient-based white-box attacks can achieve near 100% attack success rate. However, these advances in attack performance vanish when faced with black-box models, *i.e.*, the gradients are unavailable. The

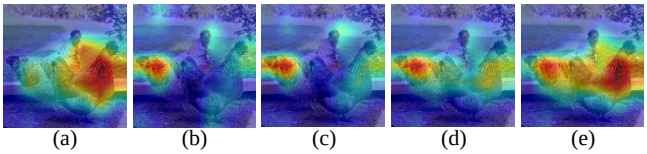

Figure 3: Illustration of the proposed attention shift mechanism. From (a)-(e): The initial attention map, the attention map generated from the iterations $I = 0, 1, 2$, and the final iteration. The attention can cover the object mostly as the gradients accumulate.

transfer-based methods are proposed to utilize a white-box surrogate model to generate AEs and attack the target model by improving the transferability. That is, obtaining $\delta = \text{sign}\left(\nabla_x \mathcal{L}_\phi(x, y)\right)$ from a white-box surrogate model f_ϕ and achieve the maximization of $\mathcal{L}_\theta(x + \delta, y)$ on target model f_θ. Since θ is variable and unpredictable, this process requires the perturbation δ to contain more generalized information.

3.2 Bi-Level Bias Mitigated Attack

To fully prompt the generalized knowledge in adversarial perturbations, we propose a Bi-Level Bias Mitigated Attack (BBMA) to mitigate bias in two levels. Specifically, within the classic framework of iterative gradient-based adversarial perturbation generation, BBMA first progressively removes the high-frequency components of samples to reduce model-specific bias and then applies accumulated gradient-guided model attention shift to mitigate model-specific bias. An overview of the whole pipeline is shown in Figure 2. We present the details in the following.

3.2.1 Progressive Filtering of High-Frequency Sample-Specific Components (PFF). **Sample Jitter.** We start with a sample jitter module to improve the input diversity. Specifically, we adjust the low-level properties of images including brightness, contrast, and saturation via $x = x \times \alpha_b$, $x = \bar{x} + \alpha_c \times (x - \bar{x})$, and $x = x_Y + \alpha_s \times (x - x_Y)$, respectively, where α_b, α_c, and α_s are the scale factors, $\bar{x}$ is the image mean value and x_Y is the Y component of image x. Note that the scale factors are randomly selected within a specified bound in each iteration. Such transformation can contribute to 1) enhancing the input diversity; and 2) the transformations in the low-frequency can encourage the general information extraction in DNN models. Thus, we can get a set of transformed samples denoted as $X = \{x_1, x_2, ..., x_N\}$, N is the total number of transformations.

Progressive high-frequency removal. To reduce the sample-specific bias, we make use of the property of DNNs in the frequency domain, *i.e.*, a well-trained DNN might capture high-frequency, sample-specific features, and propose to remove the high-frequency components of samples. We devise a stride-wise step-tuning strategy to alleviate the immoderate removal. Progressively removing high-frequency components simulates how a model prioritizes features at different stages, revealing which details are critical for decision-making and which can be disregarded. Specifically, for each x_i in X, we first apply the Fast Fourier Transform [6] (FFT) to transform the input from spatial domain into the frequency domain, followed by shifting the zero-frequency component to the center (SHIFT), which can be represented as:

$$\widetilde{x_i} = SHIFT(FFT(x_i)), \tag{3}$$

where $\widetilde{x_i}$ denotes the frequency map of image x_i. Next, we apply an all-0 mask M with a centered all-1 window w of size $s \times s$ to remove the high-frequency components. That is, when we apply the mask M on the frequency map $\widetilde{x_i}$, only the components under the window w remain. We use a stride d to gradually adjust the size s of the window w, which can be described as:

$$s = s - l \cdot d. \tag{4}$$

where l is the step number and the total number is denoted as L. s is initialized as the radius of $\widetilde{x}$. The mask M_l is then applied to $\widetilde{x_i}$ to produce the masked frequency domain representation:

$$\widetilde{x_i} = \widetilde{x_i} \odot M_l, \tag{5}$$

where $\odot$ is the element-wise multiplication. Finally, we rearrange the frequency domain image to move the zero-frequency component back to the edges (ISHIFT), and apply Inverse Fast Fourier Transform (IFFT) to transform the masked frequency-domain images back to the spatial domain, which can be represented as:

$$x_i = IFFT(ISHIFT(\widetilde{x_i})). \tag{6}$$

Note that in this process, we apply L masks with different values of s on the frequency map, and $L = N$. Consequently, N resulting images are generated with different extents of high-frequency removal. We denote the resulting images as $\mathcal{T}(x) = \{x_1, x_2, ..., x_N\}$, where $\mathcal{T}(\cdot)$ indicates the whole PFF process.

When fed to the source model, the gradient obtained is the average of the gradients in all the samples in $\mathcal{T}(x)$, which can be represented as:

$$\bar{g} = \frac{1}{N} \sum_{n=1}^{N} g(x_n). \tag{7}$$

Since semantically object-aware features and gradients are robust to the low-level transformation and the model-specific ones are vulnerable to the low-frequency components, those robust features and gradients will be highlighted after aggregation, while the others would be neutralized.

3.3 Accumulated Gradient-Guided Model Attention Shift

The model-level bias mitigation is insufficient in comprehensively prompting the adversarial perturbation generalization. It has been widely known that different models can perform differently in processing the same sample, exhibiting significant differences in the regions or features of the input data they focus on. An appealing solution is to maximize the distortion on more model-agnostic and object-relevant features thus it can be highly likely to disturb other models. In BBMA, we keep along with this solution and try to distort the features that are highly likely to be shared across models. Specifically, we follow the previous methods to evaluate the importance of features by utilizing gradients since they specify the contribution of features to the final decision. However, the source model gradients are also model-specific, thus limiting the transferability. To address this issue, we propose to use adversarial perturbations as a clue to shift the model attention, and then accumulate the model gradients on these perturbed images.

Algorithm 1 Bi-level Bias Mitigated Attack (BBMA)

Input: The clean image x, classification model f, intermediate layer k, ensemble number N, max perturbation ϵ, attack iteration T, attention shift iterations I, the extent of disruption D.
Output: The adversarial image x^{adv}
Initialize: $A = 0$, $g_0 = 0$, $\mu = 1$, $\alpha = \epsilon/T$, $\delta = 0$.

Obtain feature importance A_0 via PFF.
for $v = 0, 1, \cdots, I-1$ **do**
  $x_0^{adv} = x$
  for $u = 0, 1, \cdots, D-1$ **do**
    $\mathcal{L} = \Sigma(A_v \odot f_k(x_u^{adv}))$
    $g_{u+1} = \mu \cdot g_u + \frac{\nabla_x \mathcal{L}(x_u^{adv})}{\|\nabla_x \mathcal{L}(x_u^{adv})\|_2}$
    $x_{u+1}^{adv} = Clip_{x,\epsilon}\{x_u^{adv} - g_{t+1}\}$
  end for
  Update feature importance A_v by ***Eq***. (9).
end for
$x_0^{adv} = x$
$A = A_I$
for $t = 0, 1, \cdots, T-1$ **do**
  $\mathcal{L} = \Sigma(A \odot f_k(x_i^{adv}))$
  $g_{t+1} = \mu \cdot g_t + \frac{\nabla_x \mathcal{L}(x_t^{adv})}{\|\nabla_x \mathcal{L}(x_t^{adv})\|_2}$
  $x_{t+1}^{adv} = Clip_{x,\epsilon}\{x_t^{adv} - \alpha \cdot sign(g_{t+1})\}$
end for
return x_T^{adv}.

Given the source model f, the features of sample x from k-th layer are denoted as $f_k(x)$. The gradient with respect to $f_k(x)$ can be derived as:

$$g_{x,k} = \frac{\partial p(x,y)}{\partial f_k(x)}, \tag{8}$$

where $p(\cdot,\cdot)$ is the logit output w.r.t. the groud-truth label y. An attention map that can evaluate the importance of features can be generated via the normalization of $g_{x,k}$. To reduce the model-specific bias, we propose an accumulated gradient-guided attention shift mechanism. Specifically, we iteratively use adversarial perturbations as a clue to shift the model gradient, and accumulate the current gradients on the previous gradients, in which the accumulated gradient G_i in the i-th iteration can be represented as:

$$G_i = G_{i-1} + \frac{1}{N}\sum_{n=0}^{N} g_{\mathcal{T}(x_n+\delta_i),k}, \tag{9}$$

where δ_i is the adversarial perturbation generated in the i-th iteration and can be obtained by optimizing Eq. (2). Note that the objective function $\mathcal{L}$ here in Eq. (2) is the same as that in the final attack as described in Eq. (11). $\mathcal{T}(x_n + \delta_i)$ denotes the PFF process for the image $x_n + \delta_i$. The attention map A_i can be derived by normalizing G_i as:

$$A_i = \frac{G_i}{\|G_i\|_2}. \tag{10}$$

We show an intuitive example in Fig. 3, where we can see that the final attention map can cover correct and more objects as the iteration increases.

3.4 Attack Algorithm

After obtaining the attention map A, we follow the general pipeline of feature-level attacks and aim to distort the key features by utilizing the following objective function:

$$\mathcal{L}(x) = \Sigma(A \odot f_k(x)), \tag{11}$$

and the optimization can be represented as:

$$\arg\min_{\delta} \mathcal{L}(x+\delta, y), \ s.t. \ \|\delta\|_\infty \le \epsilon. \tag{12}$$

This minimization can be achieved by the existing gradient-based attacks. Here in our method, we adopt the classic and the strongest MI-FGSM attack integrating momentum into the gradient to iteratively generate δ and generate the adversarial example x^{adv}. A summarized algorithm of the proposed BBMA is shown in Algorithm 1.

4 Experiments

4.1 Experimental Setup

Dataset. We follow the previous work [41, 48] to conduct our experiments on the ImageNet-compatible dataset, which contained 1,000 images used for the NIPS 2017 adversarial competition. We also follow the experimental setup of AdaEA [3] and conducted experiments on the CIFAR datasets using their pre-trained models.
Models. For our experiments on the ImageNet dataset, we choose models from both branches of CNNs and ViTs for the black-box attack task. The normally trained models are ResNet18 (RN-18) [35], ResNet50(RN-50) [35], ResNext50 (RX-50) [43], DenseNet121 (DN-121) [17], Inception-v3 (Inc-v3) [36], Vgg16 (Vgg-16) [14], ViT [10], PiT [16], Visformer (VF) [4] and Swin [27]. With adversarial training [22, 39], the corresponding defense models are Adv-Inc-v3 (Inc-v3$_{adv}$), Ens3-Inc-v3 (Inc-v3$_{Ens3}$), Ens4-Inc-v3 (Inc-v3$_{Ens4}$), and Ens-IncRes-v2 (IncRes-v2$_{Ens}$). Moreover, we also tested on three defense methods namely HGD [25], RS [12], and NRP [32]. For the CIFAR datasets, we choose ResNet50 as the substitute model and select targeted models from both branches of CNNs and ViTs for the black-box attack task, including ResNet-18 (RN-18) [35], WideResNet-101-2 (WRN101-2) [45], Inception v3 (Inc-v3) [36], BiT-M-R50×1 (BiT-50) [21] in CNN branch; and ViT-Base (ViT-B) [9], ViT-Tiny (ViT-T) [9], DeiT-Base (DeiT-B) [38], DeiT-Tiny (Deit-T) [38], swin-Base (Swin-B) [27] and Swin-Small (Swin-S) [27] in ViT branch.
Implementation Details. For the ImageNet dataset, by default, during the attack process, the maximum perturbation was set to $\epsilon = 16$, the number of iterations $T = 10$, and the step size $\alpha = \epsilon/T$ across all experiments. For the CIFAR-10 dataset, the parameters were set as $\epsilon = 8/255$, $\alpha = 2/255$, $T = 10$. For SSA [28], FIA [41] and RPA [46], the ensemble number $N = 30$, the patch size of RPA alternately sets $n = 1, 3, 5, 7$. For NAA [48], we let $\gamma = 1$ and the transformation functions degrade to the linear functions. For ILPD [24], we run 100 iterations with a step size of $1/255$. For PFF process, the progressive number $N = 20$, $I = 3$, $D = 10$, $\beta_b = 1.0$, $\beta_c = 1.0$, $\beta_s = 0.3$, $d = 5$. Furthermore, we set the decay factor $\mu = 1.0$ for all the baselines because all the baselines utilize the momentum method as the optimizer. For feature-level attacks, we choose the same layer, *i.e.*, Conv3_2 for RN-50, Conv3_3 for Vgg-16, Mixed_6b for Inc-v3.

Source Models	Attack	CNNs							VITs				Average
		RN-18	RN-50	Inc-v3	Vgg-16	RN-101	RX-50	DN-121	ViT	PiT	VF	Swin	
RN-50	MI[8]	42.9	80.4	32.6	45.4	28.7	33.1	39.6	10.6	14.8	19.5	25.5	33.9
	SSA[28]	76.3	93.0	71.6	78.5	76.6	77.9	79.0	44.8	58.6	64.3	63.5	71.3
	FIA[41]	74.9	76.9	64.8	73.7	52.5	55.0	71.7	24.0	28.3	36.2	43.9	54.7
	NAA[48]	57.2	60.4	45.9	62.9	42.0	44.5	56.3	19.7	23.9	31.0	35.4	43.6
	RPA[46]	75.0	88.4	62.8	75.7	71.0	71.2	76.8	27.4	39.9	52.5	53.5	63.1
	ILPD[24]	75.9	87.6	66.5	79.1	67.0	68.8	72.4	35.1	46.9	53.0	53.9	64.2
	Ours	**91.2**	**99.2**	**81.6**	**91.7**	**91.4**	**91.3**	**93.8**	**49.6**	**64.4**	**78.0**	**75.5**	**82.5**
Inc-v3	MI[8]	46.8	30.7	98.0	49.7	28.1	29.2	44.9	15.8	18.7	24.7	29.4	37.8
	SSA[28]	72.7	58.5	99.4	71.8	55.8	59.0	73.4	32.1	37.1	45.0	50.7	59.6
	FIA[41]	79.2	59.7	99.6	78.4	53.1	56.2	78.7	27.6	31.6	43.3	48.2	59.6
	NAA[48]	73.0	58.7	99.4	72.6	55.7	56.0	73.7	29.9	35.5	46.0	49.6	59.1
	RPA[46]	79.0	65.0	99.4	80.8	62.0	62.9	79.4	34.9	39.9	49.3	53.6	64.2
	ILPD[24]	79.7	66.1	98.0	79.0	65.1	64.3	79.6	39.1	45.9	56.2	54.4	66.1
	Ours	**91.3**	**80.1**	**100.0**	**90.4**	**77.3**	**77.8**	**91.7**	**46.6**	**50.4**	**65.6**	**66.1**	**76.1**
Vgg-16	MI[8]	67.9	42.4	46.3	99.8	32.6	39.3	61.2	14.8	21.4	30.4	37.3	44.9
	SSA[28]	92.7	69.7	73.9	**100.0**	59.0	65.9	87.5	29.6	38.8	57.2	63.1	67.0
	FIA[41]	76.9	52.1	50.5	99.3	38.2	42.8	67.5	16.1	20.7	35.2	38.1	48.9
	NAA[48]	84.1	69.2	69.1	99.9	59.8	64	89.1	27.5	40.6	56.3	60.0	65.4
	RPA[46]	89.6	68.7	64.7	99.9	58.0	62.2	84.4	22.6	30.6	48.0	51.5	61.8
	ILPD[24]	94.0	83.1	74.5	99.9	73.1	77.0	91.3	31.8	46.5	65.9	68.5	73.2
	Ours	**98.2**	**88.4**	**85.0**	**100.0**	**82.6**	**87.2**	**96.8**	**39.1**	**55.5**	**76.6**	**77.7**	**80.6**

Table 1: The attack success rates (%) on undefended models by various momentum optimization-based attacks. The bolded numbers indicate the best results, while the shaded cells represent our method.

Source Model	Attack	Desfense Models				Defense Methods			Average
		Inc-v3$_{Adv}$	Inc-v3$_{Ens3}$	Inc-v3$_{Ens4}$	IncRes-v2$_{Ens}$	HGD	RS	NRP	
RN-50	MI[8]	21.1	21.9	18.0	14.8	16.3	15.1	21.7	18.4
	SSA[28]	67.5	65.5	64.4	58.2	64.5	41.5	65.2	61.0
	FIA[41]	51.1	48.1	47.8	38.2	41.1	32.1	42.5	43.0
	NAA[48]	34.7	32.4	29.9	25.8	30.1	24.5	29.6	29.6
	RPA[46]	53.7	51.3	48.5	40.3	49.0	27.2	48.3	45.5
	ILPD[24]	61.0	56.9	56.4	50.9	57.1	39.2	54.6	53.7
	Ours	**77.4**	**76.1**	**72.0**	**64.9**	**77.5**	**43.5**	**66.6**	**68.3**
Inc-v3	MI[8]	35.5	32.5	31.5	26.7	27.1	20.5	22.2	28.0
	SSA[28]	72.6	67.5	65.8	55.7	58.9	36.2	42.5	57.0
	FIA[41]	72.3	66.8	65.4	51.6	55.7	27	32	53.0
	NAA[48]	67.6	64.1	63.3	52.0	64.4	34.5	34.3	54.3
	RPA[46]	75.1	73.0	69.4	59.4	62.7	32.1	39.0	58.7
	ILPD[24]	75.1	73.8	70.8	62.2	66.4	35.5	40.1	60.6
	Ours	**90.8**	**86.6**	**84.8**	**74.4**	**79.8**	**43.0**	**47.6**	**72.4**
Vgg-16	MI[8]	33.1	32.0	28.3	21.6	27.1	17.5	22.1	26.0
	SSA[28]	66.6	60.7	57.2	43.0	55.8	34.1	**42.5**	51.4
	FIA[41]	39.4	36.0	30.1	24.3	31.0	15.5	21.5	28.3
	NAA[48]	59.6	56.6	52.9	41.6	53.0	24.1	31.3	45.6
	RPA[46]	56.7	51.9	45.5	33.5	47.7	21.2	28.9	40.8
	ILPD[24]	66.4	64.4	61.0	48.8	65.3	34.5	37.4	54.0
	Ours	**78.1**	**75.1**	**72.2**	**59.0**	**75.4**	**35.5**	42.1	**62.5**

Table 2: The attack success rate (%) of different attacks against defense models and defense methods. The bolded numbers indicate the best results, while the shaded cells represent our method.

4.2 Main Results

We first compare our proposed method with two input transformation-based attacks, MI and SSA, as well as four feature-level attacks, FIA, NAA, RPA, and ILPD. We choose RN-50, Inc-v3, Vgg-16 as the source model for ImageNet dataset, and RN-50 for the CIFAR datasets. For ImageNet dataset, we generate the adversarial examples on a single model and test them on the other models, including normally trained models (Table 1), defense models, and defense

Datasets	Method	RN-50	RN-18	WRN-101-2	Inc-v3	Bit-50	ViT-B	ViT-T	DeiT-B	Deit-T	Swi-B	Swin-S	Average
CIFAR-10	MI[8]	**100.0**	52.4	34.5	79.4	25.3	5.0	15.1	8.5	10.8	14.9	24.2	33.6
	SSA[28]	**100.0**	78.9	62.3	80.1	40.7	10.5	33.6	17.5	20.6	36.9	52.5	48.5
	FIA[41]	61.3	19.0	15.2	73.0	11.2	0.9	2.8	0.8	2.1	2.4	5.0	17.6
	NAA[48]	90.3	64.0	51.1	79.3	35.6	10.2	27.5	16.2	18.9	30.4	40.9	42.2
	ILPD[24]	99.5	92.2	**83.0**	80.1	48.5	10.5	42.0	20.5	24.8	51.9	70.1	56.6
	BBMA	99.9	**93.2**	80.4	**83.7**	**51.5**	**13.5**	**45.2**	**24.0**	**27.8**	**52.4**	**71.9**	**58.5**
CIFAR-100	MI[8]	99.9	70.3	62.7	**65.9**	51.8	24.3	42.1	33.9	39.2	36.1	48.6	52.3
	SSA[28]	**100.0**	91.0	78.9	85.4	68.7	36.1	67.4	51.1	59.7	60.9	77.8	70.6
	FIA[41]	97.6	67.9	70.7	69.6	53.8	17.5	36.1	21.7	31.3	30.8	46.3	49.4
	NAA[48]	97.6	79.7	71.7	77.9	63.7	37.9	59.2	50.0	52.8	60.1	69.2	65.4
	ILPD[24]	98.9	89.7	80.2	86.8	70.7	**47.3**	73.9	**64.7**	**67.5**	78.3	86.1	76.7
	BBMA	**100.0**	**95.1**	**82.9**	**89.8**	**72.8**	45.5	**76.0**	63.4	66.1	**79.8**	**87.3**	**78.1**

Table 3: The attack success rates (%) on undefended models by various momentum optimization-based attacks. The bolded numbers indicate the best results, while the shaded cells represent our method.

methods (Table 2). For the CIFAR datasets, we test on models that are normally trained (Table 3).

Attacking Normally Trained Models. We first craft adversarial examples on RN-50, Inc-v3, and Vgg-16 to attack various normally trained models in Table 1. As for the black-box performance, our method consistently surpasses well-known baseline attacks on both CNN-based and transformer-based models. In particular, our method outperforms the runner-up by a maximum of 11.2% and achieves an attack success rate that is at least 7.4% higher than the best baseline across all models. Such consistent and superior performance demonstrates that the proposed BBMA can boost transferability to various model architectures.

Attacking Defense Models. To further verify the superiority of our method, we conduct attack experiments against four adversarially-trained models and three defense methods. The results are reported in Table 2. We can observe that our algorithm can significantly boost existing attacks. The attack performance on adversarially-trained models and defense methods is significantly improved by a margin of 9.2% on average. In particular, our method outperforms the runner-up by a maximum of 11.8% and achieves an attack success rate that is at least 7.3% higher than the best baseline across all models. It is worth noting that under NRP defense, the performance of SSM slightly surpasses our method when the source model is Vgg-16. We speculate that this is because NRP maximizes the distance between the model's intermediate layer features for clean and adversarial examples, resulting in better defense against feature-level perturbations. Despite this, our method achieves the best performance across all the tasks.

Additional experimental results on CIFAR datasets. We also conduct experiments on CIFAR datasets and summarize the results in Table 3. As shown in the table, our experiments on the CIFAR datasets demonstrate that our method consistently outperforms existing baseline attacks across both CNN and transformer-based models. These results provide strong evidence for the effectiveness and broad generalizability of our method in improving adversarial transferability across a wide range of model architectures and benchmark datasets.

Visualization of attack performance. To intuitively show the attack performance, we visualize the heatmaps of clean image and adversarial examples generated by different methods in both source model and target models in Fig. 4. As can be observed, other methods can disrupt label-related features in the image to some extent. However, they fail to capture more generalized features and tend to overfit the source model, allowing model-agonistic and class-relevant features to remain intact. In contrast, our method effectively disrupts analogous robust features in both the source and target models, preventing them from capturing class-relevant features.

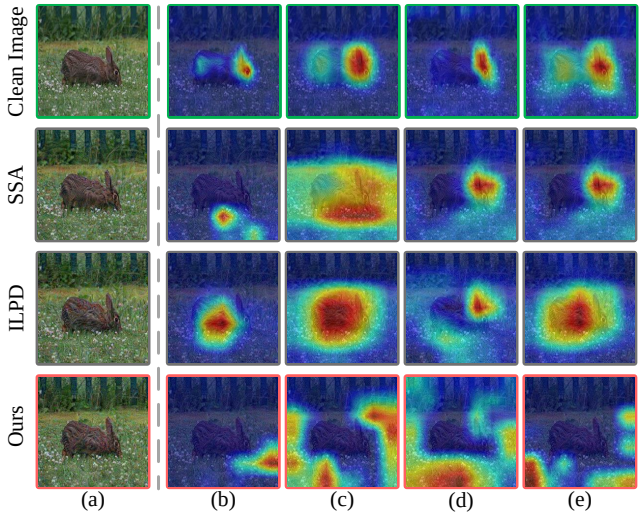

Figure 4: Heatmaps of different inputs in the source model and target models, with the class label set to 'rabbit'. (a) input images, including clean images and adversarial examples generated by each attack method. (b)-(e) are the heatmaps on the surrogate models (RN-50) and target models (RN-18, Vgg-16, DN-161), respectively.

We also utilize Spectrum Saliency Map [28] to visualize the changes brought about by BMBA. As seen in the Fig. 5, FIA and RPA primarily focus on the low-frequency components of the model, lacking effective utilization of mid and high frequency information. BMBA can mitigate the model's bias toward high frequencies, while also making the importance assessment for low and mid frequencies smoother and more accurate.

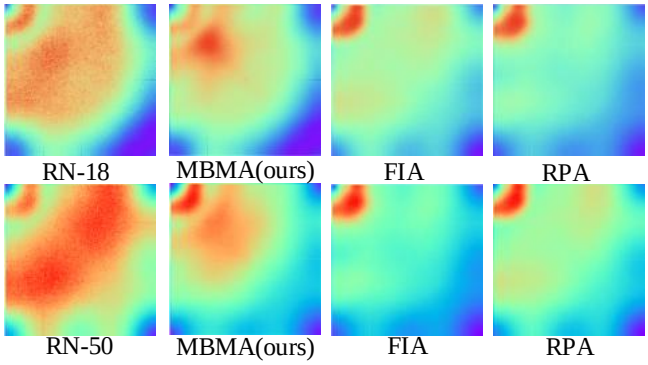

Figure 5: The spectrum saliency map [28] generated in different attacks on RN-50 and RN-18. Compared to the other attacks, our method preserves low-frequency components and remain proper high-frequency components.

Attack	RN-50	Inc-v3	Vgg-16
BBMA w/o all	32.3	50.2	59.5
BBMA w/o PFF	54.1	64	63.9
BBMA w/o AAS	57.2	64.0	73.5
BBMA	**82.5**	**76.1**	**80.6**

Table 4: The average attack success rates (%) on normally trained models. The adversarial examples are crafted via RN-50, Inc-v3, and Vgg-16, respectively.

4.3 Ablation Studies

In this subsection, we conduct a series of ablation experiments to study the effects of key components in our BBMA. To further gain insight into the performance improvement of BBMA, we conduct hyper-parameter studies by generating the adversarial examples on Souce Models and evaluating them on the normally trained models.

On the components of BBMA. To further understand the superior attack performance achieved by our BBMA, we conduct a series of ablation studies to validate that the PFF, and AAS modules contribute to improved transferability. The results are summarized in Tab. 4. From the result, we observe that PFF, and AAS modules are useful for enhancing the transferability of adversarial examples.

On the target layer k. Early layers primarily extract data-specific features, while later layers focus on model-specific features to optimize classification, making them less ideal for transferable attacks. In contrast, middle layers, with well-separated class representations and less reliance on model architecture, are the best targets for improving transferability. We apply BBMA to various target layers to craft adversarial examples and evaluate their transferability, using RN-50, Inc-v3 as the source models. As shown in Fig. 6, attacking deeper layers (conv3_2, conv3_3) yields the best transferability. Therefore, we select these layers as the target layers.

On the progressive number N. To determine a good value for N, we evaluate BBMA with N from 0 to 20 and report the attack success rate in Fig. 7. When we increase the value of N, the impact of model-level bias can be mitigated, resulting in more accurate feature importance with BBMA.

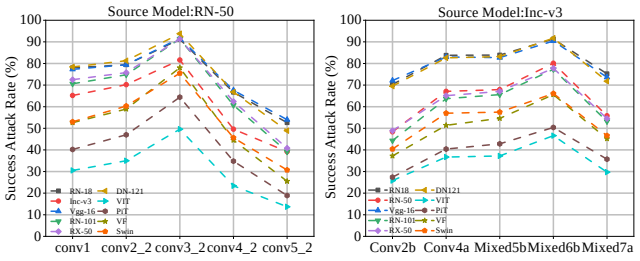

Figure 6: Effect of layer choice on attack success rate. Different layers from the source models are selected to generate adversarial examples, whose success rates are reported against different normally trained models.

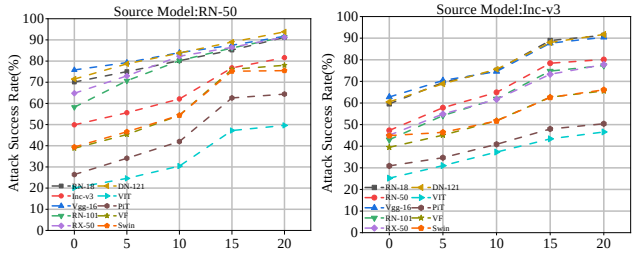

Figure 7: Attack success rates (%) of adversarial examples generated by BBMA with various number of images N. As the value of N increases, the attack success rate improves and reaches the peak value at 20.

5 Conclusion

In this work, we focus on the generalization of adversarial examples, and proposed a Bi-Level Bias Mitigated Attack (BBMA) to generate highly transferable adversarial examples. The proposed BBMA employs Progressive Filtering of High-Frequency Sample-Specific Components (PFF) to mitigate the model-level bias and obtain more accurate feature importance. The Accumulated gradient-guided model-specific Attention Shift (AAS) module is further introduced to redirect the model's focus and uncover more robust features in the source model, thereby mitigating sample-level bias. Extensive evaluations demonstrate that our proposed BBMA can achieve remarkably better transferability than the existing state-of-the-art attacks from multiple analysis in terms of both qualitative and quantitative perspectives.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China under Grant Nos. 62202104, U2441239, 62172243, 62402425, 62302468, U244120033, U24A20336, and 62402418; the China Postdoctoral Science Foundation under No. 2024M762829, the Zhejiang Provincial Natural Science Foundation under No. LD24F020002, the "Pioneer and Leading Goose" R&D Program of Zhejiang under 2025C01082, 2025C02033 and 2025C02263, and the Zhejiang Provincial Priority-Funded Postdoctoral Research Project under No. ZJ2024001.

References

[1] Qi-Zhi Cai, Min Du, Chang Liu, and Dawn Song. 2018. Curriculum adversarial training. *arXiv preprint arXiv:1805.04807* (2018).

[2] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*. Ieee, 39–57.

[3] Bin Chen, Jiali Yin, Shukai Chen, Bohao Chen, and Ximeng Liu. 2023. An adaptive model ensemble adversarial attack for boosting adversarial transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4489–4498.

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

[5] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*. PMLR, 1310–1320.

[6] James W Cooley and John W Tukey. 1965. An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation* 19, 90 (1965), 297–301.

[7] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2019. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 113–123.

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

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

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, and Sylvain and Gelly. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[11] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2018. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1625–1634.

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

[13] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. EXPLAINING AND HARNESSING ADVERSARIAL EXAMPLES. *stat* 1050 (2015), 20.

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

[15] Xianglong He, Yuezun Li, Haipeng Qu, and Junyu Dong. 2023. Improving transferable adversarial attack via feature-momentum. *Computers & Security* 128 (2023), 103135.

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

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2016. Densely Connected Convolutional Networks. *IEEE Computer Society* (2016).

[18] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. 2019. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4733–4742.

[19] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems* 32 (2019).

[20] S Kariyappa and MK Qureshi. 1901. Improving adversarial robustness of ensembles with diversity training. 2019. *Available: arXiv* (1901).

[21] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. 2020. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, 491–507.

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

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

[24] Qizhang Li, Yiwen Guo, Wangmeng Zuo, and Hao Chen. 2024. Improving adversarial transferability via intermediate-level perturbation decay. *Advances in Neural Information Processing Systems* 36 (2024).

[25] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. 2018. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1778–1787.

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

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

[28] Yuyang Long, Qilong Zhang, Boheng Zeng, Lianli Gao, Xianglong Liu, Jian Zhang, and Jingkuan Song. 2022. Frequency domain model augmentation for adversarial attack. In *European conference on computer vision*. Springer, 549–566.

[29] Tao Luo, Zheng Ma, Zhi-Qin John Xu, and Yaoyu Zhang. 2019. Theory of the Frequency Principle for General Deep Neural Networks. *ArXiv* abs/1906.09235 (2019). https://api.semanticscholar.org/CorpusID:195317121

[30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* (2017).

[31] Aleksander Mądry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *stat* 1050, 9 (2017).

[32] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. 2020. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 262–271.

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

[34] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. 2019. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*. PMLR, 4970–4979.

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

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

[37] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *International Conference on Learning Representations*. http://arxiv.org/abs/1312.6199

[38] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*. PMLR, 10347–10357.

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

[40] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. 2021. Admix: Enhancing the transferability of adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 16158–16167.

[41] Zhibo Wang, Hengchang Guo, Zhifei Zhang, Wenxin Liu, Zhan Qin, and Kui Ren. 2021. Feature importance-aware transferable adversarial attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*. 7639–7648.

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

[43] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2016. Aggregated Residual Transformations for Deep Neural Networks. *IEEE* (2016).

[44] Huanrui Yang, Jingyang Zhang, Hongliang Dong, Nathan Inkawhich, Andrew Gardner, Andrew Touchet, Wesley Wilkes, Heath Berry, and Hai Li. 2020. Dverge: diversifying vulnerabilities for enhanced robust generation of ensembles. *Advances in Neural Information Processing Systems* 33 (2020), 5505–5515.

[45] Sergey Zagoruyko and Nikos Komodakis. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146* (2016).

[46] Jianping Zhang, Weibin Wu, Jen-tse Huang, Yizhan Huang, Wenxuan Wang, Yuxin Su, and Michael R Lyu. 2022. Improving adversarial transferability via neuron attribution-based attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14993–15002.

[47] Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. 2020. Attacks which do not kill training make adversarial learning stronger. In *International conference on machine learning*. PMLR, 11278–11287.

[48] Yaoyuan Zhang, Yu-an Tan, Tian Chen, Xinrui Liu, Quanxin Zhang, and Yuanzhang Li. 2022. Enhancing the Transferability of Adversarial Examples with Random Patch.. In *IJCAI*. 1672–1678.