Generative Imperceptible Attack With Feature Learning Bias Reduction and Multi-Scale Variance Regularization

Weicheng Xie, *Member, IEEE*, Zenghao Niu, Qinliang Lin, Siyang Song, and Linlin Shen, *Senior Member, IEEE*

***Abstract*— Existing studies have shown that malicious and imperceptible adversarial samples may significantly weaken the reliability and validity of deep learning systems. Since gradient-based attack algorithms may result in higher generation latency or demand large computation overhead, generative attack methods are frequently considered. However, the effectiveness and imperceptibility are still the main concerns for these generative attacks, 1) biased feature learning may occur, i.e., these algorithms may generate undesirable feature perturbations for samples that are less likely to be successfully attacked; 2) the produced perturbation noises may be easily perceived by human eyes. To this end, we propose a novel generative attack by manipulating the feature update. The proposed algorithm has two main merits, 1) our Bias-reduced Feature Manipulation (BrFM) that differentiates the hard-to-attack (Hard2Attack) and easy-to-attack (Easy2Attack) features, can avoid the possible learning shortcut for different difficulties of features in attack process, by customizing perturbations for Hard2Attack features to make them behave oppositely to those of benign features; 2) our Multi-scale Variance Regularization (MsVR) can reduce the unnatural transitions of perturbations in mask edges and flat areas with low contrast, while simultaneously trading off a reasonable attack capacity. Extensive experiments on the datasets of Caltech-101 and Imagenette in terms of the attack success rate and four imperceptibility metrics, show the effectiveness of our attack paradigm over the related state-of-the-art generative attack methods. Our codes will be made publicly available.**

Manuscript received 17 March 2024; revised 23 July 2024; accepted 14 August 2024. Date of publication 29 August 2024; date of current version 6 September 2024. This work was supported in part by the Natural Science Foundation of China under Grant 62276170 and Grant 82261138629, in part by the Science and Technology Project of Guangdong Province under Grant 2023A1515011549 and Grant 2023A1515010688, in part by the Science and Technology Innovation Commission of Shenzhen under Grant JCYJ20220531101412030, and in part by Guangdong Provincial Key Laboratory under Grant 2023B1212060076. The associate editor coordinating the review of this article and approving it for publication was Dr. Dusit Niyato. *(Corresponding author: Linlin Shen.)*

Weicheng Xie, Zenghao Niu, and Qinliang Lin are with the Computer Vision Institute, School of Computer Science and Software Engineering, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Guangdong Provincial Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China.

Siyang Song is with the School of Computer Science, University of Exeter, EX4 4PY Exeter, U.K.

Linlin Shen is with the Computer Vision Institute, School of Computer Science and Software Engineering, Shenzhen Institute of Artificial Intelligence and Robotics for Society, Guangdong Provincial Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China, and also with the Department of Computer Science, University of Nottingham, Ningbo 315100, China (e-mail: llshen@szu.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIFS.2024.3451689, provided by the authors.

Digital Object Identifier 10.1109/TIFS.2024.3451689

***Index Terms*— Generative adversarial attack, imperceptible perturbation, imperceptibility metric, feature regularization loss, robust object classification.**

I. Introduction

DEEP neural networks have been shown to be vulnerable in a variety of computer vision (CV) tasks, e.g., classification, detection and segmentation. Traditional attack methods are gradient-based, such as L-BFGS [4], Fast Gradient Sign Method (FGSM) [5], Basic Iterative Method (BIM) [6] and Projecture Gradient Descent (PGD) [7].

Subsequently, numerous gradient attack works based on FGSM [5], such as MI-FGSM [8], NI-FGSM [9], and IE-FGSM [10] were developed. To make adversarial samples more robust, RAP [11] and PGN [12] sought adversarial examples in a flat local region. Meanwhile, the category of gradient attack approaches that enhance the generalization capacity of adversarial samples through input transformation have been widely studied, such as DIM [13], TIM [14], Admix [15], SSM [16], and DeCowA [17]. Recently, gradient attack approaches based on diffusion models were proposed to make use of their unrestricted and high-fidelity characteristics, such as adversarial guidance in reverse diffusion [18], generating imperceptible perturbations in latent space [19], and DiffAttack [20] that is proposed to against diffusion-based defenses.

Gradient-based attack methods often achieve reasonable white-box attack success rates and enable an adversarial sample to well transfer across different models. However, the above gradient-based attack methods, including the gradient attack methods based on diffusion models, often suffer from the requirement of the target model and image labels in the inference stage. With the introduction of AdvGan [21], a lot of works on producing attack perturbations via generative models were proposed, where the target model or the ground truth label of images is not required at the inference stage. Meanwhile, they generate attack samples much faster once the model is trained than the gradient-based attacks, especially those based on diffusion models.

Despite the advantages of generative adversarial attack methods, the attack capability and imperceptibility of generated perturbations are still the two main concerns. Lu et al. [1] innovatively use features obtained from models that do not contain fully connected (FC) layers, for attack training, thus eliminating reliance on real labels throughout the processing.

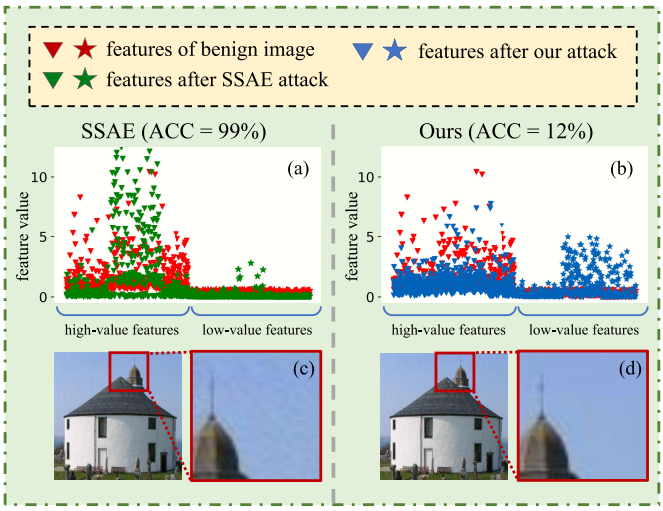


Fig. 1. Differences between SSAE [1] (the 1st column) and our method (the 2nd column) in attack effect ((a),(b)) as well as the imperceptibility of the results ((c),(d)) when using GoogLeNet [2] as the target model. All the features are averaged over those of samples with the golf ball class in Imagenette [3]. (b) shows that the resulted high-value features by our attack are desirably lowered, and the low-value features are desirably enlarged.

Additionally, it is shown that generative models obtained by training attacks in the feature space can generate adversarial samples with more powerful attack capacity compared to previous trained generative models [21] when labels are used.

For more powerful attack capacity in generative attack methods, the perturbations in the feature space are frequently considered, where the cosine similarity loss or L2 norm loss is used as the feature regularization constraint. SSAE [1] reduces the cosine similarity between adversarial and clean samples before the FC layer, while BIA [22] reduces this similarity at the output of a specified layer. LTAP [23] and ADA [24] enlarge the L2 norm between the features of attacked and original samples, and LPD [25] combines L2 norm with local patch difference for the attack over different feature spaces in multi-object scenes.

Despite the improved attack performance by the use of the aforementioned two losses, they suffer from a dilemma of learning bias for features with different difficulties. As shown in Fig. 1(a), when attacking the difficult golf ball class with SSAE [1], the high-value features of the generated adversarial samples are undesirably enlarged (left part), and the low-value features are undesirably lowered (right part). This may be due to the cosine similarity loss used by SSAE, which tends to take a learning shortcut when dealing with features of hard samples, i.e., maximizing its loss value regardless of the success of the hard-to-attack features.

To address the above problem, we categorize the features into the hard-to-attack (Hard2Attack) and easy-to-attack (Easy2Attack) ones according to the consistency degree of the features of the benign and adversarial samples, and accordingly specify the update direction for each category of features, to adaptively perturb the Hard2Attack ones and overlook the Easy2Attack ones. The corresponding attack effect is visualized in Fig. 1(b), and thanks to this, our attack successfully reduces the classification accuracy (ACC) on this difficult class.

In addition to enhancing the attack capacity, the imperceptibility of the perturbation noises to the perception of human eyes, is another main concern. As shown in Fig. 1(c), the most frequently applied constraints based on the $\mathcal{L}_p$ criterion [26] are no longer sufficient to further optimize the imperceptibility with respect to (w.r.t.) human eyes, since there is a large bias between model perception and human eye perception. Specifically, human eyes utilize the eye hopping mechanism for information processing [27], its relative imperceptibility of perturbation noises is somewhat incompatible with existing evaluation metrics that process the information pixel by pixel. To improve this human-perception imperceptibility, the latest constraints [28] on image structural noises, as well as constraint distances [29] in the color space, constraints [30] on low-frequency spatial limitations, and entropy-based contrast or information richness [31], have made great strides in optimizing the pixel distribution.

To further take into account the characteristics of human eye perception, inspired by the study [32] that the human eye's ability to perceive noise in high-contrast regions is much lower than that in low-contrast regions, Luo et al. [33] attempted to use variance as a contrast metric to craft imperceptible regions to improve the invisibility of aggressive perturbations w.r.t. human eyes. However, the single scale of the crafted imperceptible regions [33] may make the perturbation noises distribute on the region boundaries that are quite easy to be perceived, and attack capacity may be largely impaired. To this end, we devise a multi-scale variance mask to weigh an image globally, and use it as a soft constraint to regularize the perturbation noises, so as to further improve their human-perception imperceptibility while maintaining a reasonable attack capacity.

To realize our imperceptible attack modules, a CNN-based U-Net architecture is used as the generative model. Though the Transformer [34] and the Variational Autoencoder (VAE) [35] architectures seem to be promising, they are not suitable for this task due to reasons: (1) For a Transformer, its training requires extensive data, and it typically has a slower inference speed compared to a CNN architecture. (2) For VAE, the pixel-level reconstruction loss may result in overly smooth outputs, making them better suited for tasks that generate low-frequency contents, while adversarial perturbations are primarily high-frequency.

In summary, this work makes the following contributions:

- To avoid learning shortcut in feature perturbations with traditional losses, we propose a new feature-based regularization loss that differentiates Hard2Attack and Easy2Attack features to reduce the learning bias, and optimize the features of the perturbed sample to behave oppositely to those of the benign sample.
- To trade off the attack capacity and the imperceptibility of perturbation noises aware to human eyes, we propose a novel generative framework of imperceptible masks as a soft regularization, i.e., devising multi-window variance-based masks with different perceptual scales to limit perturbations within more difficult-to-perceive regions while maintaining a reasonable attack capacity.

- Quantitative and qualitative experiments demonstrate the effectiveness of our method in terms of both attack success rate and imperceptibility metrics under various scenarios, compared with the state of the arts.

II. Related Work

A. Gradient-Based Adversarial Attack Methods

With the increasing security awareness in the applications of deep learning, network adversarial attack and defense have been hot topics. Szegedy et al. [4] first revealed that adding tiny pixel perturbations can result in incorrect inference to classification network, and proposed L-BFGS. Furthermore, Fast FGSM based on gradient solving [5], applying FGSM multiple times to develop BIM [6], improving BIM to develop PGD [7], Momentum Iterative Fast Gradient Sign Method (MI-FGSM) [8], Nesterov Iterative (NI-FGSM) [9], improved Euler's (IE-FGSM) [10], devising adversarial samples in flat local regions [11], [12], and exploring gradient cues in the parameter space of color filters [36], were studied.

However, the above methods may still suffer from issues such as insufficient robustness or transferability of adversarial samples [20]. Thus, leveraging the powerful capability of pre-trained diffusion models in image generation, Kang et al. [20] proposed to attack diffusion-based defenses using a deviated-reconstruction loss, Dai et al. [18] conducted adversarial sampling during the reverse generation process of diffusion models, Liu et al. [19] proposed the conditioned diffusion generative model that is identity-sensitive. Though these diffusion model-based methods demonstrate the potential in adversarial attack tasks, they still suffer from slow generation speed and the demand of the target model during the inference process.

B. Generative Adversarial Attack Methods

Gradient-based attack methods including those based on diffusion models mentioned above generally require access to both the target model and the ground truth labels. Especially, diffusion models demand gradient cues of surrogate models [18], [20], and [19] and standard true labels that are often unavailable in generating adversarial samples, and they face low efficiency. By contrast, the universal attacks with generation approaches that do not require access to the target model during inference, can produce perturbations freely once trained.

Recently, GAN-based (AdvGAN) [21], perturbation-natural [37], image-agnostic [38], noise-sparse [39], local structure-matched [40], low-frequency constrained [41], natural scenario-embedded [42], target pattern-injected [43], random optical-flows [44] and gradient-editing [45] were proposed to use generative models to produce adversarial perturbations, due to their high efficiency and low requirements for the target model and label cues.

Among them, AdvGAN++ by Jandial et al. [46] further demonstrates that latent features can serve as prior information to help a generative network to achieve more perceptually realistic instances and higher attack success rates. These regularization perturbations conducted in the feature space primarily fall into two categories: (1) Using cosine similarity loss to enlarge feature difference of clean and adversarial samples. SSAE [1] reduces this loss before the fully connected (FC) layer, while BIA [22] diminishes the cosine similarity between these features at specified layer outputs to achieve a notable attack capability. (2) Using L2 norm loss for feature perturbation regularization, where LTAP [23] and ADA [24] achieve powerful attack capabilities by enlarging the L2 norm distance between the features of the benign and post-attack samples. Additionally, Aich et al. [25] combine L2 norm with local patch difference to execute potent attacks in different feature spaces, yielding appealing results in multi-object scenes.

However, current generative attacks that rely on these two losses, are likely to result in learning shortcut of feature perturbations, i.e., the feature optimization tends to over-learn the Easy2Attack features while producing undesirable perturbations that are not conducive to attack capacity improvement for the Hard2Attack ones. To this end, we propose to differentiate the updates of the two categories of features to reduce this learning bias.

C. Imperceptible Adversarial Attack

To enhance the imperceptibility of perturbation noises w.r.t. not only network model but also human eyes, two categories of regularization constraints, i.e., the perceptual distance constraint and the spatial constraints, are frequently considered.

1) Perceptual Distance Constraint: In order to make the adversarial perturbations imperceptible, Carlini and Wanger [26] made use of $\mathcal{L}_{\text{inf}}$, $\mathcal{L}_2$ and $\mathcal{L}_0$ norms to constrain these perturbations, generating small magnitude of noises in terms of these metrics. However, recent works have revealed that $\mathcal{L}_p$ norms do not well align with human eye perception. Hence, other perceptual distance metrics have been proposed to enhance this imperceptibility. Muhammed et al. [28] constrained noises in image structure and offered a new constraint paradigm namely SSIM attack. In terms of color space of an image, Zhao et al. [29] imposed perturbations on images by minimizing their magnitudes w.r.t. perceptual color distance.

Semantic similarity attack on high-frequency components (SSAH) [30] crafted perturbations based on the restriction in the low-frequency space. Zhang et al. [50] proposed to perturb high-frequency components that easily affect model decisions but are not easy-to-perceive by human eyes. Gao et al. [42] enhanced the noise imperceptibility in terms of the injected natural scenes.

2) Spatial Position Constraint: Perceptual distance constraints pertaining to human eyes gain success on improving imperceptibility, while perturbing benign examples globally may introduce redundant cues. Therefore, many works resort to perturbation restrictions in local regions, i.e., generally restricting the noises into highly relevant semantic regions or blocks with strong masking characteristics for human eyes. Dong et al. [39] used the distortion map for measuring the invisibility, and accordingly constructed the distortion-aware sparse adversarial attack, namely GreedyFool. Wang et al. [51] proposed content/spacial structure-aware adversarial perturbation. Due to the importance of attention regions to CV

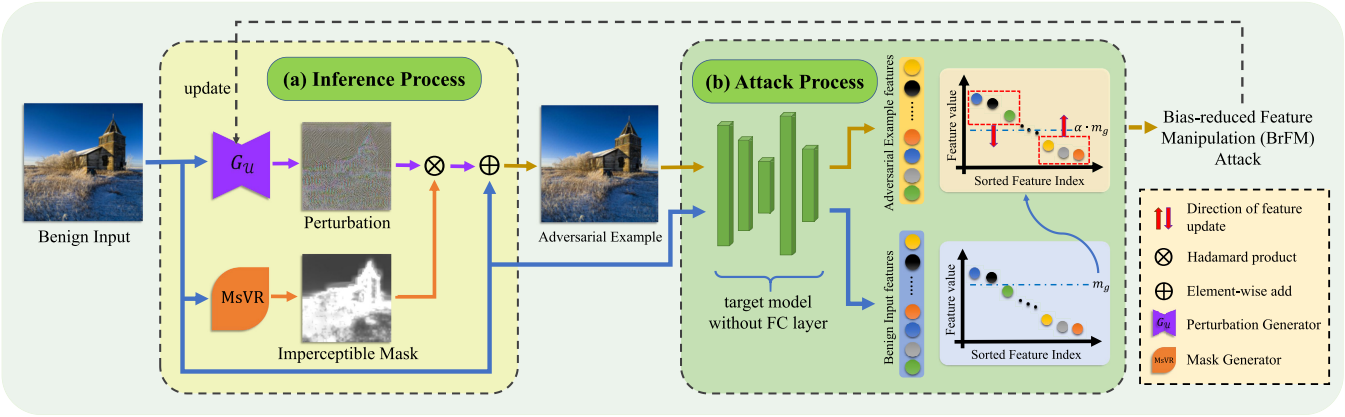


Fig. 2. Overview of the proposed method, including the inference process (in the yellow box), the attack process (in the green box), and the training process (the entire procedure). In (a), 'MsVR' stands for 'multi-scale variance regularization'. In (b), the features during the attack process are rearranged according to feature indexes of the benign sample sorted in a descending order, and separated into two categories based on a threshold, where the Hard2Attack features marked in the red dashed box are then adaptively perturbed.

tasks, attention-based constraints are frequently considered. Dong et al. [52] proposed the superpixel-guided attention attack, to make perturbations distribute on only the salient regions. Coincidentally, Zhang et al. [53] designed a salient-region-based perturbation regularization to restrict noises to task-sensitive regions. Chen et al. [54] proposed to attack on attention (AoA) by perturbing the attention heat map, and Lu et al. [1] used an auto-encoder to generate a salient mask for the perturbations.

Despite achieving improved performances, these loss-driven position constraints did not actually align with the human eye perception. Motivated from the human eye's heightened sensitivity to pixel noises in low variance blocks [32], Luo et al. [33] generated a weight matrix to represent the noise tolerance of human eyes across different regions by quantifying the variance of each region block, and tended to add perturbations in areas of relatively high variance. Despite this approach is straightforward and intuitive, the significance variance map specific to fixed-size region blocks may result in clearly visible edges of the perturbation distribution. To this end, we propose a variance-based weight mask via a multi-scale-driven region partition, and use it as a soft regularization constraint to enhance the human eye-aware imperceptibility while maintaining a reasonable attack capacity.

III. The Proposed Approach

A. Overview and Notation

1) Overview: As shown in Fig. 2, our method consists of two parts. (1) To avoid the learning shortcut of feature perturbations or biased learning toward different difficulties of features, we differentiate the features into Hard2Attack and Easy2Attack ones and update them separately. Specifically, we proposed the Bias-reduced Feature Manipulation (BrFM) to disrupt the output of benign samples in the feature representation layer of a proxy model, i.e., making perturbated features opposite to those of benign samples in the activation intensities. (2) To reduce the crafted noises distributed on easy-to-perceive object boundaries, we introduce the invisibility constraint of Multi-scale Variance Regularization (MsVR) into the training to produce attack perturbations that conform to this human-perceived imperceptible mask. Specifically, we acquire the variances of partitioned regions using multiple window sizes to generate contrast saliency maps at different scales, crafting a soft regularization with a weight mask that highlights regions imperceptible to human eyes.

2) Notation: Before going into the details of the proposed method, we first define some notations based on a common deep classification network (*e.g.,* ResNet [55]).

Assume that $x \in \mathcal{X}^{C\times H\times W}$ is a benign image and y is its ground-truth label, we denote a classifier $f_l : x \longrightarrow f_l(x)$ that outputs the feature map $f_l(x) \in \mathcal{R}^N$ of the l-th layer with the input of x, where N is the feature dimension. Our goal is to train a generator $G_{\mathcal{U}}$ that produces adversarial perturbations $\mathcal{P} \in \mathbb{R}^{C\times H\times W}$, and then add noises to the benign image x to generate the adversarial example $x' \in \mathcal{X}_{adv}^{C\times H\times W}$ that could mislead the classifier f_l to produce wrong classification. This process can be formulated as:

$$\mathcal{P} = G_{\mathcal{U}}(x) \tag{1}$$

where $G_{\mathcal{U}}$ represents the U-net [56] and the generated perturbation $\mathcal{P}$ is added to the original image x as:

$$\begin{cases} x' = x + \text{clamp}(\mathcal{P}, -\delta, \delta) \otimes M(x) \\ f(x') \neq y \end{cases} \tag{2}$$

where $\text{clamp}(\mathcal{P}, -\delta, \delta) = \min(\max(\mathcal{P}, -\delta), \delta)$. The function $M(\cdot)$ is used to generate a mask that limits the strength of perturbations in terms of a soft constraint or weighs the pixel-wise perturbations globally. The symbol $\otimes$ represents the element-wise multiplication.

B. Bias-Reduced Feature Manipulation (BrFM)

1) Motivation: For regularizing the strengths and directions of adversarial perturbations, common attack losses of feature-level L_2 loss [23], [24], [25] and cosine similarity loss [1], [22] are frequently used in generative attacks, whose attack

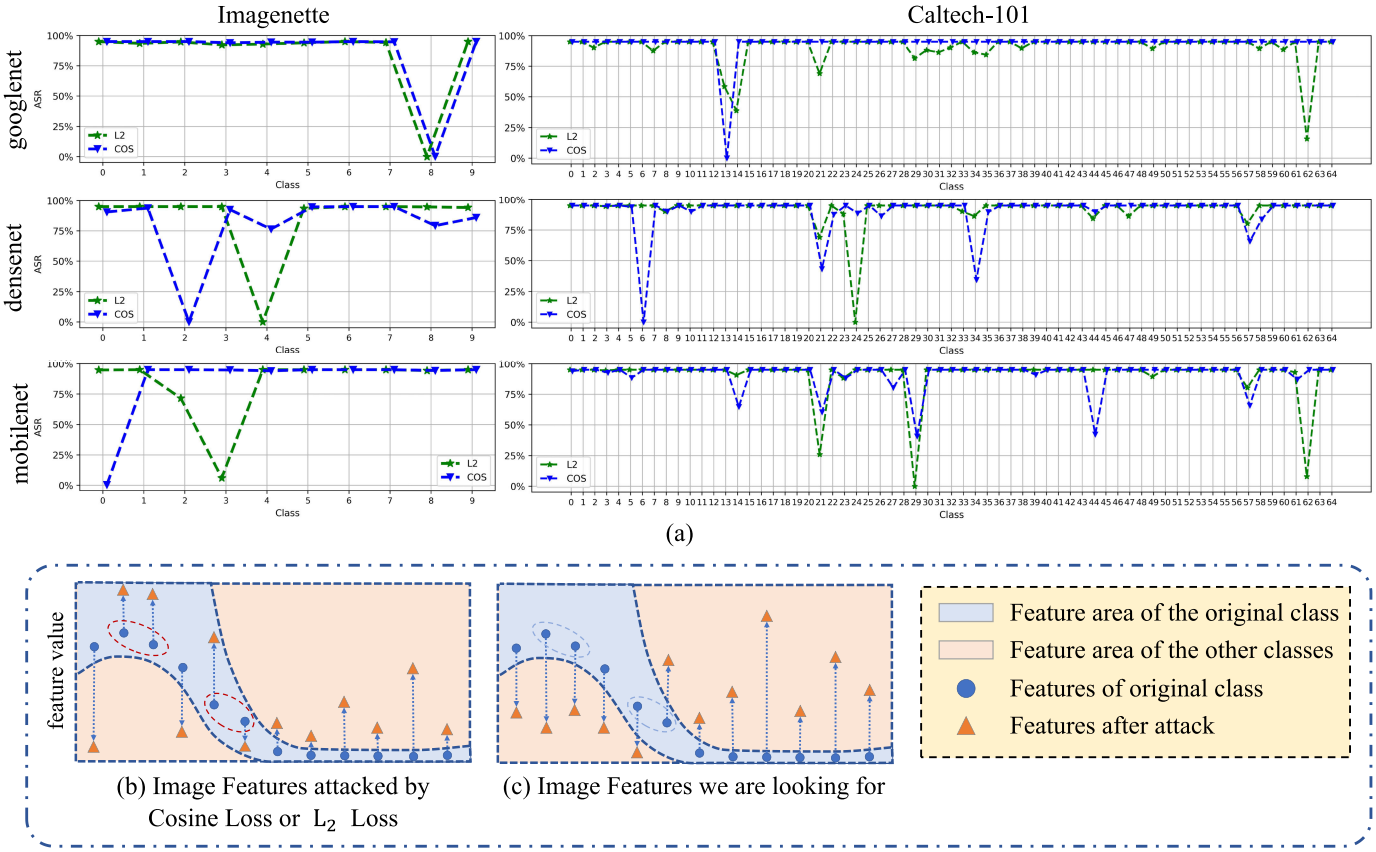


Fig. 3. The motivation of our feature perturbation loss. (a) Visualization of attack success rates (ASR) of three surrogate models (GoogLeNet [2], DenseNet [47], MobileNet [48]) with feature perturbation losses of L2-norm distance (L2) and cosine similarity (Cos) for different classes of two datasets (Imagenette [3], Caltech-101 [49]). For a clear and intuitive visualisation, we have chosen to show only those categories in Caltech-101 whose number of instances is greater than 10. (a) The weaknesses of previous feature regularization losses on some hard-to-succeed classes, (b) the reason for their failures in producing perturbations at the feature level, and (c) the perturbations we expect with feature-based attacks.

principles are presented as follows:

$$\begin{cases} \mathcal{L}_{L_2} = -\left\| f_l(x') - f_l(x) \right\|_2 \\ \mathcal{L}_{cos} = \dfrac{< f_l(x'), f_l(x) >}{\| f_l(x') \|_2 \cdot \| f_l(x) \|_2} \end{cases} \tag{3}$$

where x' denotes the adversarial sample, $< \cdot, \cdot >$ denotes the dot product of two vectors.

When the above two losses are used, as shown in Fig. 3(a), almost all the samples of the failed attacks are concentrated within some of the categories, rather than evenly distributed across all the categories. For the reasons, no matter the 'easy' class which can be easily successfully attacked, or the 'difficult' class whose feature representation is difficult to be correctly attacked, a feature perturbation in an arbitrary direction will produce the same decrease of a L2 or cosine similarity-alike loss. Consequently, an attack system may take a shortcut and is dedicated to mainly perturbing the 'easy' features to favor the feature loss decrease, while producing perturbations of Hard2Attack features that are not helpful to attack capacity, which consequently leads to the non-uniformity of the attack effect in Fig. 3(a). This possible feature perturbation is visualized in Fig. 3(b). Therefore, we expect a loss function that produces a similar effect as Fig. 3(c), i.e., treating all classes as fairly as possible for the adversarial perturbations.

To this end, we propose a Bias-reduced Feature Manipulation (BrFM) to adapt to features with different difficulties, i.e., categorizing features into Hard2Attack and Easy2Attack ones and updating them separately, so as to avoid the learning shortcut on easy but not critical features.

2) Methodology: Different from the feature attack method [57] and the distribution-aligned adversarial attack [40], our BrFM explicitly specifies the update scheme for the Hard2Attack features to avoid the learning shortcut in the previous feature attack algorithms.

Specifically, we divide the features into two categories based on a threshold determined by the distribution of the original feature values. Our bias-reduced feature perturbation loss is formulated as:

$$\mathcal{L}_{BrFM} = \sum_{i=1}^{N} f_l(x')_i \cdot sign(f_l(x)_i > \alpha \cdot m_g) \tag{4}$$

where $sign(\cdot)$ is the sign function outputting 1 or -1. By minimizing Eq. (4), it manipulates the features of the adversarial sample to weaken those behave similarly to the features of benign sample and enlarge those behave oppositely, so as to mislead the classification model. α is a hyperparameter, controlling the strength of regularization. m_g is defined as the average of the features with large activation values, which is

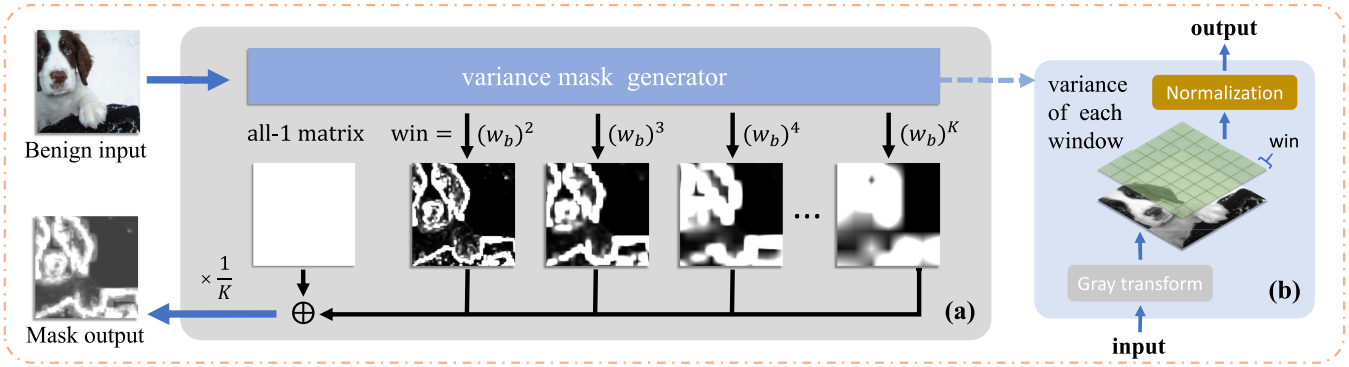
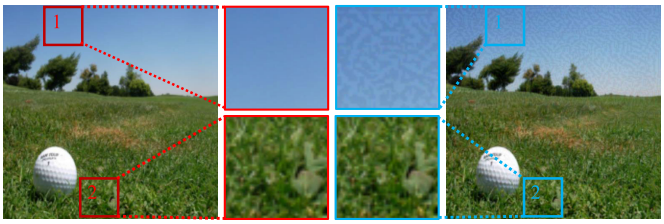


Fig. 4. The process of our imperceptible mask generator module (a) and the principle of the variance mask generation (b), $\oplus$ represents the element-wise addition.



(a) clean image (b) adv image

Fig. 5. The hiding effect of perturbations in high-contrast (labelled by '2') and low-contrast (labelled by '1') regions for (a) benign and (b) adversarial images. Contrast size refers to the degree of variation in terms of pixel values within a considered area.

formulated as:

$$m_g = \frac{1}{sum(I)} \sum_{i=1}^{N} f_l(x)_i \cdot 1_{f_l(x)_i > mean(f_l(x))} \tag{5}$$

where $I \in \{0,1\}^N$ is a binary mask, and $I_i = 1$ if $f_l(x)_i > mean(f_l(x))$, otherwise, $I_i = 0$. $sum(\cdot)$ denotes the summation, and $mean(\cdot)$ means averaging.

Via Eq. (4), a great penalty term is produced when a feature component of the adversarial sample x', i.e., $f_l(x')_i$ has similar activation value as that of the benign sample x, i.e., $f_l(x)_i$. By minimizing the loss in Eq. (4), the cases that $f_l(x')_i$ and $f_l(x)_i$ behave similarly (both large or small) are discouraged, while the cases that $f_l(x')_i$ and $f_l(x)_i$ behave oppositely are encouraged.

To obtain the threshold $\alpha \cdot m_g$ for feature separation, the average feature value, namely $mean(f_l(x))$, is obtained from the benign sample. Next, the average of the features larger than $mean(f_l(x))$, denoted as m_g, is scaled with a factor α to set the threshold. This threshold, derived from the benign sample, is used to distinguish the features of adversarial samples.

C. Multi-Scale Variance Regularization (MsVR)

1) Motivation: Fig. 5 shows the differences between the imposed noises in the regions of high-contrast and low contrast, where the perturbations in the low-contrast region are easily perceived, while those in the high-contrast region are relatively imperceptible. However, current regularization methods treating all pixels equally ignore the diverse perceptibility of human eyes to perturbations in regions with different contrasts. Considering the merit of variance [33] that can measure the perceptibility of perturbation noises aligning with human eye, we use it to evaluate the contrast size of each region containing perturbations and regularize them to be more imperceptive. For this purpose, a partition of an image into different regions is first crafted.

However, the contrast sizes of neighboring regions may differ much, and a single partition of regions may make the perturbation noises distributed on region boundaries quite perceptible, leading to poor imperceptibility w.r.t. human eyes. Meanwhile, the window size is difficult to determine. Too small partition regions may make the variance saliency maps contain too few attackable regions, and too large regions may make perturbations easily fall in the low-contrast regions. Thus, we resort to multiple window sizes to obtain variance saliency maps at different scales, so as to reduce unnatural transitions of perturbations in mask edges, as well as trade off the number of attackable regions and the human-perception imperceptibility, to maintain a reasonable attack capacity.

2) Methodology: Fig. 4 presents the generation frame of our imperceptibility-aware mask, which is constructed based on multi-scale sub-variance saliency maps, each of which is specific to a window size. Specifically, we first convert a RGB image x to the grayscale image x^g via the gray transform, i.e., the grayscale image $x^g \in \mathbb{R}^{H \times W}$ is obtained by averaging the three channels of x as $x^g = \frac{1}{C}\sum_{i=0}^{C-1} x_i$, where $x \in \mathbb{R}^{C \times H \times W}$ and x_i denotes the $i+1$-th channel of x and C denotes the number of channels of x.

To partition the image x, we divide the grayscale x^g into $k \times m$ parts, i.e., $x^g = \{w_{i,j}, 1 \le i \le k, 1 \le j \le m\}$ according to the given window size win, where $k = \text{int}(\frac{H}{win})$, $m = \text{int}(\frac{W}{win})$. Each w_{ij} ($i \in [0,k]$ and $j \in [0,m]$) contains about win^2 pixels.

Based on this partition, we compute the variance for the pixels contained in each window, resulting in the corresponding variance matrix $M_{sub}(x^g, win) = [var(w_{i,j})], \{1 \le i \le k, 1 \le j \le m\}$, where $var(w_{i,j})$ represents the variance specific to $w_{i,j}$. In terms of these sub-variance saliency maps, we derive the representation of the final variance mask as:

$$M(x) = \frac{1}{K}(1 + \sum_{ns=2}^{K} M_{sub}(x, (w_b)^{ns})) \tag{6}$$

where K is the number of scales (containing 'all-1 matrix' scale), w_b denotes the base window size, which is used to

obtain the used window sizes by incremental exponential operations. To make this mask regularization a soft constraint, a value of $1/K$ is further added to the average mask for each pixel to ensure a global allowable attack space.

By using different window sizes via Eq. (6), we can represent imperceptible regions at different scales in terms of the weight mask $M(\cdot)$, imposing a soft mask constraint for the allowable perturbation noises. In this way, the susceptibility of the perturbation noises distributed on the boundaries of partition regions can be well relieved, and the number of attackable regions and the imperceptibility can be better balanced. Furthermore, besides of the improved imperceptibility of the perturbation noises, the proposed soft mask constraint further ensures that their attack capacity is not severely impaired.

For clarity, we present the entire training of our attack algorithm in Algo. 1.

Algorithm 1 The Training of Our Algorithm

Input: number of iterations N; perturbation budget δ; a minibatch of original images $\mathcal{X}^{B\times C\times H\times W}$; the trained classifier f_l; the perturbation generator $G_{\mathcal{U}}$; learning rate η;

Output: The trained perturbation generator $G_{\mathcal{U}}$;

Initialize the parameters $\theta_{\mathcal{U}}^0$ of $G_{\mathcal{U}}$;
while $i \leftarrow 0$ to $N-1$ **do**
  For $\boldsymbol{x} \in \mathcal{X}$;
  Generate adversarial perturbation $\mathcal{P}$ by Eq. (1);
  Generate imperceptible mask $M(\boldsymbol{x})$ by Eq. (6);
  Construct current adversarial example $\boldsymbol{x}'$ by Eq. (2);
  Obtain the feature vectors of $\boldsymbol{x}$ and $\boldsymbol{x}'$ with the target classification model f_l: $f_l(x)$ and $f_l(x')$;
  Get the attack loss by $\mathcal{L}_{BrFM} \leftarrow$ Eq. (4);
  Back propagation $\nabla\mathcal{L}_{BrFM} \leftarrow Backward(\mathcal{L}_{BrFM}, \theta_{\mathcal{U}}^i)$;
  Update $\mathcal{G}_{\mathcal{U}}$'s parameter $\theta_{\mathcal{U}}^{i+1} \leftarrow \theta_{\mathcal{U}}^i - \eta\cdot\nabla\mathcal{L}_{BrFM}$;
end while

IV. Experiment

In this section, we first present the experimental settings in section IV-A, and then conduct empirical evaluation on two datasets where our algorithm is compared with the state-of-the-art imperceptible attacks on multiple models in section IV-B. Furthermore, we carry out the ablation experiments to study the function of each module in section IV-E. Finally, we provide some explanatory analysis of the proposed modules in section IV-F.

A. Experimental Setup

1) Dataset: Similar to [1], we use the datasets of Imagenette [3] and Caltech-101 [49] for the evaluation. In particular, Caltech-101 comprises of approximately 9,000 images, grouped into a total of 101 categories, with each category containing 40 to 800 samples. Since the size of the samples is not fixed, we resize them to 224×224 for training and testing. As it lacks preassigned training and testing sets, we randomly allocated 20% of the images from each category as the testing set. Imagenette is a challenging dataset sampled from a large and public dataset ImageNet [58], it contains 10 classes (`tench, English springer, cassette player, chain saw, church, etc.`) and the size of each sample is 224×224. Meanwhile, all the test images can be almost correctly classified by the target models to ensure a reasonable experimental performance.

2) Target Models: For the target models, we cover five normally trained classification models, i.e., DenseNet [47], EfficientNet [59], MobileNet-V2 [48], ResNet [55] and GoogLeNet [2]. They are trained on Imagenette and Caltech-101 separately, and the clean accuracy is shown in Table III. We evaluate the attack success rate (ASR) using the clean accuracy, i.e., ACC_{clean} in Table III and the accuracy after performing an attack, i.e., ACC_{adv} as:

$$ASR = \frac{ACC_{clean} - ACC_{adv}}{ACC_{clean}} \tag{7}$$

3) Methods for Comparison: In studying the effectiveness of our method, we adopt six generative attack methods for comparison, i.e., AdvGAN [21], SSAE [1], ADA [24], LIGAA [41], GCMA [44], GE-AdvGAN [45] and a classic gradient-based attack method FGSM [5].

4) Evaluation Metric: For the algorithm evaluation, we quantify the perceptual quality of the adversarial examples with various metrics, including ASR as the metric of the attack capacity, and the measures of the imperceptibility of conventional average $\mathcal{L}_2$ norm, maximum perturbation intensity ($\mathcal{L}_{inf}$) [26], structural similarity (SSIM) [28], [60], peak signal-to-noise ratio (PSNR) and average distortion of low-frequency components (LF) [30]. We also visualize adversarial examples to qualitatively evaluate the adversarial perturbations and their imperceptibility.

5) Hyper-Parameters Setting: We follow the attack setting in [1] with the maximum perturbation of $\delta = 0.1$ and set pixel value in [0, 1]. To train the generative attack methods, we use the Adam optimizer with a learning rate of $1e^{-4}$, a batch size of 16 and a number of training epochs $T = 60$ in all experiments. For our algorithm, we adopt the scale strength of $\alpha = 0.7$, $w_b = 2$ and $K = 4$ ($win \in \{4, 8, 16\}$). For all images, we set $C = 3$, $H = 224$, and $W = 224$. For LIGAA, we reused most of the codes from [41] to implement perceptual color distance loss and the C&W attack, while the remaining parts were coded by us.

6) Selection of Visualization Images: All displayed images are randomly selected from those with clear object edges, to facilitate observing the transition effects of the perturbations between different regions constrained by our MsVR module.

B. Attack Strength and Imperceptibility

In this section, we investigate the adversarial attack strength and invisibility of perturbations generated by different methods across various scenarios.

1) Adversarial Strength Comparison: We compared our method with seven related works in Table I and Table II. FGSM is a gradient-based optimization method, while the rest are generation-based attack ones. While FGSM requires

TABLE I

RESULTS OF GENERATIVE SEMI-WHITEBOX ATTACKS ON IMAGENETTE. THE BEST AND SECOND BEST RESULTS ARE LABELED IN **BOLD** AND UNDERLINE, RESPECTIVELY

Dataset	Target	Method	ASR↑	L_∞	SSIM↑	PSNR↑	LF↓	L_2 ↓
Imagenette	ResNet-18	**FGSM** ICLR'15 [5]	89.31	0.1	0.379	21.58	23.80	38.80
		AdvGAN IJCAI'18 [21]	87.88	0.1	0.719	21.91	22.63	37.38
		SSAE ACM MM'21 [1]	98.35	0.1	0.749	22.57	21.67	34.71
		ADA ICIP'22 [24]	88.39	0.1	0.736	22.49	22.83	34.95
		LIGAA CS'23 [41]	96.34	0.1	0.793	23.37	18.74	32.67
		GCMA MM'23 [44]	88.48	0.1	0.724	22.12	22.96	36.47
		GE-AdvGAN SDM'24 [45]	88.80	0.1	0.718	21.98	21.43	37.06
		Ours	**99.11**(↑ 0.76)	0.1	**0.883**	**25.13**	**16.17**	**25.96**
	EfficientNet	FGSM	89.99	0.1	0.378	21.58	23.99	38.79
		AdvGAN	88.12	0.1	0.712	21.75	25.37	38.07
		SSAE	91.98	0.1	0.741	22.28	23.04	36.86
		ADA	74.59	0.1	0.712	21.97	23.99	37.11
		LIGAA	94.38	0.1	0.812	24.22	17.60	29.00
		GCMA	93.93	0.1	0.719	22.11	23.61	36.55
		GE-AdvGAN	88.51	0.1	0.719	21.89	24.01	37.44
		Ours	**99.27**(↑ 4.89)	0.1	**0.881**	**25.12**	**16.84**	**26.00**
	GoogLeNet	FGSM	85.57	0.1	0.380	21.58	22.84	38.80
		AdvGAN	80.20	0.1	0.723	21.75	24.67	38.05
		SSAE	88.00	0.1	0.763	23.20	20.22	32.31
		ADA	85.68	0.1	0.738	22.78	20.99	33.84
		LIGAA	89.17	0.1	0.793	23.73	21.49	30.72
		GCMA	87.01	0.1	0.734	22.62	22.29	34.46
		GE-AdvGAN	74.43	0.1	0.730	21.95	23.10	37.18
		Ours	**94.51**(↑ 5.34)	0.1	**0.892**	**25.44**	**15.95**	**25.08**
	MobileNet	FGSM	89.50	0.1	0.377	21.58	21.74	38.80
		AdvGAN	87.46	0.1	0.709	21.82	13.09	37.78
		SSAE	90.53	0.1	0.737	22.46	14.66	35.14
		ADA	87.65	0.1	0.706	21.89	14.01	37.44
		LIGAA	94.26	0.1	0.807	23.37	**7.60**	32.17
		GCMA	89.36	0.1	0.713	22.01	15.77	36.99
		GE-AdvGAN	88.76	0.1	0.720	22.07	19.74	36.70
		Ours	**99.65**(↑ 5.39)	0.1	**0.888**	**25.26**	14.82	**25.58**
	DenseNet	FGSM	82.75	0.1	0.400	21.58	21.74	38.80
		AdvGAN	86.75	0.1	0.729	21.95	26.58	37.19
		SSAE	85.96	0.1	0.730	22.02	26.81	36.94
		ADA	85.48	0.1	0.718	22.12	27.85	36.46
		LIGAA	89.63	0.1	0.791	23.57	20.19	31.22
		GCMA	63.43	0.1	0.710	21.59	22.47	38.79
		GE-AdvGAN	90.37	0.1	0.732	21.88	26.60	37.50
		Ours	**96.90**(↑ 6.53)	0.1	**0.890**	**25.25**	**19.25**	**25.62**

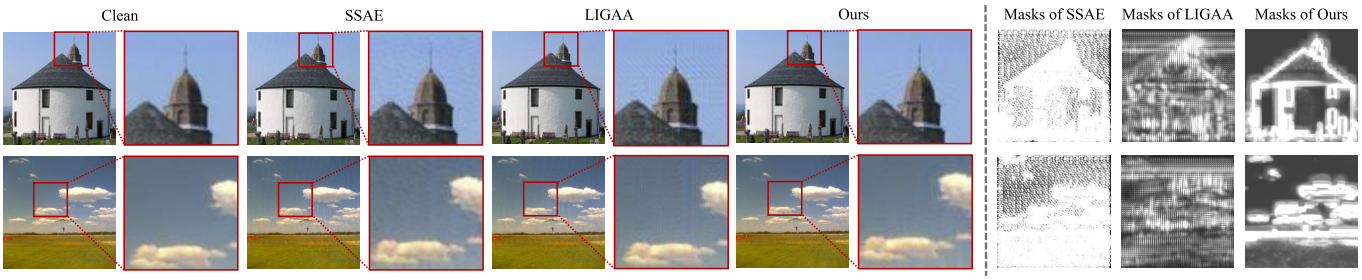


Fig. 6. Adversarial samples as well as imperceptibility masks generated by the methods of SSAE, LIGAA and ours.

complete access to all cues about the target model, AdvGAN, SSAE, ADA, LIGAA, GCMA, GE-AdvGAN and our method require access to the target model during only the training phase. In addition, SSAE, GCMA and our method do not require the real labels during the training phase.

Tables I and II show that our method achieves the highest ASR on both datasets across all models. Taking the Imagenette dataset as an example, our method achieves the optimal results in terms of ASR for all models. Besides the ResNet model, our method achieves a 4% ∼ 7% improvement based on the other seven models. It's worth noting that our method did not use the ground truth labels in achieving these large improvements.

2) Visualization Performance Comparison: Since SSAE and LIGAA are also capable of generating imperceptibility-aware masks, we visually compare the outputs of these methods with ours in Fig. 6. LIGAA uses distance constraints in colour space, thus it has better imperceptibility compared to SSAE as well as faster convergence of mask generation during training. Compared with LIGAA, our approach possesses stronger imperceptibility with the generated masks via MsVR,

TABLE II

RESULTS OF GENERATIVE SEMI-WHITEBOX ATTACK ON CALTECH101. THE BEST AND SECOND BEST RESULTS ARE LABELED IN **BOLD** AND UNDERLINE, RESPECTIVELY

Dataset	Target	Method	ASR↑	L_∞	SSIM↑	PSNR↑	LF↓	L_2 ↓
Caltech-101	ResNet-18	**FGSM** ICLR'15 [5]	88.04	0.1	0.653	21.58	22.92	38.80
		AdvGAN IJCAI'18 [21]	98.45	0.1	0.669	22.31	22.05	35.67
		SSAE ACM MM'21 [1]	98.45	0.1	0.684	22.36	21.93	35.53
		ADA ICIP'22 [24]	98.79	0.1	0.661	22.21	23.80	36.11
		LIGAA CS'23 [41]	98.12	0.1	0.731	23.31	17.66	32.46
		GCMA MM'23 [44]	98.59	0.1	0.663	22.17	21.21	36.27
		GE-AdvGAN SDM'24 [45]	98.52	0.1	0.663	21.93	22.91	37.30
		Ours	**99.26**(↑ 0.47)	0.1	**0.857**	**25.27**	**16.25**	**25.52**
	EfficientNet	FGSM	95.38	0.1	0.653	21.58	24.39	38.80
		AdvGAN	93.74	0.1	0.659	21.78	26.24	37.93
		SSAE	95.31	0.1	0.665	21.90	24.91	37.42
		ADA	95.80	0.1	0.657	21.82	25.80	37.76
		LIGAA	95.73	0.1	0.752	22.93	22.35	33.54
		GCMA	94.45	0.1	0.671	22.15	24.82	36.33
		GE-AdvGAN	94.74	0.1	0.663	21.97	25.63	37.11
		Ours	**96.23**(↑ 0.43)	0.1	**0.853**	**25.12**	**16.70**	**25.97**
	GoogLeNet	FGSM	88.47	0.1	0.653	21.58	20.51	38.80
		AdvGAN	94.52	0.1	0.706	22.61	20.80	34.48
		SSAE	97.62	0.1	0.708	22.80	21.37	33.77
		ADA	93.80	0.1	0.672	22.62	21.82	34.44
		LIGAA	96.83	0.1	0.769	23.28	17.61	32.33
		GCMA	93.43	0.1	0.689	22.33	20.09	35.63
		GE-AdvGAN	90.78	0.1	0.667	22.02	22.88	36.88
		Ours	**97.69**(↑ 0.07)	0.1	**0.873**	**25.61**	**15.01**	**24.56**
	MobileNet	FGSM	87.20	0.1	0.652	21.58	24.16	38.80
		AdvGAN	91.64	0.1	0.656	21.67	27.50	38.42
		SSAE	90.06	0.1	0.667	22.01	22.97	36.95
		ADA	79.74	0.1	0.652	21.79	25.64	37.88
		LIGAA	90.59	0.1	0.811	23.78	**13.76**	30.40
		GCMA	91.11	0.1	0.667	22.05	23.03	36.77
		GE-AdvGAN	92.85	0.1	0.668	22.04	25.91	36.83
		Ours	**97.52**(↑ 4.67)	0.1	**0.859**	**25.19**	17.60	**25.77**
	DenseNet	FGSM	88.37	0.1	0.663	21.58	23.91	38.80
		AdvGAN	84.58	0.1	0.673	21.70	17.98	38.30
		SSAE	90.16	0.1	0.671	22.07	22.69	36.70
		ADA	93.74	0.1	0.668	22.07	21.10	36.70
		LIGAA	92.09	0.1	0.756	23.80	19.53	30.58
		GCMA	94.77	0.1	0.680	22.13	23.22	36.45
		GE-AdvGAN	67.52	0.1	0.692	21.88	20.06	37.49
		Ours	**95.39**(↑ 0.62)	0.1	**0.869**	**25.29**	**16.85**	**25.45**

TABLE III

CLEAN ACCURACY OF TARGET MODELS ON THE DATASETS OF CALTECH-101 AND IMAGENETTE

Datasets	ResNet	EfficientNet	GoogLeNet	MobileNet-V2	DenseNet
Caltech-101	84.1	79.4	78.4	75.0	82.1
Imagenette	94.0	91.1	89.1	93.7	86.3

since it obtains a weight mask that is free of a training process, and will not compromise the imperceptibility when striving for a better attack capacity. The adversarial samples generated by our method in Fig. 6(d) show that, few perturbations appear on the low-contrast background, except for the region where the main object is located. However, for the adversarial samples generated by SSAE and LIGAA, obvious perturbation textures can be easily perceived in the background. This demonstrates that our method can hide perturbations within high-contrast regions where the main objects are located.

As shown in Fig. 6(e), we also compared the imperceptibility-aware masks generated by SSAE, LIGAA and our method. It shows that our method can better guide the network to learn the distribution of imperceptible perturbations. For the reasons, our mask generation is completely separated from the attack process, requiring only the computation of the local variance of an image. By contrast, SSAE and LIGAA use globally constrained regularization terms to train a mask decoder, which may not accurately learn the imperceptible regions when striking a balance between the quality of mask generation and the attack capability.

C. Defense Robustness and Transferability Experiments

To study the robustness of generative adversarial attacks, we evaluated their performances under defense mechanisms and averaged the ASR across all models, and present the results in Tab. IV. It shows that our method achieves a competitive attack effect, i.e. at least the top 2 performance, under a defense or purification, compared to other generative attack methods (excluding non-generative FGSM). Besides, it shows that low-cost JPEG [61] compression and decompression may be promising in defending against generative adversarial attacks, and the advanced DISCO [64] method also stands out as an effective defense measure for these attacks.

TABLE IV

ASR RESULTS AVERAGED OVER 5 TARGET MODELS OF GENERATIVE SEMI-WHITEBOX ATTACKS WITH DIFFERENT DEFENSE METHODS ON IMAGENETTE AND CALTECH-101. THE BEST AND SECOND BEST RESULTS ARE LABELED IN **BOLD** AND UNDERLINE, RESPECTIVELY

Dataset	Method	JPEG[61]	Bit_Red[62]	NRP[63]	DISCO[64]
Imagenette	FGSM	70.59	85.66	54.63	7.81
	AdvGAN	28.92	83.17	34.26	18.62
	SSAE	25.72	87.30	32.58	12.88
	ADA	26.10	78.66	34.99	10.62
	LIGAA	14.26	74.99	29.17	11.29
	GCMA	27.54	82.06	**36.16**	20.47
	GE-AdvGAN	36.93	84.68	32.43	28.10
	Ours	**42.18**	**96.95**	35.05	**28.27**
Caltech-101	FGSM	74.95	88.54	67.73	11.67
	AdvGAN	40.99	88.33	18.31	8.27
	SSAE	36.15	89.11	21.69	6.29
	ADA	36.44	87.82	25.62	5.88
	LIGAA	13.53	63.73	13.06	5.93
	GCMA	41.37	89.74	20.38	9.11
	GE-AdvGAN	**52.92**	84.86	21.98	12.26
	Ours	43.10	**96.76**	**32.99**	**15.73**

TABLE V

ASR RESULTS OF GENERATIVE SEMI-WHITEBOX ATTACKS BY THE GENERATIVE MODEL TRAINED ON IMAGENETTE AND TRANSFERRED TO THE CALTECH-101. THE BEST AND SECOND BEST RESULTS ARE LABELED IN **BOLD** AND UNDERLINE, RESPECTIVELY

Method	ResNet-18	EfficientNet	MobileNet	Avg
AdvGAN	5.11	2.63	0	2.58
SSAE	5.58	3.20	0	2.93
ADA	6.65	2.28	0.98	3.30
LIGAA	5.17	1.64	0	2.27
GCMA	5.58	**4.41**	0.53	3.51
GE-AdvGAN	4.77	2.28	1.28	2.78
Ours	**10.69**	3.37	**3.31**	**5.79**

To study whether cross-dataset feature interference exists when using different datasets, we train the training set of Imagenette, and evaluate the ASR on the test set of Caltech-101, and show the results in Table V. It shows that our method still performs optimally on ResNet, EfficientNet, and MobileNet, although the primary goal of our attack was not for cross-dataset or cross-distribution transfer. Despite the relatively lower ASR compared with that on the same dataset, it still indicates that the surrogate model can represent features with certain similarities for datasets with different distributions.

D. Performance With ViT Models

To study the performance of our method with Vision Transformer (ViT) models, we conducted evaluations on ViT-Base [65] and showed the results in Tab. VI. While all generative models struggle to train properly on ViT, with the training loss hardly decreasing at all, Tab. VI shows that our approach still achieves at least the top 2 ASR with competitive imperceptibility.

To shed light on the perturbations specific to the ViT-based models, we visualize them in Fig. 7. It shows that almost all methods produced abnormal perturbations with large continuous areas of activation values.

TABLE VI

RESULTS OF GENERATIVE SEMI-WHITEBOX ATTACK ON IMAGENETTE AND CALTECH-101 USING VIT-BASE AS THE TARGET MODEL. THE BEST AND SECOND BEST RESULTS ARE LABELED IN **BOLD** AND UNDERLINE, RESPECTIVELY. $L_\infty = 0.1$

Dataset	Method	ASR ↑	SSIM ↑	PSNR ↑	LF ↓	L_2 ↓
Imagenette	AdvGAN	0.64	0.843	21.71	35.97	38.23
	SSAE	0.79	0.734	21.74	26.42	38.11
	GCMA	0.67	0.815	21.71	30.09	38.27
	GE-AdvGAN	0.21	0.713	21.76	**9.25**	38.01
	Ours	**2.18**	**0.908**	**25.94**	19.91	**23.67**
Caltech-101	AdvGAN	**5.16**	0.683	21.75	26.32	38.05
	SSAE	0.93	0.671	22.01	28.17	36.99
	GCMA	0.35	0.771	23.94	24.44	29.61
	GE-AdvGAN	0.93	0.675	22.03	**17.11**	36.87
	Ours	3.77	**0.885**	**25.40**	19.20	**25.16**

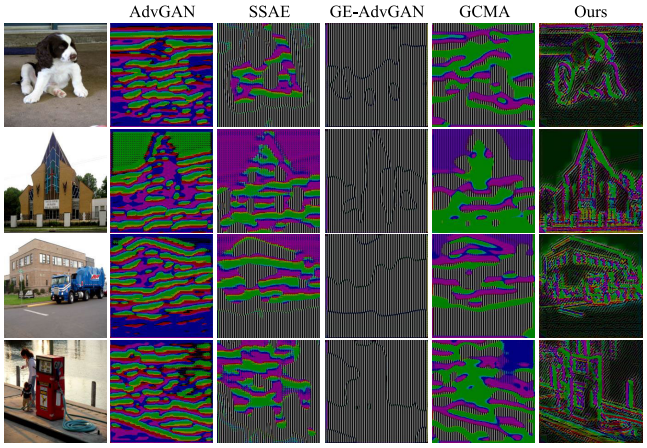


Fig. 7. Visualization of perturbations generated by different methods on Imagenette when using ViT as the target model.

E. Ablation Experiments

1) Description: We further analyze the contributions of the modules in our approach in terms of attack capability and imperceptibility. In Table VII, we study the two main modules of our approach and visualize the results in Fig. 8. For the attack setting of 'BASE', the proposed modules are not used but the ground truth labels of data are used in the cross entropy loss.

To evaluate the effects of different hyperparameters in the proposed modules, we conduct a fine-grained ablation experiment and show the results in Table VIII, where we show the performance of each component for the mask generation, as well as the entire module. GoogLeNet is used as the target model for testing on the Imagenette dataset.

To shed light on the produced masks with different window sizes, we visualize the perturbations and the generated adversarial samples in Fig. 9.

2) Coarse-Grained Ablation Study: Table VII shows that our 'BrFM' attack is able to successfully attack almost all of the test images when the 'MsVR' module is excluded, reaching an ASR of 98.23%, which outperforms 'BASE' by a margin of 10% in terms of ASR, with a comparable imperceptibility degree. When the 'MsVR' module is employed, our method largely outperforms 'BASE' in terms of imperceptibility with four invisibility metrics, even though the

TABLE VII

THE ABLATION STUDY OF THE PROPOSED MODULES. 'BRFM' AND 'MSVR' DENOTE THE BIAS-REDUCED FEATURE MANIPULATION (BRFM) AND THE MULTI-SCALE VARIANCE REGULARIZATION (MSVR), RESPECTIVELY. $L_\infty = 0.1$

Methods			Evaluation Metric				
BASE	BrFM	MsVR	ASR↑	SSIM↑	PSNR↑	LF↓	L_2 ↓
✓	-	-	88.23	0.737	22.61	22.40	34.50
-	✓	-	**98.23**	0.720	22.32	22.11	35.63
✓	-	✓	87.63	0.891	25.35	16.07	25.37
-	✓	✓	94.51	**0.892**	**25.44**	**15.95**	**25.08**

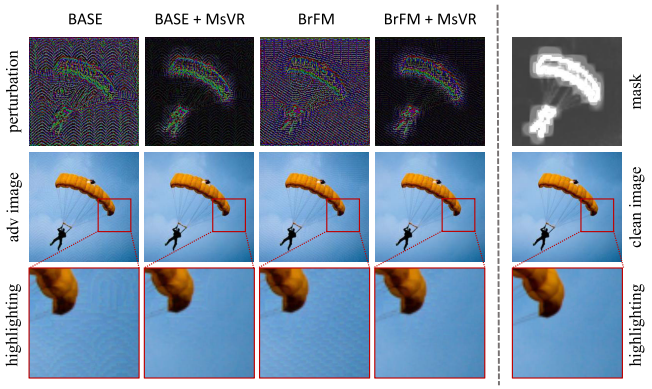


Fig. 8. Visualization of the perturbation noises corresponding to different variants of module combinations in Tab. VII.

TABLE VIII

THE PERFORMANCE SENSITIVITY AGAINST DIFFERENT SETTINGS OF win AND K, WHERE $w_b = 2$, $K = 4$ MEANS THAT THE ACQUIREMENT OF VARIANCE MASKS USES THE WINDOW SIZES OF $win \in \{(w_b)^k, k \in \{2, \ldots, K\}\}$ IN EQ. (6). $L_\infty = 0.1$

Methods					Evaluation Metric				
all-1	*win*=4	*win*=8	*win*=16	*win*=32	ASR ↑	SSIM↑	PSNR↑	LF↓	L_2 ↓
-	✓	-	-	-	56.00	**0.969**	**26.70**	**14.02**	**21.87**
-	-	✓	-	-	76.14	0.945	25.95	15.19	23.74
-	-	-	✓	-	94.00	0.903	25.21	16.33	25.77
-	-	-	-	✓	97.06	0.847	24.41	17.66	28.17
✓	-	-	-	-	**98.23**	0.720	22.32	22.11	35.63
(*K*=2)✓	✓	-	-	-	96.83	0.841	24.78	17.03	26.97
(*K*=3)✓	✓	✓	-	-	94.77	0.884	25.36	16.08	25.30
(*K*=4)✓	✓	✓	✓	-	94.51	0.892	25.44	15.95	25.08
(*K*=5)✓	✓	✓	✓	✓	95.63	0.883	25.32	16.14	25.43

ASR is slightly reduced compared with the variant without this module (decreased by less than 1%). Table VII also reflects that our method attacking the feature layer without using data labels, could outperform the method, i.e., the baseline of 'BASE' using labels with the cross-entropy loss.

3) Fine-Grained Ablation Study: Table VIII shows that the imperceptibility performance is excellent, but the attack capability is relatively weak when $win = w$ ($w \in \{4, 8, 16\}$). Meanwhile, as shown in Fig. 9, when a single window size is used to generate the weight mask, an undesirable perturbation boundary is observed, making it easier to be perceived by human eyes. In contrast, the setting of different window sizes for generating various subvariance significance maps, can produce a mask with much better performance in terms of imperceptibility, i.e., better invisible perception by human eyes.

TABLE IX

RESULTS OF GENERATIVE SEMI-WHITEBOX ATTACKS BY USING THE GRAD-CAM OR OUR MSVR MODULE ON IMAGENETTE

Target	Method	ASR ↑	SSIM ↑	PSNR ↑	LF ↓	L_2 ↓
Grad-CAM	Resnet-18	98.78	0.884	25.68	15.17	24.32
	EfficientNet	99.41	0.820	23.92	18.95	29.70
	GoogLeNet	90.45	0.857	25.62	15.30	24.52
	MobileNet	99.51	0.833	24.37	16.38	28.25
	DenseNet	95.38	0.851	24.48	20.99	27.97
	Average	96.71	0.849	24.81	17.36	26.95
MsVR	Resnet-18	99.11	0.883	25.13	16.17	25.96
	EfficientNet	99.27	0.881	25.12	16.84	26.00
	GoogLeNet	94.51	0.892	25.44	15.96	25.08
	MobileNet	99.65	0.888	25.26	14.82	25.58
	DenseNet	96.90	0.890	25.25	19.25	25.62
	Average	**97.89**	**0.887**	**25.24**	**16.61**	**25.65**

Based on the quantitative results in Table VIII and the qualitative results in Fig. 9, it shows that our mask regularization MsVR largely enhances imperceptibility w.r.t. human eye perception, while maintaining a superior attack capability (compared with the all-1 setting, the ASR achieved by ours decreases by less than 4%, but the achieved SSIM has improved by 20%). For instance, excellent performances in terms of invisibility metrics were achieved with a reasonable ASR of 94.51%, after using the 'BrFM' and 'MsVR' modules. By comparing with the images in the 3rd column ($win = 16$) of Fig. 9, the images in the 8th column ($K = 3$) have stronger imperceptibility, reflecting the advantage of the proposed soft regularization with the smooth global mask obtained by our multi-window variance.

F. Algorithm Analysis

1) Hyperparameter Analysis: In this section, we analyze the sensitivity of ASR performance against the hyperparameter α in Eq. (4), where this parameter is mainly used to categorize the Hard2Attack and Easy2Attack features. For this analysis, we selected three models, i.e., ResNet, GoogLeNet and DenseNet, as the test models to evaluate the hyperparameter sensitivity on the imagenette dataset, and show the results in Fig. 10. It shows that all the three models basically achieve good performances under the setting of $\alpha = 0.7$, which is thus used as the default setting. A larger α will result in too few Hard2Attack features being properly perturbed, while a smaller α will unduly mix more Easy2Attack features with Hard2Attack ones. When this happens, the effect of customized perturbations for Hard2Attack features will be impaired, or the opportunity to implicitly activate Easy2Attack features will be reduced.

2) Visualization in Terms of Grad-CAM: To shed light on the impact of imperceptibility constraints on adversarial perturbations, we use Grad-CAM [66] to visualize the activation maps of both benign and adversarial samples by different methods in Fig. 11. It shows that the perturbations produced by our method essentially cover the target object while minimizing the perturbation area, i.e., achieving a better attack performance with a smaller perturbation area, compared with the approaches of SSAE and LIGAA. It also shows that our method effectively diverts the model's attention away from the target object.

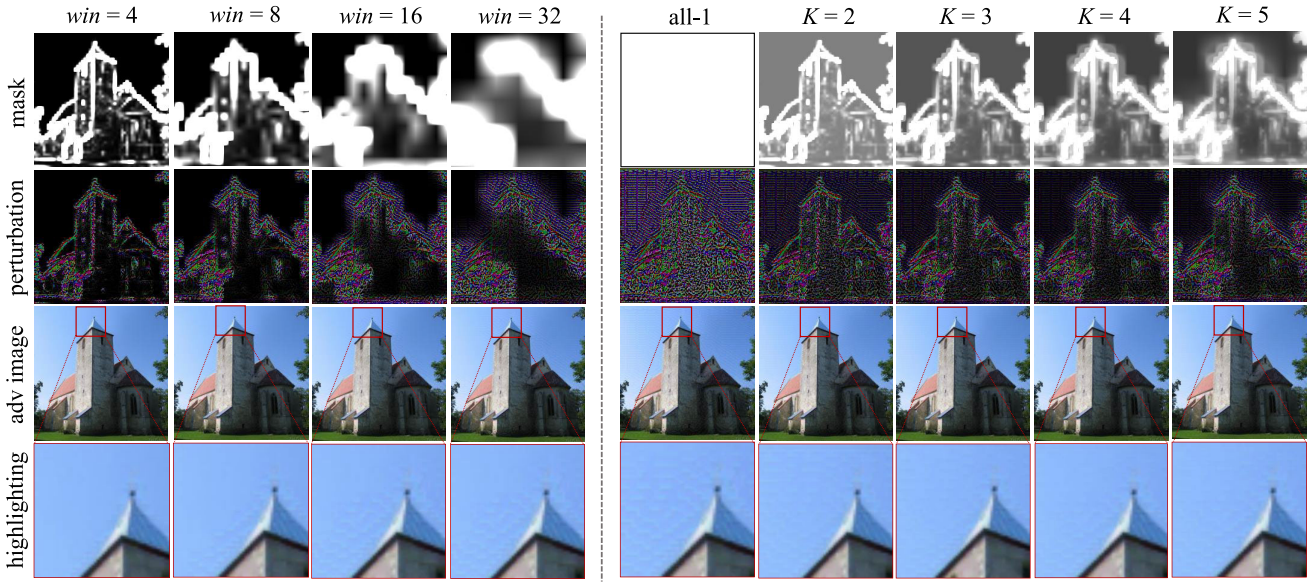


Fig. 9. The masks produced by our module MsVR with different window sizes win and K values.

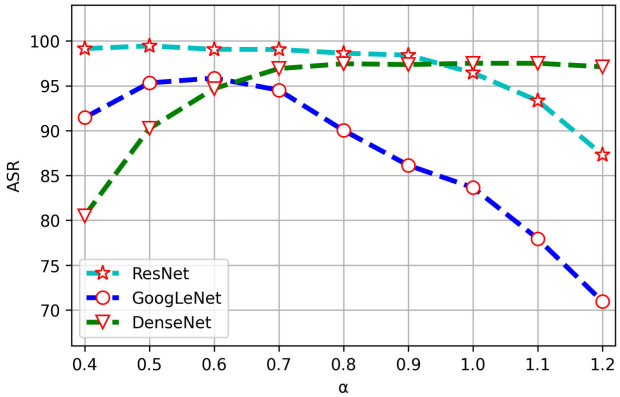


Fig. 10. The attack performance sensitivity of three models, i.e., ResNet, GoogLeNet and DenseNet, against the hyperparameter α (Eq. (4)) on the imagenette dataset.

To evaluate the usefulness of our MsVR module, we replace it with that completely uses the outputs of Grad-CAM, and present the results in Tab. IX, as well as the specific perturbations and Grad-CAM attention maps of the adversarial samples in Fig. 11. For this experiment, we used $M(x) = \text{clamp}\left(\frac{\text{cam}(x)}{\text{mean}(\text{cam}(x))}, 0, 1\right)$ to process the outputs of Grad-CAM, i.e. $cam(\cdot)$, and used it as the mask to replace Eq. (6). Tab. IX shows that MsVR outperforms the Grad-CAM in terms of the average performance of the five models. In terms of the visualization performance, Fig. 11 shows that the perturbation maps by our MsVR largely overlap with the Grad-CAM attention map, and it covers the main object features more completely than the variant using Grad-CAM.

3) Comparison in Loss Functions: To study the performance of our feature loss in Eq. (4), we compared it with two mainstream loss functions, i.e., L2-norm paradigm and cosine similarity in Eq. (3), and present the results in Table X. In this comparison, except the loss function, the other conditions are kept the same. The challenging GoogLeNet model is chosen as the proxy, and the 'MsVR' module is not used to make the comparison of attack capabilities more intuitive. We performed 40 epochs of training iterations for each model. As shown in Table X, the model trained with our loss function largely outperforms that with the other two losses in terms of attack capability. These results, together with those in Tables I and II, further validate the effectiveness of our entire paradigm.

TABLE X

COMPARISON ON L2 NORM (L2), COSINE SIMILARITY (COS), AND OUR LOSS FUNCTION IN THE BRFM MODULE (OURS) ON GOOGLENET. $L_\infty = 0.1$

Loss	ASR↑	SSIM↑	PSNR↑	LF↓	L_2 ↓
Cos	88.37	0.740	22.77	21.78	33.88
L2	87.63	0.740	**22.87**	**20.86**	**33.47**
Ours	**97.20**	**0.740**	22.82	21.66	33.67

We argue that the relatively poor performance of L2 and Cos is due to the fact that their goal is to optimize the representation of all features indiscriminately, which encourages networks to optimize the 'easy' features to rashly strike this goal, while ignoring more 'hard' features whose perturbations do not bring immediate improvement toward the goal, but are critical to attack capacity improvement. In contrast, our approach distinguishes between Hard2Attack and Easy2Attack features, and update Hard2Attack features to mandatorily behave oppositely as those of benign features, so as to avoid the learning shortcut for features with different difficulties in attack process and reduce the learning bias. The obvious improvements by our method for challenging samples over other methods also validate the above conclusion, e.g., the uneven performances in terms of ASR in Fig. 3(a), and the explanation of this phenomenon in Fig. 11.

4) Visualization of Feature Manipulation: In order to shed light on our feature manipulation proposed in section III-B, we visualised the perturbed features by our loss (Eq. (4)) as well as those by the L2 and Cos losses in Fig. 12, where the features of the benign images are used as the benchmark. Meanwhile, the most challenging category of samples, i.e., the 8th class (golf ball) from Imagenette are used for the analysis, on which the worst performance is achieved by

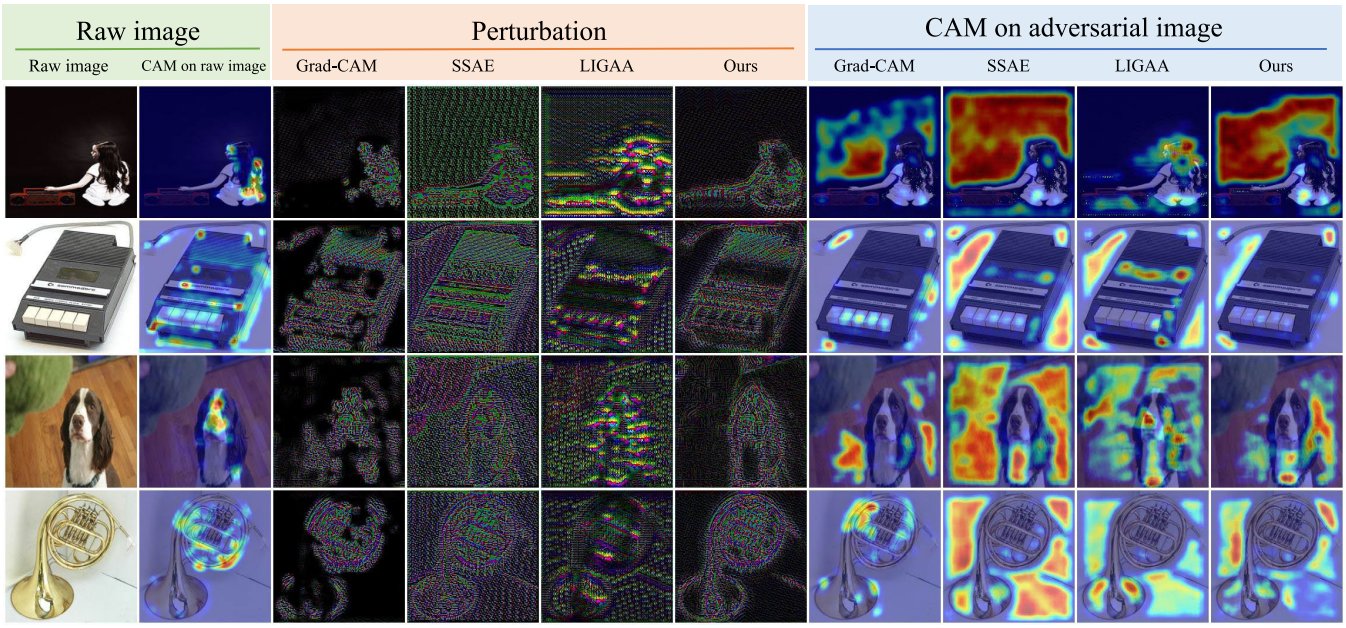
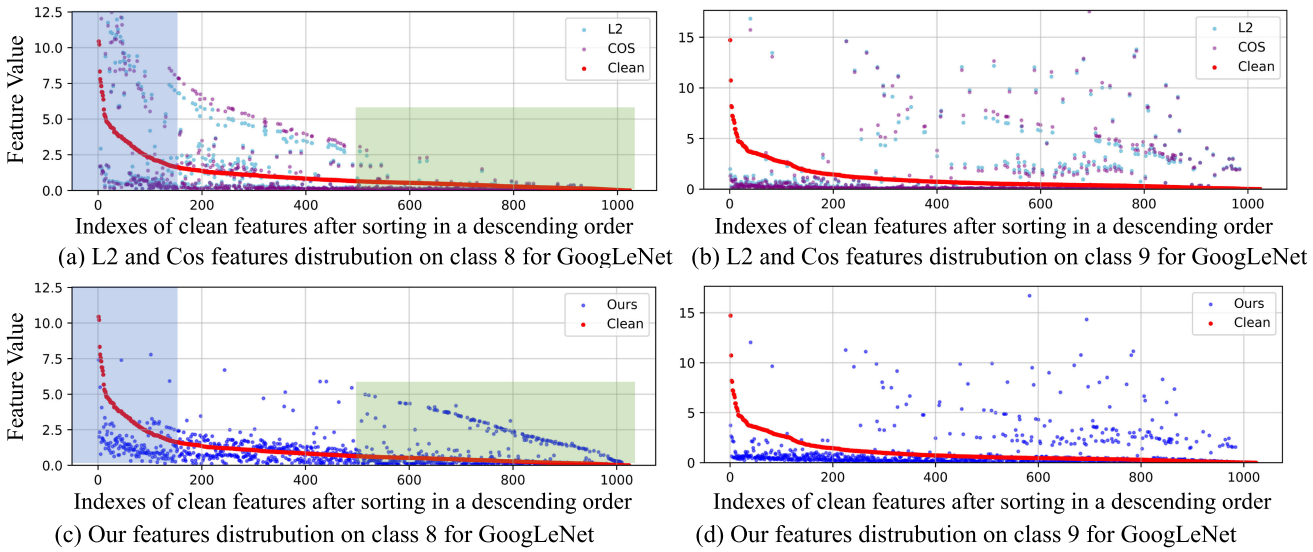


Fig. 11. Visualization of the activation attention maps of the clear images (2nd column) and the adversarial images (7th-10th columns) based on Grad-CAM [66] on GoogLeNet [2]. The 3rd-6th columns present the perturbation noises.



(a) L2 and Cos features distrubution on class 8 for GoogLeNet (b) L2 and Cos features distrubution on class 9 for GoogLeNet

(c) Our features distrubution on class 8 for GoogLeNet (d) Our features distrubution on class 9 for GoogLeNet

Fig. 12. (a)(c) represent the visualisation of attacked features of samples from the 8th class (golf ball), i.e., the most challenging class in Imagenette, using the traditional L2, Cos losses, and the proposed loss in Eq. (4). (b)(d) represent the visualisation of attacked features of the samples from a less challenging class, i.e., the 9th class (parachute), respectively. The horizontal axis label the indexes of benign-samples' features after sorting them in a descending order. The shaded regions in (a)(c) indicate the distributions of the Hard2Attack features by traditional feature regularization and ours.

GoogLeNet with traditional algorithms. Meanwhile, another non-challenging class of samples (the 9th class, parachute) are used for the contrast analysis.

To further shed light on the learning-bias phenomenon observed in Fig. 3, we visualised the distribution of benign and attacked features specific to both a challenging and non-challenging class in Fig. 12. As shown in Fig. 12(a)(b), the Hard2Attack features with large activation values (blue shaded regions) have not been sufficiently suppressed, and those with small activation values (green shaded regions) have not been sufficiently activated when using the L2-norm and cosine similarity-alike losses. By contrast, Fig. 12(c)(d) show that our method can achieve the above two goals for these Hard2Attack features. In particular, as shown in the green shaded regions in Figs. 12(a),(c), the Hard2Attack features with small activation values for challenging-class-samples in Fig. 12(a) are obviously enlarged in Fig. 12(c).

5) Inference Speed and GPU Memory Usage: To evaluate the efficiency of our algorithm, we present the attack speed and runtime memory overhead of different attack methods in Table XI. It shows that the generation speed of adversarial samples with the generative attack methods is much higher than that of the gradient-based attacks, although our generation speed is slightly lower than that of AdvGAN and GCMA, and our occupied memory is larger than that needed for other generative attack methods.

Since the BrFM module only involves the design of the loss function, it is unrelated to the inference speed and GPU

TABLE XI
COMPARISON OF INFERENCE SPEEDS IN TERMS OF FRAME PER SECOND (FPS) AND GPU MEMORY USAGE OF DIFFERENT MODELS

Inference speeds (FPS)					
Models	efficientnet	densenet	resnet	googlenet	mobilenet
FGSM [5]	27.0	16.7	64.2	16.2	33.1
SSAE [1]	90.2	86.2	97.5	75.5	68.7
AdvGAN [21]	127.5	139.8	140.5	133.1	120.9
LIGAA [41]	93.9	96.5	95.2	84.6	83.2
GCMA [44]	131.8	136.5	136.5	121.4	119.0
Ours	136.0	129.1	129.1	105.6	111.5
Ours(w/o MsVR)	139.1	131.4	131.2	106.5	113.4
GPU Memory Usage (M)					
Models	efficientnet	densenet	resnet	googlenet	mobilenet
FGSM [5]	783.4	2301.2	482.0	2460.9	985.8
SSAE [1]	228.1	256.3	291.1	250.3	220.1
AdvGAN [21]	207.0	235.3	270.0	229.0	198.8
LIGAA [41]	230.7	263.0	296.1	256.3	224.5
GCMA [44]	248.3	280.6	313.0	273.1	241.3
Ours	291.3	320.7	354.4	313.4	283.4
Ours(w/o MsVR)	291.3	320.7	354.4	313.4	283.4

memory consumption during the inference phase. Table XI shows that our MsVR module does not affect the maximum GPU memory usage during the inference process (since model inference and the MsVR module are calculated serially, not in parallel). Additionally, thanks to our parallelized design of the window variance calculation, the computation overhead of the MsVR module only reduces the inference speeds (FPS) by about $1\% - 2\%$.

V. CONCLUSION AND DISCUSSION

In this work, we propose a simple yet effective generative attack, to improve the imperceptibility of perturbation noises w.r.t. human eyes while maintaining a reasonable attack capacity. (i) Considering the feature regularization losses used in classical generative adversarial attacks may result in learning shortcut, i.e., the successful perturbations are biased toward the Easy2Attack features, while producing undesirable perturbations not conducive to attack capacity for those Hard2Attack features, we newly propose to differentiate between Hard2Attack and Easy2Attack features, and regularize the Hard2Attack features to behave oppositely to those of benign samples in feature space. (ii) To reduce the perturbations distributed on region boundaries in enhancing their imperceptibility w.r.t. human eyes, we devise a soft regularization based on multi-scale variance masks. In this way, our mask-based soft regularization can not only trade off the number of partition regions and the imperceptibility via the multi-scale modeling, but also maintain a reasonable attack capacity in improving imperceptibility via a soft constraint. Experimental results on different datasets validate the effectiveness of our method in terms of attack success rate and various imperceptibility metrics, compared with the related state-of-the-art generative attack algorithms.

Despite the excellent attack and imperceptibility performances achieved by our method, there is still a potential for further improvement in feature selection with the loss function setup. (i) Since updating of some features can have undesirable knock-on effects on many others, we will attempt to locate the most Hard2Attack features in terms of the importance of interacting with others for this update. (ii) Despite the simplicity and effectiveness of our mask generation and regularization, the model may only focus on a small portion of an image rather than the entire highly semantic regions in the image. We will integrate these global semantics with local regions in generating masks, so as to enhance semantic rationality in improving the imperceptibility of the imposed perturbations. (iii) Since our method may fail in acquiring a sufficient feature representation capacity for the images whose class has few samples, improving our modules to adapt to few-shot classes is another future directions.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful comments and constructive suggestions.

REFERENCES

[1] S. Lu et al., "Discriminator-free generative adversarial attack," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 1544–1552.
[2] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[3] J. Howard. (2020). *Imagenette*. [Online]. Available: https://github.com/fastai/imagenette
[4] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Representat.*, 2014.
[5] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Representat.*, 2015, pp. 1–11.
[6] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Int. Conf. Learn. Representat.*, 2017.
[7] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learn. Representat.*, 2018.
[8] Y. Dong et al., "Boosting adversarial attacks with momentum," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9185–9193.
[9] J. Lin, C. Song, K. He, L. Wang, and J. E. Hopcroft, "Nesterov accelerated gradient and scale invariance for adversarial attacks," 2019, *arXiv:1908.06281*.
[10] A. Peng, Z. Lin, H. Zeng, W. Yu, and X. Kang, "Boosting transferability of adversarial example via an enhanced Euler's method," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.
[11] Z. Qin et al., "Boosting the transferability of adversarial attacks with reverse adversarial perturbation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 29845–29858.
[12] Z. Ge, H. Liu, W. Xiaosen, F. Shang, and Y. Liu, "Boosting adversarial transferability by achieving flat local maxima," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 70141–70161.
[13] C. Xie et al., "Improving transferability of adversarial examples with input diversity," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2730–2739.
[14] Y. Dong, T. Pang, H. Su, and J. Zhu, "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4312–4321.
[15] X. Wang, X. He, J. Wang, and K. He, "Admix: Enhancing the transferability of adversarial attacks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 16158–16167.
[16] Y. Long et al., "Frequency domain model augmentation for adversarial attack," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 549–566.
[17] Q. Lin et al., "Boosting adversarial transferability across model genus by deformation-constrained warping," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 4, pp. 3459–3467.
[18] X. Dai, K. Liang, and B. Xiao, "AdvDiff: Generating unrestricted adversarial examples using diffusion models," 2023, *arXiv:2307.12499*.

[19] D. Liu, X. Wang, C. Peng, N. Wang, R. Hu, and X. Gao, "Adv-diffusion: Imperceptible adversarial face identity attack via latent diffusion model," in *Proc. AAAI Conf. Artif. Intell.*, Mar. 2024, vol. 38, no. 4, pp. 3585–3593.

[20] M. Kang, D. Song, and B. Li, "Diffattack: Evasion attacks against diffusion-based adversarial purification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., Red Hook, NY, USA: Curran Associates, 2023, pp. 73919–73942.

[21] C. Xiao, B. Li, J. Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 3905–3911.

[22] Q. Zhang et al., "Beyond ImageNet attack: Towards crafting adversarial examples for black-box domains," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–18.

[23] K. K. Nakka and M. Salzmann, "Learning transferable adversarial perturbations," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 13950–13962.

[24] W. J. Kim, S. Hong, and S.-E. Yoon, "Diverse generative perturbations on attention space for transferable adversarial attacks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 281–285.

[25] A. Aich, S. Li, C. Song, M. S. Asif, S. V. Krishnamurthy, and A. K. Roy-Chowdhury, "Leveraging local patch differences in multi-object scenes for generative adversarial attacks," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 1308–1318.

[26] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2017, pp. 39–57.

[27] M. Ibbotson and B. Krekelberg, "Visual perception and saccadic eye movements," *Current Opinion Neurobiol.*, vol. 21, no. 4, pp. 553–558, Aug. 2011.

[28] M. Z. Hameed and A. Gyorgy, "Perceptually constrained adversarial attacks," 2021, *arXiv:2102.07140*.

[29] Z. Zhao, Z. Liu, and M. Larson, "Towards large yet imperceptible adversarial image perturbations with perceptual color distance," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 1039–1048.

[30] C. Luo, Q. Lin, W. Xie, B. Wu, J. Xie, and L. Shen, "Frequency-driven imperceptible adversarial attack on semantic similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 15315–15324.

[31] J. P. Göpfert, A. Artelt, H. Wersing, and B. Hammer, "Adversarial attacks hidden in plain sight," in *Proc. 18th Int. Symp. Intell. Data Anal.*, Konstanz, Germany. Cham, Switzerland: Springer, Apr. 2020, pp. 235–247.

[32] G. E. Legge and J. M. Foley, "Contrast masking in human vision," *J. Opt. Soc. Amer.*, vol. 70, no. 12, pp. 1458–1471, 1980.

[33] B. Luo, Y. Liu, L. Wei, and Q. Xu, "Towards imperceptible and robust adversarial example attacks against neural networks," in *Proc. AAAI*, 2018, pp. 1–8.

[34] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[35] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[36] Z. Zhao, Z. Liu, and M. Larson, "Adversarial image color transformations in explicit color filter space," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 3185–3197, 2023.

[37] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," in *Proc. Int. Conf. Learn. Representat.*, 2018, pp. 1–15.

[38] O. Poursaeed, I. Katsman, B. Gao, and S. Belongie, "Generative adversarial perturbations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4422–4431.

[39] X. Dong et al., "GreedyFool: Distortion-aware sparse adversarial attack," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 11226–11236.

[40] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "On generating transferable targeted perturbations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7708–7717.

[41] H. Zhu, Y. Zhu, H. Zheng, Y. Ren, and W. Jiang, "LIGAA: Generative adversarial attack method based on low-frequency information," *Comput. Secur.*, vol. 125, Feb. 2023, Art. no. 103057.

[42] X. Gao et al., "Scale-free and task-generic attack: Generating photo-realistic adversarial patterns with patch quilting generator," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2024, pp. 2985–2989.

[43] W. Feng, N. Xu, T. Zhang, and Y. Zhang, "Dynamic generative targeted attacks with pattern injection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 16404–16414.

[44] K. Chen, Z. Wei, J. Chen, Z. Wu, and Y.-G. Jiang, "GCMA: Generative cross-modal transferable adversarial attacks from images to videos," in *Proc. 31st ACM Int. Conf. Multimedia*, 2023, pp. 698–708.

[45] Z. Zhu et al., "GE-AdvGAN: Improving the transferability of adversarial samples by gradient editing-based adversarial generative model," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, 2024, pp. 706–714.

[46] S. Jandial, P. Mangla, S. Varshney, and V. Balasubramanian, "AdvGAN++: Harnessing latent layers for adversary generation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 2045–2048.

[47] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[48] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[49] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, 2004, p. 178.

[50] S. Zhang, B. Zheng, P. Jiang, L. Zhao, C. Shen, and Q. Wang, "Perception-driven imperceptible adversarial attack against decision-based black-box models," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 3164–3177, 2024.

[51] Z. Wang, M. Song, S. Zheng, Z. Zhang, Y. Song, and Q. Wang, "Invisible adversarial attack against deep neural networks: An adaptive penalization approach," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 3, pp. 1474–1488, May 2021.

[52] X. Dong et al., "Robust superpixel-guided attentional adversarial attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12892–12901.

[53] S. Zhang, D. Zuo, Y. Yang, and X. Zhang, "A transferable adversarial belief attack with salient region perturbation restriction," *IEEE Trans. Multimedia*, vol. 25, pp. 4296–4306, 2022.

[54] S. Chen, Z. He, C. Sun, J. Yang, and X. Huang, "Universal adversarial attack on attention and the resulting dataset DAmageNet," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 2188–2197, Apr. 2022.

[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[56] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf.*, Munich, Germany. Cham, Switzerland: Springer, Oct. 2015, pp. 234–241.

[57] N. Inkawhich, W. Wen, H. H. Li, and Y. Chen, "Feature space perturbations yield more transferable adversarial examples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7066–7074.

[58] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[59] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[60] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[61] G. Karolina Dziugaite, Z. Ghahramani, and D. M. Roy, "A study of the effect of JPG compression on adversarial images," 2016, *arXiv:1608.00853*.

[62] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," 2017, *arXiv:1704.01155*.

[63] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "A self-supervised approach for adversarial robustness," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 262–271.

[64] C.-H. Ho and N. Vasconcelos, "Disco: Adversarial defense with local implicit functions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 23818–23837.

[65] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do vision transformers see like convolutional neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12116–12128.

[66] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.