Diffusion Models for Imperceptible and Transferable Adversarial Attack

Jianqi Chen, Hao Chen, Keyan Chen, Yilan Zhang, Zhengxia Zou, and Zhenwei Shi*, *Senior Member, IEEE*

***Abstract*—Many existing adversarial attacks generate L_p-norm perturbations on image RGB space. Despite some achievements in transferability and attack success rate, the crafted adversarial examples are easily perceived by human eyes. Towards visual imperceptibility, some recent works explore unrestricted attacks without L_p-norm constraints, yet lacking transferability of attacking black-box models. In this work, we propose a novel imperceptible and transferable attack by leveraging both the generative and discriminative power of diffusion models. Specifically, instead of direct manipulation in pixel space, we craft perturbations in the latent space of diffusion models. Combined with well-designed content-preserving structures, we can generate human-insensitive perturbations embedded with semantic clues. For better transferability, we further "deceive" the diffusion model which can be viewed as an implicit recognition surrogate, by distracting its attention away from the target regions. To our knowledge, our proposed method, *DiffAttack*, is the first that introduces diffusion models into the adversarial attack field. Extensive experiments conducted across diverse model architectures (CNNs, Transformers, and MLPs), datasets (ImageNet, CUB-200, and Standford Cars), and defense mechanisms underscore the superiority of our attack over existing methods such as iterative attacks, GAN-based attacks, and ensemble attacks. Furthermore, we provide a comprehensive discussion on future research avenues in diffusion-based adversarial attacks, aiming to chart a course for this burgeoning field.**

***Index Terms*—Adversarial attack, diffusion model, imperceptible attack, transferable attack.**

I. Introduction

RECENT years have witnessed remarkable performance exhibited by deep neural networks (DNNs) across a range of domains, including autonomous driving [1], [2], medical image analysis [3], [4], remote sensing [5], [6], *etc*. Notwithstanding the indisputable advances, early investigations [7] have elucidated the susceptibility of DNNs to meticulously engineered subversions (hereafter referred to as "adversarial examples"), which may induce grievous mistakes in real-world applications [8], [9]. Moreover, the transferability of these adversarial examples across distinct model architectures [10] poses an even greater hazard to practical implementations. Therefore, it is of the utmost necessity to uncover as many lacunae in machine perception – what may be termed "blind spots" – as can feasibly be achieved, so as to bolster the DNNs' resilience when faced with adversarial challenges.

Compared to white-box attacks [11], [12] that the attacker can access the architecture and parameters of the target model, black-box attacks [13], [14], [10] can not obtain the target's information and thus are much closer to real-world scenarios. Among black-box directions, we here focus on the transfer-based attacks [10] that directly apply the adversarial examples constructed on a surrogate model to fool the target model. By adopting different optimization strategies [15], [16], designing various loss functions [17], [18], leveraging multiple data augmentations [19], [20], [21], *etc.*, existing approaches have achieved much success and improved the attack's transferability.

L_p-norm based methods. Most of the above methods adopt L_p-norm in RGB color space as an indicator of human perception and constrain the amplitude of the adversarial perturbations under a specific value. Despite the efforts paid, these pixel-based attacks are still easy to be perceived by human eyes, and L_p-norm was recently found unsuitable to measure the perceptual distance between two images [22], [23]. From the examples displayed in Figure 1, the perturbations optimized by L_p loss are noticeable and appear similar to high-frequency noise, which indicates overfitting on the surrogate model, as revealed in [24], [25]. Despite having low L_∞ values, these perturbations can hinder the transferability to other black-box models [26], [24] and is easy to be defended against by purification defenses [27], [28].

Towards imperceptible attacks. Recent works [26], [29], [22] explored new ways to deceive human perception without using the L_p-norm constraint (*a.k.a.* unrestricted attacks). By applying perturbations on spaces such as object attribute [26], color mapping matrix [29], *etc.*, the adversarial examples are well imperceptible despite large L_p-norm values in RGB space. Furthermore, these studies [26], [29] have shown that perturbations generated by unrestricted attacks—often characterized by large-scale patterns with high-level continuous semantics, as opposed to the high-frequency noise typical of pixel-level perturbations—enhance transferability to other black-box models and even to defended models. However, these methods' transferability still lags behind the pixel-based ones.

In this work, we propose a novel unrestricted attack based on diffusion models [30]. Instead of manipulating pixels directly, we optimize the latent of an off-the-shelf pretrained diffusion

The work was supported by the National Natural Science Foundation of China under the Grants 62125102, the National Key Research and Development Program of China (Grant No. 2022ZD0160401), the Beijing Natural Science Foundation under Grant JL23005, and the Fundamental Research Funds for the Central Universities. *(Corresponding author: Zhenwei Shi (email: shizhenwei@buaa.edu.cn))*

Jianqi Chen, Keyan Chen, Yilan Zhang, and Zhenwei Shi are with the Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China, and with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China, and also with the Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China.

Hao Chen is with the Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China.

Zhengxia Zou is with the Department of Guidance, Navigation and Control, School of Astronautics, Beihang University, Beijing 100191, China.

Our code is publicly accessible at https://github.com/WindVChen/DiffAttack.

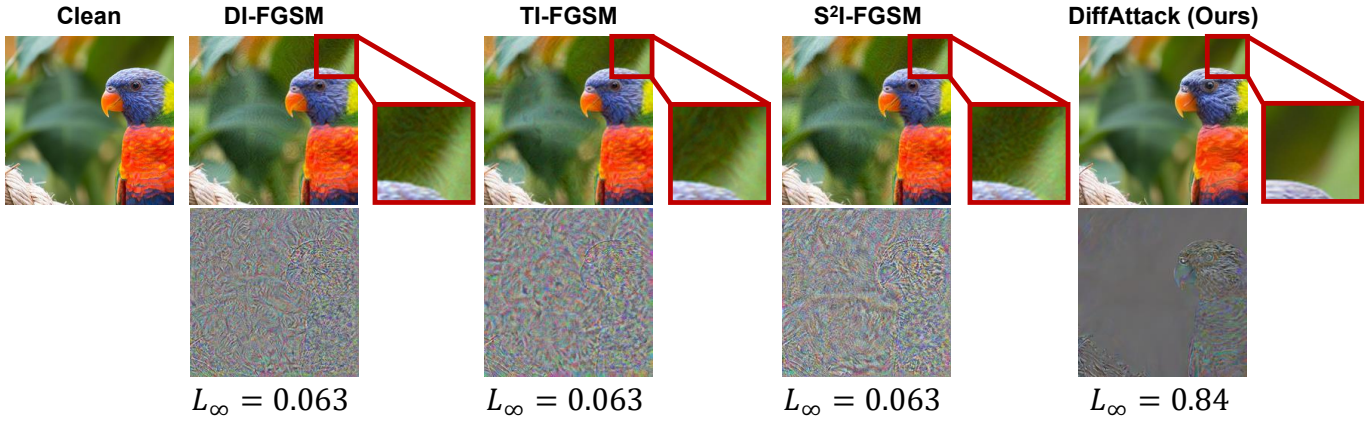

Fig. 1. **Adversarial perturbations crafted by some attacks.** The second row denotes the difference between the clean image and the adversarial example. Please zoom in for a better view.

model [30]. Besides the basic transferability advantages of high-level perturbations mentioned above, our motivation for introducing the diffusion model into the adversarial attack domain stems primarily from its two beneficial properties. 1) *Good imperceptibility*. Diffusion models, originally designed for image synthesis, tend to generate natural-looking images in line with human perception. This inherent quality aligns well with the imperceptibility requirement of adversarial attacks. Moreover, the iterative denoising process within diffusion models aids in reducing perceptible high-frequency noise. 2) *Approximation of an implicit surrogate*. Despite being initially designed for image synthesis, diffusion models trained on large-scale datasets exhibit a notable discriminative capability [31], [32]. This feature enables us to approximate them as implicit surrogate models for transfer-based attacks. Leveraging this "implicit surrogate", we can potentially enhance transferability across different models and defenses. Furthermore, the denoising process of diffusion models, akin to a robust purification defense [27], can further bolster the effectiveness of our attack against defensive mechanisms.

To harness the favorable attributes of diffusion models, our work encompasses three key aspects. *Firstly*, we establish a foundational attack framework that initially converts clean images into noise and subsequently introduces modifications in the latent space. In evaluating different methods for content editing, such as guided text manipulation or latent code manipulation, we opt for operating on the latents of diffusion models. This choice significantly enhances the success of our attack. *Secondly*, we scrutinize the potential recognition capabilities of the pretrained diffusion model. We propose to deviate the cross-attention maps between text and image pixels, in which way we can transform the diffusion model into an implicit surrogate model that can be practically deceived and attacked. *Finally*, to avoid distorting the initial semantics, we delve into the self-attention's structure extraction capability within the diffusion model. We propose leveraging it as a constraint to mitigate context distortion, while also considering specific measures like controlling inversion strength. We term the proposed unrestricted attack as *DiffAttack*, and our contributions can be summarized as follows:

- As far as we know, we are the first to reveal that with its remarkable generative and implicit discriminative capabilities, the diffusion model is a promising foundation for creating adversarial examples that exhibit both high imperceptibility and transferability.
- We propose *DiffAttack*, a novel unrestricted attack where the good properties of diffusion models are leveraged by careful designs. By utilizing the cross- and self-attention maps and attacking the latent of the diffusion model, *DiffAttack* is both imperceptible and transferable.
- Extensive experiments on a variety of model architectures, datasets, and defense methods have demonstrated the superiorities of our work over the existing attack methods. These findings underscore the promising potential of our research direction.

II. Related Works

Adversarial Attacks. Since Szegedy *et al.* [7] found that DNNs can be deceived by imperceptibly small perturbations applied on images, adversarial attacks have long attracted significant attention in the deep learning field [12], [10]. Generally, the existing attacks can be divided into two parts: white-box attack and black-box attack. For white-box scenarios [11], [12], the attacker can get access to the model structures and parameters of the target model. Thus, strong adversarial examples can be crafted by directly using the backpropagated gradients. For black-box scenarios, there is limited information on the target model and it is closer to the real-world situation. Current approaches are either based on queries [13], [33] or on the cross-model transferability of adversarial examples [16], [10]. Specifically, the former ones query the black-box model many times. With the queried results, they generate adversarial examples either by gradient approximation [33] or by random search [13]. The transfer-based attacks resort to a surrogate model. By crafting perturbations in the same way as white-box attacks, they expect these adversarial examples can also have a good effect on the target model. In this work, we focus on the transfer-based part.

Transferable Attacks. To enhance the generalization of adversarial examples crafted on surrogate models, previous works put a lot of effort into keeping perturbations from getting stuck in a model-specific local optimum that overfits the

surrogate model and cannot transfer well to other methods. [34], [35] adopted the straightforward strategy of *model ensembles* to attack as many models as possible by finding an optimum updated direction. [19], [20], [21] proposed to leverage *data augmentations* to diversify the inputs, which ensures the attack robustness under different scenarios. [17], [36], [18] applied *loss functions* on the feature space which demonstrated good performance on black-box targets. [15], [16] combined momentum into *optimization schedules* to help jump out of local optimum. Despite the much improvement in the transferability, these works mostly conduct attacks with L_p-norm constraint on RGB pixel space, resulting in high-frequency noises and patterns (see Figure 1) which, though hold a relatively low value on L_p-norm, are easy to be perceived by humans. In contrast, our *DiffAttack* perturbs the latent in diffusion models, achieving good imperceptibility together with excellent transferability across various black-box models.

Unrestricted Attacks. Since L_p-norm in RGB space was found not ideal for measuring the perceptual distance [26], [29], recent research turns to unconstraint and proposes unrestricted but imperceptible attacks. Bhattad *et al.* [37] perturbed images from the perspective of color and texture. Zhao *et al.* [22] adopted CIEDE2000 which can better indicate the perceptual color loss. Qiu *et al.* [38] and Jia *et al.* [26] achieve imperceptibility by modifying the attributes of the images, especially human faces. Yuan *et al.* [29] constructed a color distribution library, which is used to find a successful distribution for adversarial attacks. However, despite their good imperceptibility, these methods generally cannot compete with the aforementioned pixel-based methods in terms of transferability. Our work also falls in this direction but achieves better transferability and imperceptibility, and is the first to explore the strength of diffusion models in crafting unrestricted attacks.

Diffusion Models. Recently, diffusion models [39] have attracted extensive attention and shown their fabulous power. Images are first converted into purely Gaussian noise in the *forward process* and then a U-Net structure is trained to predict the added noise in each timestep of the *denoising process*. Being trained on large numbers of data, the diffusion models [40], [41], [30] can either generate high-quality images from randomly sampled noise, or more specific ones that follow the guidance of text prompt. Due to its significant performance, the diffusion model has also diffused to other areas, such as image inpainting [42], [43], image super-resolution [44], real image editing [45], [46], *etc.* Recent work further showed that the pretrained diffusion models can be taken as good recognition models [31], [32] and denoisers [27]. Despite the many applications mentioned above, the potential of diffusion models in the adversarial attack field remains underexplored.

III. Method

A. Problem Formulation

Given a clean image x and its corresponding label y, attackers aim to craft perturbations that can deviate the decision of a classifier F_θ (θ denotes the model's parameters) from correct to wrong:

$$F_\theta(\mathrm{Attack}(\mathrm{x};\mathrm{G}_\phi)) = F_\theta(x') \neq y \tag{1}$$

where $\mathrm{Attack}(\cdot)$ is the attack approach and x' is the crafted adversarial example. Since F_θ is inaccessible in black-box scenarios, the adversarial examples are crafted on a surrogate model G_ϕ.

Different from previous pixel-based attacks [16], [19] that apply L_p-norm constraints on pixel values ($\|\epsilon\|_p < c$, where ϵ is the perturbation and c is a hyperparameter), we impose perturbations in the latent space of the diffusion model and rely on properties of the diffusion model to achieve visually natural and successful attacks. In the following parts, we will first outline essential background information on diffusion models and subsequently provide a detailed description of our design.

B. Formulation of DDPM and DDIM Inversion

Denoising Diffusion Probabilistic Models (DDPMs) [48] are a class of generative models that sample images by gradually denoising an initial Gaussian noise. There is a *forward process* and a *denoising process* in DDPMs. The *forward process* is to gradually add Gaussian noise to the original image x_0 and thus produces a series of noisy latents $x_1, x_2, \cdots, x_T$:

$$q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \tag{2}$$

where $\beta_t \in (0,1)$. When T is large enough, the last latent x_T will approximately follow an isotropic Gaussian distribution.

Instead of iteratively calculating the intermediate latents to get x_t, a good property of the *forward process* is that we can directly sample x_t from x_0:

$$q(x_t|x_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{3}$$

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0,\mathbf{I}) \tag{4}$$

where $\alpha_t = 1-\beta_t$, $\bar{\alpha}_t = \prod_{s=0}^{t}\alpha_s$.

The *denoising process* is to draw a new sample from the distribution $q(x_0)$. Starting from $x_T \sim \mathcal{N}(0,\mathbf{I})$, we can get a new sample by iteratively sampling the posteriors $q(x_{t-1}|x_t)$. Since $q(x_{t-1}|x_t)$ is intractable due to the unknown data distribution $q(x_0)$, a neural network p_θ is trained to approximate that by predicting the mean and covariance of $q(x_{t-1}|x_t)$, which is shown to also be Gaussian distributions [49]:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(\mu_\theta(x_t,t), \Sigma_\theta(x_t,t)) \tag{5}$$

Since $\mu_\theta(x_t,t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right)$, Ho *et al.* [48] simplified the objective function by only predicting the noise $\epsilon_\theta(x_t,t)$:

$$\min_\theta \mathcal{L}(\theta) = \mathbb{E}_{x_0,\epsilon\sim N(0,I),t}\|\epsilon - \epsilon_\theta(x_t,t)\|_2^2 \tag{6}$$

After we get the trained $\epsilon_\theta(x_t,t)$ (normally a U-Net structure), we can conduct a sampling as follows:

$$x_{t-1} = \mu_\theta(x_t,t) + \sigma_t z, \quad z \sim N(0,I). \tag{7}$$

Since the classic DDPMs are essentially a Markov chain and they require a large timestep T to achieve good performance. To accelerate DDPMs sampling process, Song *et al.* [47] generalize

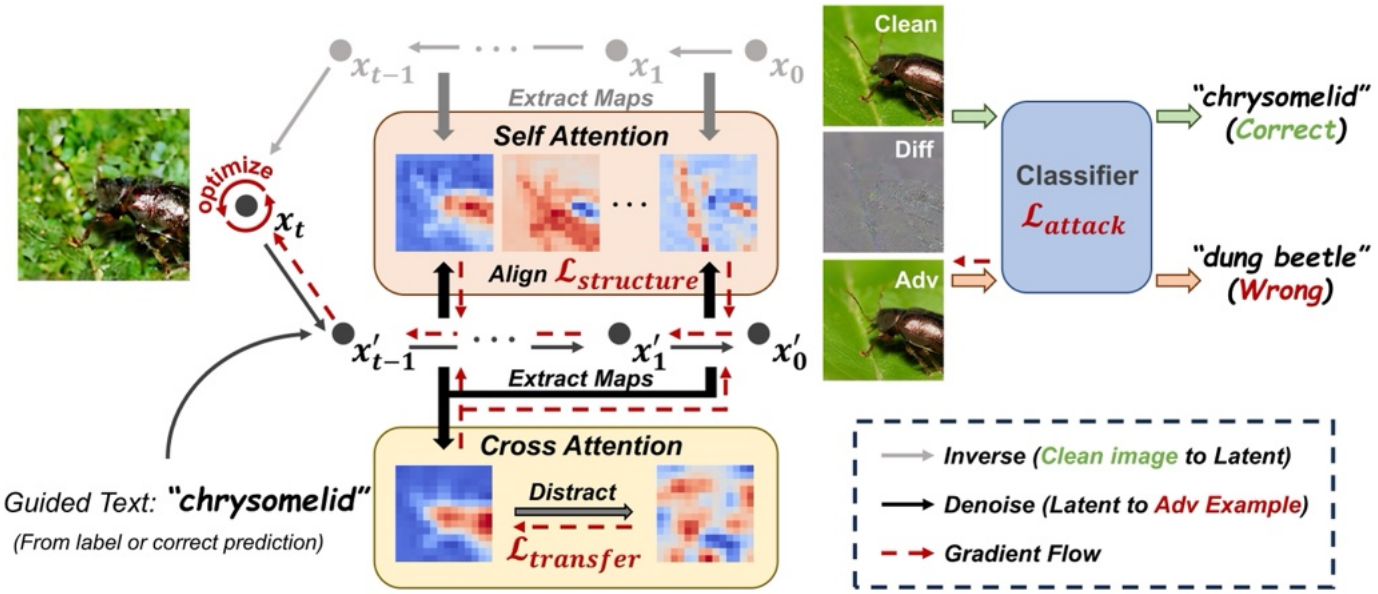

Fig. 2. **Framework of *DiffAttack*.** We adopt Stable Diffusion [30] and leverage DDIM Inversion [47] to convert the clean image into the latent space. The latent is optimized to deceive the classifier. The cross-attention maps are leveraged to "deceive" the diffusion model, and we use self-attention maps to preserve the structure. For simplicity, we here do not display the unconditional optimization, whose details can be referred to Section III-E.

DDPMs from a particular Markovian process to non-Markovian processes:

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \left(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, t)}{\sqrt{\bar{\alpha}_t}} \right) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2} \cdot \epsilon_\theta(x_t, t) + \sigma_t z, \quad z \sim N(0, I) \tag{8}$$

By setting $\sigma_t = 0$, we then get a deterministic sampling process (from x_T to x_0), which is the DDIM's principle.

Since the deterministic process of DDIM can be further taken as Euler integration for solving ordinary differential equations (ODEs)[47], we can map a real image back to its corresponding latent by reversing the process. This operation, named DDIM Inversion, paves the way for later editing of real images [50], [46]. By rewriting Eq. 8, the denoising process of DDIM is as follows:

$$x_{t-1} - x_t = \sqrt{\bar{\alpha}_{t-1}} \left[\left(\sqrt{1/\bar{\alpha}_t} - \sqrt{1/\bar{\alpha}_{t-1}} \right) x_t + \left(\sqrt{1/\bar{\alpha}_{t-1} - 1} - \sqrt{1/\bar{\alpha}_t - 1} \right) \epsilon_\theta(x_t, t) \right] \tag{9}$$

We can then encode the real image into the latent space by reversing the above formulation:

$$x_{t+1} - x_t = \sqrt{\bar{\alpha}_{t+1}} \left[\left(\sqrt{1/\bar{\alpha}_t} - \sqrt{1/\bar{\alpha}_{t+1}} \right) x_t + \left(\sqrt{1/\bar{\alpha}_{t+1} - 1} - \sqrt{1/\bar{\alpha}_t - 1} \right) \epsilon_\theta(x_t, t) \right] \tag{10}$$

C. Basic Framework

We display in Figure 2 the whole framework of *DiffAttack*, where we adopt the open-source Stable Diffusion [30] that pretrained on extremely massive text-image pairs. Since adversarial attacks aim to fool the target model by perturbing the initial image, they can be approximated as a special kind of real image editing. Similar to recent diffusion editing approaches [50], [46], [45], our framework leverages the DDIM Inversion technology [47], where the clean image is mapped back into the diffusion latent space by reversing the deterministic sampling process:

$$x_t = \text{Inverse}(x_{t-1}) = \underbrace{\text{Inverse} \circ \cdots \circ \text{Inverse}}_{t}(x_0) \tag{11}$$

where $\text{Inverse}(\cdot)$ denotes the DDIM Inversion operation (Please see Section III-B for details. In Eq. 11, we ignore the autoencoder stage of the Stable Diffusion [30] for simplicity). We apply the inversion for several timesteps from x_0 (the initial image) to x_t. A high-quality reconstruction of x_0 can then be expected if we conduct the deterministic denoising process from x_t [51], [47].

Many of the existing image editing approaches [50], [46] proposed to modify text embeddings for image editing, through which way, the image latent x_t can gradually shift to the target semantic space during the iterative denoising process with the text guidance. However, in our explorations (see Section IV-D), we found that the perturbations on the guided text embeddings would be hard to work on the other black-box models, leading to weak transferability. Therefore, different from the editing approaches, we here propose to directly perturb the latent x_t:

$$\arg\min_{x_t} \mathcal{L}_{attack} = -J(x', y; G_\phi) \tag{12}$$

$$x' = x'_0 = \underbrace{\text{Denoise} \circ \cdots \circ \text{Denoise}}_{t}(x_t) \tag{13}$$

where $J(\cdot)$ is the cross-entropy loss and $\text{Denoise}(\cdot)$ denotes the diffusion denoising process. An initial concern might arise regarding the potential generation of unnatural results using this straightforward method. However, we can observe in Figure I that the difference is almost indistinguishable between the

image reconstructed from the perturbed latent and the initial clean one. Furthermore, we can notice that the difference image encapsulates numerous high-level semantic cues, as opposed to the high-frequency noise typically associated with pixel-based attacks. We attribute this phenomenon to the denoising process of the diffusion model, which effectively reduces perceptible high-frequency noise. These perturbations on high-level semantics not only enhance imperceptibility but also improve the attack's transferability [26].

D. "Deceive" Diffusion Model

According to the research by [27], the denoising process of the diffusion model is a strong adversarial purification defense. Thus, our perturbed latent will experience purification before being decoded to the final image, which then ensures the naturalness of crafted adversarial examples and also the robustness towards other purification denoises (see Section IV-B2). In addition to leveraging the denoising component, we here go a further step to enhance our attack's transferability, by leveraging cross-attention maps within the diffusion model.

For a text-to-image diffusion model, such as Stable Diffusion, the noise prediction model transitions from $\epsilon_\theta(x_t, t)$ in Eq. 6 to $\epsilon_\theta(x_t, t, C)$, incorporating an additional guidance text prompt C. Information from the guidance text is integrated into the normal U-Net flow with cross-attention layers. Specifically, for a latent representation x_t and a guidance text C, we obtain the deep features $\phi(x_t) \in \mathbb{R}^{N\times N\times d_\phi}$ of the latent in the U-Net structure and the text embeddings $\psi(C) \in \mathbb{R}^{M\times d_\psi}$ of the text. These are projected to Q and K matrices with linear projection matrices $W_Q \in \mathbb{R}^{d_\phi \times d}$ and $W_K \in \mathbb{R}^{d_\psi \times d}$. The fusion of information then proceeds as follows:

$$\mathcal{A} = \text{softmax}(\frac{QK^T}{\sqrt{d}}) = \text{softmax}(\frac{(\phi(x_t)W_Q)(\psi(C)W_K)^T}{\sqrt{d}}) \tag{14}$$

where the resulted attention map $\mathcal{A} \in \mathbb{R}^{N\times N\times M}$. Since M corresponds to the number of text tokens, a specific word's matched attention map $\mathcal{A}_m \in \mathbb{R}^{N\times N}$ can be easily queried from $\mathcal{A}$.

Utilizing the aforementioned cross-attention layers, we can easily extract attention maps by pairing the inversed latent of an input image with words from its corresponding caption. As depicted in Figure 3, the cross-attention maps derived from the reconstruction (denoising) process of the inversed latent exhibit a strong correlation between the guided text and image pixels. This correlation underscores the recognition potential of pre-trained diffusion models, corroborated by Hertz *et al.* [52]. Recent studies [31], [32] have capitalized on this recognition capability for downstream tasks. Thus, we posit that the diffusion model, trained extensively on text-image paired data, can be approximated as an implicit recognition model. Given the widespread text-image paired training data, Stable Diffusion can synthesize a wide range of objects, demonstrating its effective image-text data representation [32], which is closely tied to its object recognition capability [31]. Thus, the implicit recognition capability of Stable Diffusion should generalize well across most object classification scenarios. Our aim lies in exploiting this model by enabling our crafted attacks to "deceive" it, potentially enhancing their transferability to other black-box models.

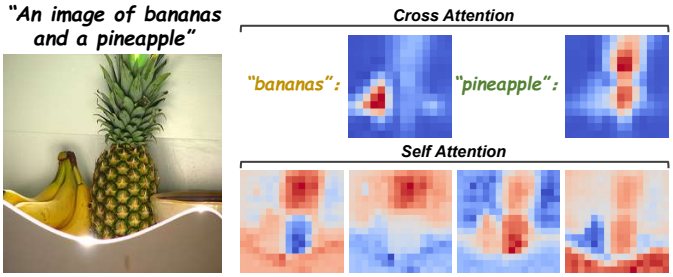

Fig. 3. **Visualization of cross- and self- attention maps.** There is a strong relationship between text and pixels in cross-attention, while self-attention can well reveal structure.

Denote C as the caption of the clean image, which we set to the groundtruth category's name (we can also simply use the predicted category of G_ϕ, and thus not rely on true labels). We accumulate the cross-attention maps between image pixels and C in all the denoising steps and get the average. To "deceive" the pretrained diffusion model, we propose to minimize the following formula:

$$\arg\min_{x_t} \mathcal{L}_{transfer} = \text{Var}(\text{Average}(\text{Cross}(x_t, t, C; \text{SDM}))) \tag{15}$$

where $\text{Var}(\cdot)$ calculates the input's variance, $\text{Cross}(\cdot)$ denotes the accumulation of all the cross-attention maps in the denoising process, and SDM is the Stable Diffusion. The insight is to distract the diffusion model's attention from the labeled objects. By evenly distributing attention to each pixel, we can disrupt the original semantic relationship, ensuring that our crafted adversarial examples well "deceive" the diffusion model. With such a design, *DiffAttack* exhibits an *implicit ensemble characteristic*. Note that it differs significantly from typical explicit ensemble attacks [53], about which we give a detailed analysis in Section IV-E.

E. Preserve Content Structure

As mentioned in Section III-C, our unrestricted attack can be approximated as an image editing approach, thus the change of the content structure is unavoidable. If the degree of the changes is not under control, the resulting adversarial examples may lose most semantics of the initial clean image (see Figure 6), which loses the significance of the adversarial attacks and is not what we want. Therefore, we here preserve the content structure mainly from two perspectives.

Self-Attention Control. Early work by [54] revealed that self-similarly-based descriptors can capture structural information while disregarding image appearance. In this study, we investigate whether the self-attention map within the U-Net structure of Stable Diffusion possesses similar structure extraction capabilities. Unlike cross attention (as illustrated in Eq. 14), self-attention maps are computed between the latent feature $\phi(x_t)$ and itself. Specifically, in the self-attention calculation, $W_K \in \mathbb{R}^{d_\phi \times d}$ and $K = \phi(x_t)W_K$, resulting in self-attention maps $S \in \mathbb{R}^{N\times N\times N^2}$. Examining several self-attention maps from the reconstruction process of the

inversed latent, Figure 3 reveals that self-attention in diffusion models also possesses inherent structure extraction properties. In contrast to cross attention, which focuses on high-level semantics, these self-attention maps excel at extracting the input image structure. Therefore, we propose to leverage the self-attention maps for structure retention. We set a copy $x_{t(fix)}$ of the inversed latent which is fixed without perturbations. By respectively calculating the self-attention maps (denoted as $S_{t(fix)}$ and S_t) of $x_{t(fix)}$ and x_t, we force S_t to get close to $S_{t(fix)}$ as follows:

$$\arg\min_{x_t} \mathcal{L}_{structure} = \|S_t - S_{t(fix)}\|_2^2 \tag{16}$$

Similar to Eq. 15, we here apply the self-attention constraint to all the denoising steps. Since $x_{t(fix)}$ reconstructs the initial clean image well [47], we can in this way preserve the structure.

Inversion Strength Trade-off. With DDIM Inversion strength increased, the latent x_t will get closer to pure Gaussian distribution and the perturbations on it may cause serious distortion due to influence on more denoising steps (see Figure 6). Whereas, a limited inversion cannot provide enough space for attacking, since the latent image prior is too strong. The inversion strength is a trade-off between imperceptibility and the attack success. Recent work [55] has found that the diffusion models tend to add coarse semantic information (*e.g.*, layout) in the early denoising steps while more fine details in the later steps. Thus, we control the inversion at the back of the denoising process for retention of high-level semantics, and reduce the total DDIM sample steps for more editing space.

Besides the above operations, we also adopt the approach of [46] to get a good initial reconstruction by optimizing unconditional embeddings. Details can be found in their source paper.

In general, the final objective function of *DiffAttack* is as follows, where α, β, and γ represent the weight factors of each loss:

$$\arg\min_{x_t} \mathcal{L} = \alpha\mathcal{L}_{attack} + \beta\mathcal{L}_{transfer} + \gamma\mathcal{L}_{structure} \tag{17}$$

IV. Experiments

A. Experimental Setup

Datasets. Following the previous methods [19], [29], we evaluate the performance of our attack on the development set of ImageNet-Compatible Dataset[1], which consists of 1,000 images with size 299×299×3. Considering that the Stable Diffusion cannot handle the original input size of the ImageNet-Compatible Dataset, we focused on a resized version of 224×224×3 in all the experiments. *DiffAttack* also generalizes well to other datasets. Please refer to Section IV-B5 where we conduct further experiments on CUB-200-2011 [56] and Stanford Cars [57].

Models. We evaluate the transferability of the attacks across a variety of network structures, including CNNs, Transformers, and MLPs. For CNNs, we adopt normally trained models including ConvNeXt [58], ResNet-50 (Res-50) [59], VGG-19 [60], Inception-v3 (Inc-v3) [61], and MobileNet-v2 (Mob-v2) [62]. For Transformers, we consider normally trained ViT-B/16 (ViT-B) [63], Swin-B [64], DeiT-B and DeiT-S [65]. For MLPs, we adopt normally trained Mixer-B/16 (Mix-B) and Mixer-L/16 (Mix-L) [66]. Furthermore, we also consider various defense methods, including DiffPure [27], SR [67], R&P [68], HGD [69], NIPS-r3 [70], NRP [28], and adversarially trained models (Adv-Inc-v3 [71], Inc-v3$_{ens3}$, Inc-v3$_{ens4}$, and IncRes-v2$_{ens}$ [53]).

Implementation Details. We leverage DDIM [47] as the sampler of the Stable Diffusion [30]. The number of steps is set to 20 and we apply 5 DDIM Inversion steps of the initial clean image. In the inversion process, the guidance scale is set to 0, while in the denoising process, we set it to 2.5. For optimizing the latent x_t, we adopt AdamW [72] with the learning rate set to $1e^{-2}$ and the iterations set to 30. The weight factors α, β, γ in Eq. 17 are set to 10, 10000, 100 respectively. All experiments are run on a single RTX 3090 GPU.

Evaluation Metrics. We adopt top-1 accuracy to evaluate the performance of the attack methods and leverage Frechet Inception Distance (FID) [73] as the indicator of the human imperceptibility of the crafted adversarial examples. A full-referenced metric, LPIPS [74], is also used to assess the perceptual differences.

B. Comparisons

1) Results on Normally Trained Models: Here, we compared the performance of *DiffAttack* on normally trained models with other transfer-based black-box attacks. We select four pixel-based attacks (DI-FGSM [20], TI-FGSM [21], PI-FGSM [75], S^{2}I-FGSM [19]) and three unrestricted attacks (ReColorAdv [76], cAdv [37], NCF [29]). Except that the resolution is changed to 224×224×3, the implementations of these methods follow their original optimal settings. All I-FGSM-based ones [20], [21], [75], [19] are constrained by L_{inf} with steps set to 10, maximum perturbation set to 16, and step size set to 1.6. For DI-FGSM, we set its transformation probability to 0.5. For TI-FGSM, we set its kernel size to 7. For PI-FGSM, we set its amplification factor to 10. For S^{2}I-FGSM, we set its inner iteration number to 20, its tuning factor to 0.5, and its standard deviation to 16. As for the unrestricted attacks [76], [37], [29], we set ReColorAdv's minimum and maximum iteration numbers to 50 and 100, respectively, and removed its constraint of upper bound to adapt it to an unrestricted attack. For cAdv, we set the cluster number to 8. For NCF, we set its random search number to 50, neighborhood search number to 15, reset number to 10, and step size to 0.013. We craft the adversarial examples via Res-50, VGG-19, Mov-v2, Inc-v3, ConvNeXt, and Swin-B (Performance on more surrogate models can be found in Appendix D). The transferability of different attack methods is displayed in Table I.

From the results, we can observe that *DiffAttack* can achieve the best transferability across a variety of model structures, while other unrestricted attacks (ReColorAdv, cAdv and NCF) usually fail to compete with pixel-based attacks. In some architectures such as VGG-19 and Mob-v2, our method can even outperform the second-best method by nearly 10 points

[1] https://github.com/cleverhans-lab/cleverhans/tree/master/cleverhans_v3.1.0/examples/nips17_adversarial_competition/dataset.

TABLE I

TRANSFERABILITY AND IMPERCEPTIBILITY COMPARISONS ON NORMALLY TRAINED MODELS. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. "S." DENOTES SURROGATE MODELS WHILE "T." DENOTES TARGET MODELS. FOR WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET), WE SET THE BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE. "FID" IS CALCULATED BETWEEN THE 1,000 IMAGES OF THE IMAGENET-COMPATIBLE DATASET WITH THE IMAGENET VALIDATION SET. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

S. \ T.	Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
		Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
	Clean	92.7	88.7	86.9	80.5	97.0	93.7	95.9	94.5	94.0	82.5	76.5	89.4	57.8	—
Res-50	DI-FGSM	0	21.2	20.5	34.5	71.6	82.0	75.3	80.5	76.0	61.3	56.8	58.0	85.3	0.143
	TI-FGSM	0	42.4	37.1	46.0	83.6	81.6	83.7	84.5	79.0	66.0	61.7	66.6	66.0	0.139
	PI-FGSM	0	14.1	15.0	24.0	72.5	65.3	77.5	76.7	65.0	50.5	43.8	50.5	97.9	0.356
	S^2I-FGSM	0	9.2	6.6	18.6	44.1	63.9	52.0	65.9	59.0	45.6	44.3	40.9	79.8	0.157
	ReColorAdv	0.8	42.4	36.2	52.1	71.4	68.3	71.9	73.2	67.7	63.5	59.2	60.6	63.4	0.154
	cAdv	2.4	51.8	42.5	53.2	79.0	63.1	75.3	68.3	62.3	51.7	44.0	54.0	65.7	0.186
	NCF	11.3	30.5	30.3	52.6	78.3	65.7	76.8	75.1	67.0	53.7	47.6	57.8	70.9	0.383
	DiffAttack(Ours)	3.7	24.4	22.9	31.0	41.0	48.8	43.8	49.5	45.0	42.9	42.2	**39.2**	**62.6**	**0.137**
VGG-19	DI-FGSM	32.2	0	23.9	46.5	67.2	84.7	71.9	84.8	80.0	65.7	60.9	61.8	70.9	0.152
	TI-FGSM	44.5	0	32.8	47.4	77.8	81.4	79.3	83.6	79.0	64.9	60.3	65.1	66.6	0.154
	PI-FGSM	22.7	0	16.4	29.8	68.3	68.0	75.7	79.5	68.0	50.9	41.8	52.1	96.4	0.367
	S^2I-FGSM	17.9	0	11.3	31.8	49.5	74.1	57.9	76.0	68.0	52.6	50.8	49.0	82.9	0.155
	ReColorAdv	41.4	0.4	35.5	54.3	74.1	73.2	72.3	79.6	71.3	65.5	61.3	62.9	64.3	0.174
	cAdv	49.0	2.5	39.9	49.2	77.4	61.9	70.4	68.6	60.1	51.5	44.4	57.2	69.8	0.234
	NCF	38.3	6.8	31.5	52.4	80.5	67.5	77.6	77.4	71.0	53.5	47.2	59.7	70.4	0.392
	DiffAttack(Ours)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	**38.2**	**63.9**	**0.150**
Mob-v2	DI-FGSM	28.7	18.9	0	33.9	73.4	79.9	71.4	79.6	75.0	57.7	57.1	57.6	78.6	0.141
	TI-FGSM	47.2	37.9	0	45.2	83.0	79.9	80.9	81.8	76.0	61.7	58.3	65.1	65.6	0.140
	PI-FGSM	21.1	13.3	0	27.6	74.4	65.3	77.0	77.4	66.0	49.7	41.5	51.4	98.7	0.367
	S^2I-FGSM	21.0	13.4	0	27.2	64.3	74.1	62.6	75.2	68.0	51.4	48.3	50.5	79.4	0.157
	ReColorAdv	39.6	39.7	0.2	51.2	74.8	67.2	69.9	74.6	66.3	62.5	58.6	60.4	63.3	0.157
	cAdv	49.5	47.3	3.4	50.5	78.3	60.2	72.3	69.2	60.7	52.1	43.5	58.4	68.6	0.211
	NCF	36.0	29.4	7.4	51.9	77.4	67.2	76.1	76.1	68.0	54.9	48.3	58.6	69.7	0.387
	DiffAttack(Ours)	23.6	23.4	1.8	31.6	50.3	51.4	45.8	53.4	46.0	38.5	40.8	**40.5**	**62.9**	**0.138**
Inc-v3	DI-FGSM	61.7	57.4	51.9	0.2	89.9	84.6	86.8	86.7	82.0	68.4	62.3	73.2	67.1	0.131
	TI-FGSM	76.0	70.1	66.7	0.1	93.8	88.7	91.2	89.7	88.0	73.8	66.8	80.5	62.8	0.129
	PI-FGSM	37.9	22.4	28.4	0	81.0	74.3	83.0	81.9	72.0	57.1	45.8	**58.4**	92.5	0.368
	S^2I-FGSM	52.3	47.8	43.3	0	86.3	80.8	84.1	83.8	78.0	63.5	57.3	67.8	72.5	0.137
	ReColorAdv	68.4	64.2	60.7	1.6	88.5	80.9	85.0	85.3	80.4	73.4	67.4	75.4	63.3	0.141
	cAdv	76.9	72.3	70.3	8.3	91.1	81.4	85.7	84.1	79.7	64.8	58.8	76.5	62.6	0.155
	NCF	52.6	45.8	46.2	17.4	85.7	75.9	83.4	82.7	76.0	61.1	52.9	66.2	66.7	0.343
	DiffAttack(Ours)	59.5	55.6	55.4	13.9	76.9	75.2	72.8	74.0	71.0	58.9	54.7	65.4	**62.3**	**0.126**
ConvNeXt	DI-FGSM	33.6	24.3	29.8	46.6	0	71.0	18.8	62.2	64.0	49.6	46.7	44.6	79.6	0.156
	TI-FGSM	50.7	37.3	41.1	51.8	0	70.9	38.8	68.6	69.0	52.3	47.2	52.7	73.5	0.158
	PI-FGSM	23.6	14.2	17.1	22.4	0	43.0	37.2	48.7	43.0	33.2	31.7	31.4	101.8	0.359
	S^2I-FGSM	13.6	9.6	11.9	20.2	0	35.4	4.2	31.0	31.0	23.2	25.6	**20.5**	99.4	0.159
	ReColorAdv	52.0	49.4	44.8	53.1	0.7	59.1	39.0	57.7	56.6	55.6	48.3	51.6	75.1	0.158
	cAdv	36.2	38.0	36.7	43.0	1.1	37.3	34.0	38.4	36.7	31.8	30.3	36.2	75.9	0.244
	NCF	47.1	41.4	39.2	54.7	41.4	61.6	63.9	64.8	62.0	52.2	47.8	53.5	**67.0**	0.360
	DiffAttack(Ours)	20.9	24.8	21.8	25.8	1.9	26.7	11.4	21.6	24.0	21.7	24.0	22.2	73.3	**0.154**
Swin-B	DI-FGSM	52.7	43.0	44.5	56.4	33.9	66.6	2.7	57.2	58.0	52.4	50.8	51.5	65.7	0.138
	TI-FGSM	71.9	61.7	56.9	60.2	66.0	76.3	1.9	72.2	72.0	61.2	56.9	65.6	65.9	0.142
	PI-FGSM	38.3	21.6	25.8	35.7	54.8	48.4	0.6	52.4	47.0	43.5	38.5	40.6	89.7	0.358
	S^2I-FGSM	47.4	37.8	35.4	45.3	26.8	48.5	1.0	46.2	45.0	39.3	39.0	41.1	68.2	**0.134**
	ReColorAdv	63.5	61.4	57.9	62.4	62.8	64.8	0.9	62.7	64.8	60.8	55.2	61.6	65.7	0.147
	cAdv	59.1	59.1	53.9	56.8	62.4	48.3	1.6	46.6	46.6	46.0	41.1	52.0	67.4	0.191
	NCF	49.5	44.9	44.9	60.5	70.1	63.7	36.9	66.0	63.0	51.7	49.1	56.3	**65.5**	0.346
	DiffAttack(Ours)	43.5	42.1	40.7	41.4	34.0	39.0	9.9	35.0	37.0	37.7	37.4	**38.8**	**65.5**	0.138

(38.2% vs. 49.0%, 40.5% vs. 50.5%). While our method may not surpass PI-FGSM (S^2I-FGSM) in attack performance under the Inc-v3 (ConvNeXt) structure, the FID and LPIPS results reveal a substantial advantage over these methods. Specifically, our approach exhibits scores more than 20 points lower on FID compared to PI-FGSM and S^2I-FGSM.

Regarding the imperceptibility of the crafted adversarial examples, our method consistently achieves the best performance. In Figure 4, we visualize adversarial examples crafted by different attack approaches. It's evident that our attack is significantly more imperceptible compared to DI-FGSM, TI-FGSM, PI-FGSM, and S^2I-FGSM, which exhibit noticeable high-frequency noise. Furthermore, compared to ReColorAdv, cAdv, and NCF, *DiffAttack* demonstrates a more natural color space. These observations confirm the superiority of our method. Additional visualizations can be found in Appendix H.

2) Results on Defense Approaches: To further verify the robustness of each attack method, we evaluate the performance of the crafted adversarial examples on defense approaches. Following [29], [19], we consider both input preprocessing defenses [67], [68], [69], [70], [28] and adversarially trained models [71], [53] (see Section IV-A). We further consider the recent DiffPure defense [27] to better demonstrate our superiority. We take Inc-v3 as an example surrogate model

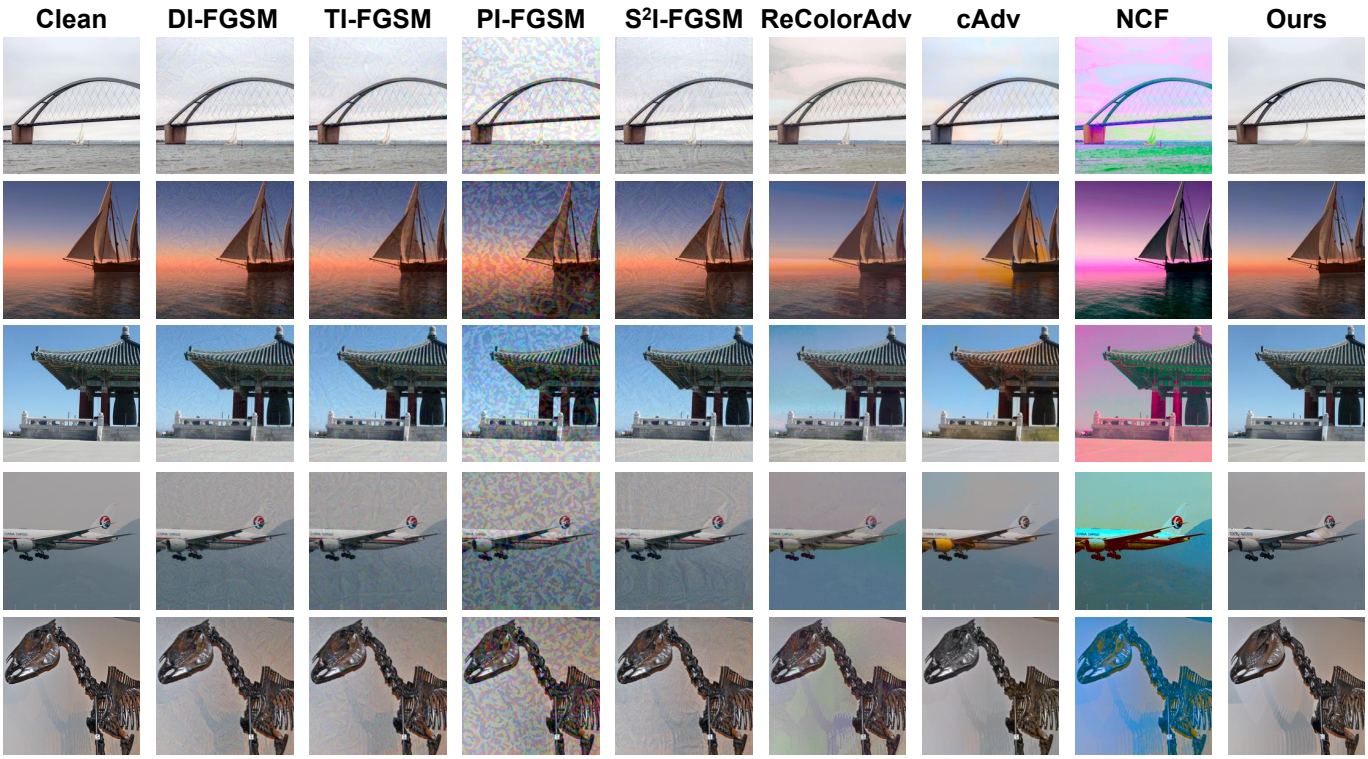

Fig. 4. **Visual comparisons among different attacks.** Please zoom in for a better view.

TABLE II

ROBUSTNESS ON DEFENSE APPROACHES. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. "A." DENOTES ATTACK METHODS WHILE "D." DENOTES DEFENSE APPROACHES. "INC-V3$_{normal}$" DENOTES THE ACCURACY ON NORMALLY TRAINED INC-V3. FOR SR, NRP, AND DIFFPURE, WE DISPLAY THE ACCURACY DIFFERENCES AFTER THE DEFENSE. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

A. \ D.	HGD	R&P	NIP-r3	Adv-Inc-v3	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$	Inc-v3$_{normal}$	SR	NRP	DiffPure
DI-FGSM	80.5	83.8	79.4	64.2	58.5	61.5	74.9	0.2	+42.9	+22.9	+52.4
TI-FGSM	84.7	86.1	87.0	66.1	62.4	64.5	76.8	0.1	+59.6	+25.8	+55.5
PI-FGSM	73.4	68.6	**57.2**	**42.3**	45.0	44.6	62.0	0	+44.2	+7.7	+21.5
S^{2}I-FGSM	72.5	76.5	73.3	51.8	47.0	52.2	67.7	0	+47.1	+3.2	+47.0
ReColorAdv	89.1	91.9	88.4	70.0	67.4	67.5	81.2	1.6	+67.5	+41.0	+47.8
cAdv	87.4	88.3	83.1	69.0	62.6	63.6	76.8	8.3	+61.8	+38.3	+39.2
NCF	71.1	66.4	74.6	48.8	47.2	49.0	60.5	17.4	+33.9	+11.0	+14.8
DiffAttack(Ours)	**62.0**	**65.5**	70.0	46.0	**43.8**	**43.1**	**58.3**	13.9	**+28.5**	**+2.3**	**+13.9**

Table values represent the top-1 classification accuracy of the classifier, where lower values indicate superior attack performance against defensive approaches.

and all the adversarial examples are crafted from it. For SR, NRP and DiffPure, we set the target model as Inc-v3 itself, thus better revealing the robustness. For other defenses, the target models are the same as the official papers. We display the results in Table II.

From the results, we can see that our method can achieve good robustness and outperform other methods when defenses are applied. For the adversarial purification defenses, it can be seen that the attack success of our attack has the least change compared with other ones, which does verify the robustness of *DiffAttack* and the effectiveness of our designs.

3) Comparisons with GAN-Based Attack Methods: The attacks compared in Table I are all iterative approaches, involving multiple optimization steps for generating adversarial perturbations. However, another category of attacks, known as GAN-based attacks [77], takes a different approach. Instead of directly optimizing perturbations, these attacks focus on training a GAN generator to produce the final perturbation. Considering that both GAN-based attacks and our *DiffAttack* utilize generative models (although *DiffAttack* fundamentally follows an iterative optimization approach), we undertake a comprehensive comparison with these GAN-based attacks. This expansion aims to enhance the inclusivity of our experiments and further underscore the advantages of *DiffAttack*.

Here, we consider four GAN-based attacks: GAP [77], CDA [36], BIA [78], and TSAA [79]. All these compared methods have their code open-source and our experiments are based on that. For BIA [78], we directly use their provided pretrained generator (for VGG-19) to generate adversarial examples. The variants of it (BIA+DA and BIA+RN) have

TABLE III

COMPARISONS WITH GAN-BASED ATTACK METHODS. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. WE CRAFT ADVERSARIAL EXAMPLES EITHER ON VGG-19 OR RES-50. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
clean	92.7	88.7	86.9	80.5	97	93.7	95.9	94.5	94	82.5	76.5	89.4	57.8	—
GAP (universal)	56.9	12.4	20.6	56.9	92.2	92.1	91.3	91.1	88.0	65.1	57.5	71.2	100.6	0.178
GAP(image dependent)	70.1	9.5	35.6	60.2	79.4	89.6	89.1	88.6	83.6	66.1	55.2	71.8	108.0	0.164
CDA	23.0	0.2	16.5	48.6	45.2	89.1	80.7	86.0	82.4	62.7	54.1	58.8	131.8	0.174
BIA	25.2	1.8	10.5	38.6	58.8	83.2	75.6	82.9	80.3	54.3	47.6	55.7	200.3	0.252
BIA+DA	16.3	1.6	7.6	33.1	44.0	85.1	74.8	84.5	80.4	55.7	49.8	53.1	246.0	0.247
BIA+RN	14.7	1.4	5.8	28.6	52.2	79.7	70.0	80.7	77.4	48.9	44.1	50.2	246.7	0.275
DiffAttack(Ours)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	**38.2**	**63.9**	**0.150**

Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
clean	92.7	88.7	86.9	80.5	97	93.7	95.9	94.5	94	82.5	76.5	89.4	57.8	—
GAP (universal)	35.6	34.0	34.3	55.3	88.8	87.0	91.9	91.4	85.8	64.7	57.4	69.1	89.7	0.248
GAP(image dependent)	42.9	21.7	25.4	55.6	87.1	88.5	89.5	88.1	84.4	62.3	55.8	65.8	102.6	0.147
TSAA (dense)	15.4	16.4	22.2	52.0	74.3	87.4	89.2	90.4	86.3	66.6	61.8	64.7	105.6	0.261
DiffAttack(Ours)	3.7	24.4	22.9	31.0	41.0	48.8	43.8	49.5	45.0	42.9	42.2	**39.2**	**62.6**	**0.137**

TABLE IV

COMPARISON WITH THE COMBINATION OF MULTIPLE ATTACK APPROACHES. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED.

Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
S^2I-FGSM	17.9	0	11.3	31.8	49.5	74.1	57.9	76.0	68.0	52.6	50.8	49.0	82.9	0.155
S^2I-MI-FGSM	6.2	0	3.6	14.5	30.1	51.4	41.1	54.3	45.7	34.5	33.0	31.4	100.0	0.286
S^2I-DI-MI-FGSM	3.6	0	2.3	9.2	24.6	44.7	33.3	49.2	38.2	28.2	29.1	26.2	104.5	0.295
S^2I-TI-DI-MI-FGSM	5.2	0	3.1	7.8	40.0	35.9	46.0	49.3	36.8	27.5	27.1	27.9	104.9	0.310
S^2I-SI-TI-DI-MI-FGSM	5.5	0	4.1	7.7	45.4	34.4	47.5	49.5	36.4	27.0	26.3	28.4	114.7	0.299
DiffAttack(Ours)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	38.2	**63.9**	**0.150**
DiffAttack(w/o Structure Controls)	19.7	3.9	15.5	19.9	32.2	35.0	28.8	30.8	30.1	20.7	21.8	**25.5**	96.2	0.279

Fig. 5. **Visualization Comparisons with GAN-Based Attacks.** Please zoom in for a better view.

also been considered for comparisons. For CDA [36], we utilize their pretrained generator (for VGG-19) to generate adversarial examples. We also take recent TSAA [79] into account. Considering the original TSAA is a sparse attack, we directly remove its last layer's mask mechanism to allow it to attack the whole image. As TSAA does not provide pretrained weight for VGG-19 but provides for Res-50, we compare *DiffAttack* with it on Res-50. For GAP [77], since it does not provide any pretrained weight, we strictly follow their provided training code and train the generator for VGG-19 and Res-50 ourselves. As GAP has two kinds of generator (universal and image dependent), we trained a total of four generators. All of these methods' maximum perturbation is set to 10, which is aligned with their source paper (we also tried 16, but it will massively distort the image and lead to a quite high FID and LPIPS). The input resolution of these methods is $224\times224\times3$, which also strictly follows their papers and is the same as our previous settings.

From the results in Table III, it is amazing to see that our *DiffAttack* can surpass other GAN-based attacks by a large margin on transferability (AVG w/o self), while also keeping quite better imperceptibility (FID and LPIPS). The GAN-based attacks tend to produce more distorted images, as evidenced by their higher FID and LPIPS values. In Figure 5, we visualize adversarial examples generated by GAN-based methods. We observe that these methods create perturbations with more

TABLE V

COMPARISONS ON CUB-200-2011 DATASET AND STANFORD CARS DATASET. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. "S." DENOTES SURROGATE MODELS WHILE "T." DENOTES TARGET MODELS. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

		CUB-200-2011						Stanford Cars					
S. \ T.	Attacks	Res-50	SENet154	SE-Res101	AVG (w/o self)↓	FID↓	LPIPS↓	Res-50	SENet154	SE-Res101	AVG (w/o self)↓	FID↓	LPIPS↓
	clean	75.7	80.5	76.6	77.6	11.1	-	73.9	76.4	74.4	74.9	11.6	-
Res-50	DI-FGSM	0.3	42.7	33.8	38.3	20.9	0.155	0.1	33.3	29.3	31.3	28.7	**0.097**
	TI-FGSM	2.8	50.6	43.9	47.3	21.1	0.136	0.1	46.9	41.0	44.0	23.2	**0.097**
	PI-FGSM	9.1	35.2	26.2	30.7	34.8	0.355	1.5	31.5	23.2	27.4	53.2	0.310
	S^2I-FGSM	0.7	35.1	28.1	31.6	24.3	0.196	0.1	25.7	24.4	25.1	34.4	0.134
	ReColorAdv	0.1	42.0	33.4	37.7	23.2	0.215	0.0	42.6	35.1	38.9	22.9	0.164
	cAdv	25.0	40.0	36.3	38.2	21.3	0.129	38.1	64.7	60.9	62.8	19.7	0.117
	NCF	0.2	22.7	13.9	18.3	35.2	0.335	6.6	46.0	38.4	42.2	24.1	0.302
	DiffAttack(Ours)	3.3	19.3	16.7	**18.0**	**20.6**	**0.122**	0.1	15.1	13.1	**14.1**	**17.8**	0.112
SENet154	DI-FGSM	54.5	0.2	48.9	51.7	23.5	0.158	45.6	0.1	45.5	45.6	29.1	0.096
	TI-FGSM	60.1	0.3	56.2	58.1	20.8	0.137	54.1	0.1	53.2	53.7	23.0	**0.095**
	PI-FGSM	30.5	0.0	33.1	31.8	46.5	0.403	21.9	0.0	26.3	**24.1**	59.6	0.333
	S^2I-FGSM	43.2	0.0	34.0	38.6	25.4	0.164	27.7	0.0	25.7	26.7	33.6	0.108
	ReColorAdv	55.2	4.3	48.9	52.1	22.4	0.153	44.7	0.0	42.9	43.8	21.3	0.130
	cAdv	31.0	5.7	31.3	31.2	20.4	0.118	63.3	20.2	60.1	61.7	17.8	0.102
	NCF	13.5	6.8	17.6	**15.5**	35.0	0.314	38.5	20.7	41.6	40.1	23.3	0.279
	DiffAttack(Ours)	53.8	2.5	51.3	52.6	**17.9**	**0.104**	37.3	0.9	32.5	34.9	**16.2**	**0.095**
SE-Res101	DI-FGSM	39.4	38.0	0.2	38.7	23.5	0.165	28.1	29.3	0.2	28.7	28.5	0.106
	TI-FGSM	53.4	55.3	0.2	54.4	**21.8**	0.136	48.7	49.8	0.0	49.3	22.5	**0.096**
	PI-FGSM	21.7	29.8	0.0	25.8	45.5	0.403	18.5	29.3	0.0	23.9	59.9	0.331
	S^2I-FGSM	30.4	31.5	0.0	31.0	26.7	0.195	20.5	17.1	0.1	18.8	36.9	0.142
	ReColorAdv	41.6	42.2	0.6	41.9	27.0	0.198	36.8	41.5	0.0	39.2	23.2	0.155
	cAdv	33.1	33.8	10.5	33.5	22.7	0.125	62.4	62.7	22.5	62.6	18.5	0.119
	NCF	9.4	20.2	3.1	**14.8**	33.3	0.316	33.4	46.8	12.1	40.1	24.0	0.298
	DiffAttack(Ours)	27.0	23.5	3.9	25.3	22.4	**0.121**	17.5	16.0	0.3	**16.8**	**18.0**	0.114

noticeable patterns compared to those produced by iterative optimization methods, as shown in Figure 1. This difference may stem from the nature of the GAN-based approach, which requires a generator capable of attacking random input images in a single step, as opposed to iterative optimization attacks that can fine-tune on a specific input image over multiple steps. Consequently, to achieve effective attacks, the GAN-based generator must introduce more distortions to the input image. These experiments not only enhance the comprehensiveness of our findings but also underscore the effectiveness of *DiffAttack*.

4) Comparisons with a Combination of Multiple Attack Approaches: Many recent L_p-norm-based attacks enhance their efficacy by combining with other attack strategies. For instance, the S^2I-SI-TI-DIM [19] approach integrates five attack methods (MI-FGSM [16], DI-FGSM[20], TI-FGSM[21], SI-FGSM[15], and their own S^2I-FGSM). While it is unfair to compare a single *DiffAttack* against an ensemble of these attack strategies, we still perform such comparisons in Table IV to better elucidate the capabilities of *DiffAttack*. The adversarial examples are crafted on VGG-19, with the powerful S^2I-SI-TI-DIM attack serving as the reference.

The results indicate that S^2I-based methods exhibit improved transferability when combined with other attacks, albeit at the cost of increased distortion. Our original *DiffAttack* cannot surpass the performance achieved by the combination of multiple attack approaches. Nevertheless, when structural controls are eliminated (as discussed in Section III-E) to align the FID and LPIPS values for fair comparisons, *DiffAttack* once again showcases superior performance.

5) Performance on More Datasets: In Section IV-B1, our comparative experiments are exclusively conducted on the ImageNet-Compatible Dataset. To bolster the credibility of *DiffAttack*'s performance and its applicability, we have expanded our evaluation to encompass two additional datasets: CUB-200-2011 [56] and Stanford Cars [57]. Aligning with the ImageNet-Compatible dataset, we randomly selected 1,000 samples from both the CUB-200-2011 and Stanford Cars datasets, respectively, for crafting adversarial examples. For normally trained models, we employed three models: ResNet50 (R-50) [59], SENet154 (S-154), and SE-ResNet101 (SR-101) [80], all initialized with pretrained weights provided by [78]. The results in Table V highlight *DiffAttack*'s strong generalization across diverse datasets.

It is also worth noting that color-based unrestricted attacks like NCF [29] achieve significantly higher attack success rates on the CUB-200-2011 dataset compared to the other two datasets (ImageNet and Stanford Cars). Upon analysis, we found that this discrepancy is primarily due to the fact that many bird categories in CUB-200-2011 are distinguished by color, such as the "Clay-colored Sparrow" and "Black-throated Sparrow". Consequently, when the NCF attack modifies the colors of clean images, it alters the birds' ground-truth attributes, inflating the observed attack success rates.

Additionally, to further verify the generalization of our method to new datasets and model types, we present further explorations in Appendix E.

C. Ablation Studies

1) Design Ablation: In Table VI, we ablate the designs mentioned in Section III-D. The adversarial examples are crafted on Inc-v3. We can observe that with the loss in Eq. 15 added, the attack success improves, verifying our design's

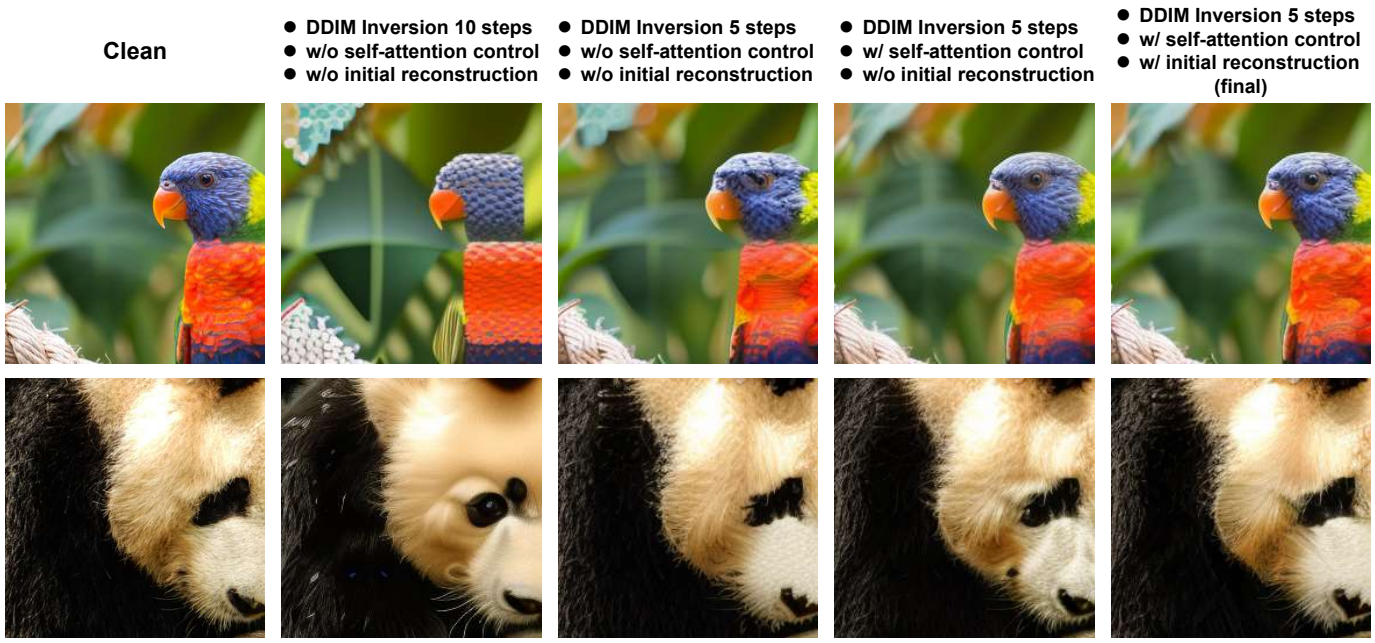

Fig. 6. **Visualization of design ablations for imperceptibility.** Please zoom in for a better view.

TABLE VI
ABLATION STUDY OF DESIGNS FOR TRANSFERABILITY.

Prompt Guidance	Diffusion Deception ($\mathcal{L}_{transfer}$)	AVG↓ (w/o self)
✗	✗	70.0
✓	✗	66.5
✓	✓	**65.4**

TABLE VII
ABLATION STUDY OF DESIGNS FOR IMPERCEPTIBILITY.

DDIM Inversion Step Number	Self-Attention Control ($\mathcal{L}_{structure}$)	Initial Reconstruction	FID↓	LPIPS↓
10	✗	✗	97.9	0.372
5	✗	✗	66.7	0.142
5	✓	✗	63.5	0.131
5	✓	✓	**62.3**	**0.126**

effectiveness. It can also be noted that prompt guidance is important for transferability, which we attribute to the fact that prompts can help guide the attack on the target objects. Results in Table VII verify the effectiveness of our designs for structure retention. With the inversion strength and self-attention controlled, the FID and LPIPS results gradually improve. We also visualize the structure ablation in Figure 6, which can display the visual improvement obviously. It can be seen that the control of inversion strength helps a lot preserve the structure, and the usage of self-attention maps can ensure better texture.

Moreover, as highlighted in Section I, the transferability of *DiffAttack* is not solely attributed to $L_{transfer}$, but also originates from our latent space perturbation and the denoising process intrinsic to the diffusion model itself. In other words, the diffusion model's structure and mechanisms inherently contribute to improving transferability.

To empirically validate this point, we conducted an ablation study by eliminating the diffusion model and directly perturbing the image pixels. This resulted in a pixel-based attack similar to I-FGSM [81]. We aligned the number of iterations with *DiffAttack* and, to mitigate the generation of unnatural high-frequency noise inherent in pixel-based attacks (as illustrated in Figure 1), we adopted settings from L_p-norm-based transferable attacks, limiting the maximum perturbation to 16. To effectively illustrate the influence of the diffusion model itself on transferability, particularly the latent space perturbation and denoising process, we compared this modified degradation model with an adapted *DiffAttack* (without $L_{transfer}$). We evaluated performance on both normally trained models and four defensive models (Adv-Inc-v3, Inc-v3$_{ens3}$, Inc-v3$_{ens4}$, and IncRes-v2$_{ens}$), yielding the results in Table VIII.

The presented results demonstrate that the diffusion model itself can enhance the transferability of adversarial examples, not only on traditionally trained models but also on defensive models. This strongly supports our assertion that the latent space perturbation and defensive denoising process in *DiffAttack* contribute to improved transferability. Additionally, with the diffusion model, the adversarial examples exhibit lower perceptibility (as indicated by FID and LPIPS scores), further substantiating the motivations outlined in Section I and reinforcing the contributions of our work.

2) Parameter Settings: We here reveal more experimental results about the parameter settings.

Settings of Guidance Scale. From Figure 7, it can be observed that with the guidance scale increased, the transferability improves while the imperceptibility deteriorates. We infer this is because larger guidance scales will tend to change the latent more and thus potentially generate more perturbations. Since there is a large gap in the attack success between the guidance

TABLE VIII

DEMONSTRATION OF THE EFFECT OF THE DIFFUSION MODEL ITSELF IN ENHANCING TRANSFERABILITY. WE REPORT TOP-1 ACCURACY(%). WE CRAFT ADVERSARIAL EXAMPLES ON INC-V3. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONES THAT HAVE A GRAY BACKGROUND. THE BEST RESULT IS BOLDED. THE FIRST TABLE DISPLAYS THE PERFORMANCE ON NORMALLY TRAINED MODELS, WHILE THE SECOND ONE ON DEFENSIVE MODELS.

Ablation	CNNs					Transformers				MLPs		AVG↓ (w/o self)
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L	
w/o Diffusion Model	62.5	60.3	56.9	0	88.3	85.9	87.6	88.2	84.8	68.0	62.8	74.5
w/o $L_{transfer}$	60.6	59.2	57.4	10.9	77.9	75.1	74.4	75.2	71.9	58.6	54.7	**66.5**

Ablation	Defensive Models				AVG↓	FID↓	LPIPS↓
	Adv-Inc-v3	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$			
w/o Diffusion Model	66.4	62.7	63.7	79.1	68.0	69.2	0.154
w/o $L_{transfer}$	45.0	43.0	42.3	57.1	**46.9**	**61.6**	**0.125**

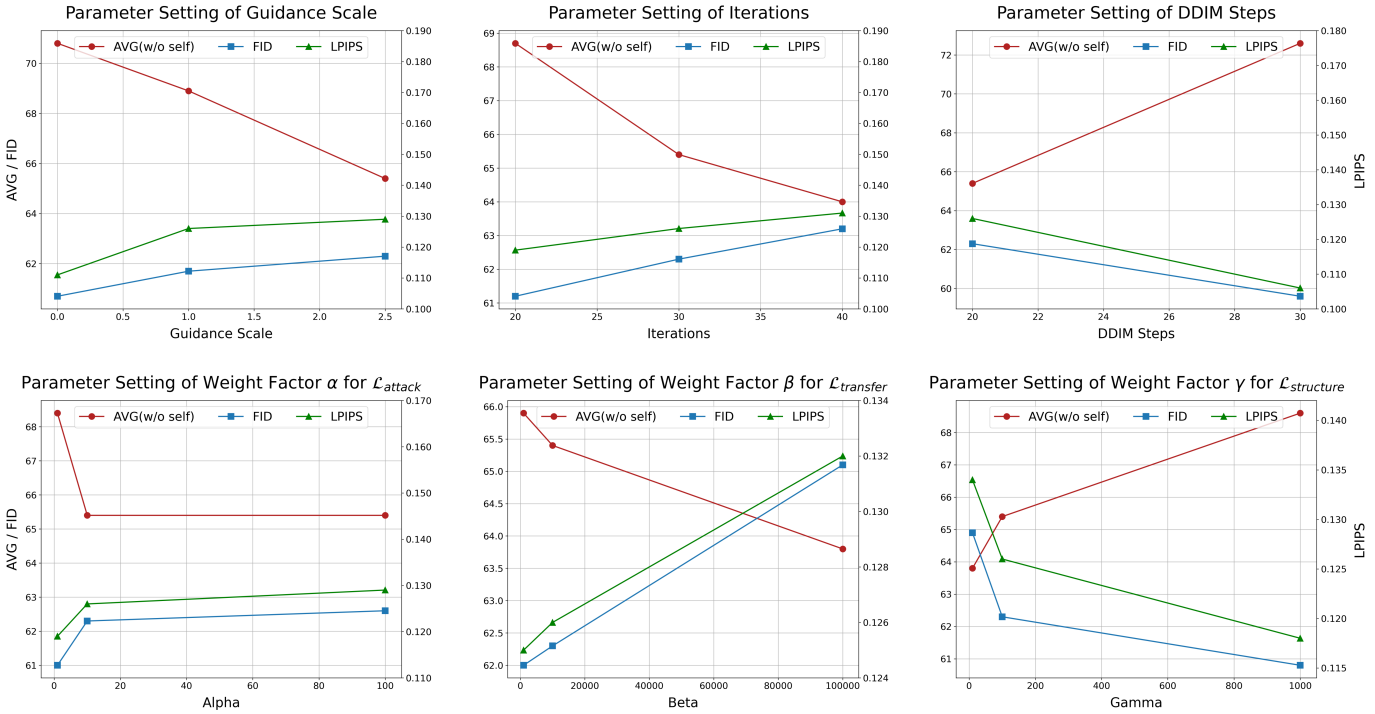

Fig. 7. **The effect of different parameter settings.** We conduct a quantitative study on the parameter settings of the guidance scale, iterations, DDIM steps, and weight factors of each loss. "AVG(w/o self)" denotes the average accuracy on all the target models except the one that same as the surrogate one.

scale set to 1.0 and 2.5, but a slight change of the FID and LPIPS value, we set the guidance scale to 2.5 finally.

Settings of Iterations. We can notice from Figure 7 that more iterations will sacrifice image quality for the attack success. As more iterations will consume longer optimization time, we here set the number of iterations to 30, which strikes a balance between time-consuming, image quality, and attack robustness.

Settings of DDIM Steps. In Figure 7, we keep the DDIM Inversion steps the same (5 inversion steps), to see the effect of different DDIM full sample steps. We do not show here the results for the step number set to 10 because the image quality is rather poor and the structure is completely changed. From the results, we can see that the step number does impact a lot both the transferability and the imperceptibility. Here we set the number of DDIM sample steps to 20, which can produce perceptually invisible adversarial samples with stronger attack robustness.

Settings of Weight Factor for Loss. We also conduct quantitative studies on the weight factor settings in Eq. 17. From Figure 7, it can be noticed that our designs of $\mathcal{L}_{transfer}$ and $\mathcal{L}_{structure}$ do make sense for improving the attack's transferability and preserving the content structure. For $\mathcal{L}_{attack}$, we can see from the results that there is a negligible performance improvement when α is increased to a certain extent, thus we set α to 10. For $\mathcal{L}_{transfer}$ and $\mathcal{L}_{structure}$, to balance both the transferability and the imperceptibility, we set them to 10000 and 100 respectively.

D. Exploration of Perturbation on Guided Text Embeddings

As mentioned in Section III-C, we choose to perturb the latent x_t but not the guided text C, which is different from

TABLE IX

COMPARISONS OF PERTURBATIONS ON THE LATENT AND TEXT. "S." DENOTES SURROGATE MODELS WHILE "T." DENOTES TARGET MODELS. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED.

S. \ T.	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
Clean	92.7	88.7	86.9	80.5	97.0	93.7	95.9	94.5	94.0	82.5	76.5	89.4	57.8	—
Text Perturbation	79.6	73.3	74.7	**13.4**	91.3	85.9	86.9	87.9	86.0	71.6	63.8	80.1	**58.8**	**0.108**
Latent Perturbation	**59.5**	**55.6**	**55.4**	13.9	**76.9**	**75.2**	**72.8**	**74.0**	**71.0**	**58.9**	**54.7**	**65.4**	62.3	0.126

TABLE X

COMPARISONS WITH EXPLICIT ENSEMBLE ATTACKS USING A ZERO-SHOT CLIP CLASSIFIER. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. WE CRAFT ADVERSARIAL EXAMPLES ON VGG-19 AND CLIP. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
PI-FGSM (VGG-19)	22.7	0	16.4	29.8	68.3	68.0	75.7	79.5	67.6	50.9	41.8	52.1	96.4	0.367
PI-FGSM (VGG-19+CLIP)	40.2	21.6	26.2	33.5	79.5	57.1	78.6	71.1	59.7	49.0	39.2	53.4	89.5	0.359
S^2I-FGSM (VGG-19)	17.9	0.0	11.3	31.8	49.5	74.1	57.9	76.0	68.0	52.6	50.8	49.0	82.9	0.155
S^2I-FGSM (VGG-19+CLIP)	16.1	0.4	9.6	26.4	46.8	58.8	50.3	63.2	56.3	44.5	42.3	41.4	84.6	0.165
NCF (VGG-19)	38.3	6.8	31.5	52.4	80.5	67.5	77.6	77.4	70.6	53.5	47.2	59.7	70.4	0.392
NCF (VGG-19+CLIP)	39.9	9.9	32.0	53.7	79.3	66.2	78.5	77.5	68.4	54.1	48.4	59.8	70.4	0.384
DiffAttack(VGG-19)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	38.2	**63.9**	**0.150**
DiffAttack (VGG-19+CLIP, w/o $L_{transfer}$)	27.2	10.0	24.1	29.4	44.1	46.1	41.5	45.1	39.7	38.7	36.9	**37.3**	64.6	0.151

TABLE XI

LEVERAGING DIFFATTACK FOR ENSEMBLE ATTACKS. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. WE CRAFT ADVERSARIAL EXAMPLES ON VGG-19 AND CLIP. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONES THAT HAVE A GRAY BACKGROUND. THE BEST RESULT IS BOLDED.

Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
S^2I-FGSM(VGG-19+Res-50)	1.5	0.0	4.5	14.8	28.8	57.7	40.4	61.0	54.0	41.2	39.2	38.0	84.7	0.163
DiffAttack(VGG-19)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	40.1	63.9	**0.150**
DiffAttack(VGG-19+Res-50,w/o $L_{transfer}$)	3.8	3.8	11.6	20.0	24.0	36.3	26.3	34.0	30.6	30.9	30.5	**27.1**	**62.1**	0.151
S^2I-FGSM(VGG-19+Swin-B)	14.5	0.3	9.5	25.8	27.2	51.3	15.4	52.5	48.1	41.8	39.0	34.4	83.1	0.152
DiffAttack(VGG-19)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	37.8	**63.9**	**0.150**
DiffAttack(VGG-19+Swin-B,w/o $L_{transfer}$)	19.3	6.9	19.4	26.0	27.6	33.5	15.6	30.4	30.1	30.7	31.4	**27.6**	64.2	0.151
S^2I-FGSM(VGG-19+Mix-L)	17.5	0.3	12.0	27.9	43.8	58.0	46.2	56.1	51.4	24.6	10.8	37.5	83.9	0.156
DiffAttack(VGG-19)	21.1	2.7	19.4	29.7	43.1	52.9	41.6	51.3	45.0	39.6	38.5	38.2	**63.9**	**0.150**
DiffAttack(VGG-19+Mix-L,w/o $L_{transfer}$)	22.7	4.2	20.5	31.2	40.7	43.9	36.8	43.1	40.4	27.5	25.8	**34.1**	64.3	0.153
S^2I-FGSM(Res-50+ViT-B)	1.1	7.7	5.9	17.5	39.3	10.9	40.5	26.7	20.3	27.2	29.6	**23.9**	79.6	0.165
DiffAttack(Res-50)	3.7	24.4	22.9	31.0	41.0	48.8	43.8	49.5	45.0	42.9	42.2	38.1	**62.6**	**0.137**
DiffAttack(Res-50+ViT-B,w/o $L_{transfer}$)	6.3	18.8	18.7	24.6	27.7	12.9	26.4	24.2	21.1	26.5	27.8	24.0	63.6	0.150

the mainstream image editing approaches [50], [46], [45]. The reason is that text perturbation will be hard to transfer to other black-box models. In the following, we display the details of text perturbation designs and some necessary experiments and analyses.

1) Design Details: Here we first define two text prompts: C_1, C_2, which are the first and second most possible categories predicted by the classifier. We leverage C_1 for the optimization of unconditional embeddings mentioned in Section III-E. Then, we replace C_1 with C_2 which follows [46], [52] and can expect the changes of object semantics in the image. For the loss functions, we remove $\mathcal{L}_{transfer}$ in Eq. 17, and modify $\mathcal{L}_{attack}$ as follows:

$$\arg\min_{C_2} \mathcal{L}_{attack} = J(x', C_2; G_\phi) \tag{18}$$

The equation above is similar to the objective function of targeted attacks, and the insight is to trick the classifier into predicting the nearest wrong label. Other implementation details are the same as Section IV-A.

2) Experiments and Analysis: In this subsection, we compare the results between the text perturbation and the latent perturbation. From Table IX, we can observe that although the text perturbation has a slightly higher attack success in a white-box way (0.5 point accuracy lower on Inc-v3), the attack itself is hard to work on the other black-box models, thus not competitive with the latent perturbations. We attribute this phenomenon to the fact that the text perturbation is more high-level than the latent perturbation, due to text semantics. Therefore, it will tend to generate more realistic results (lower FID and LPIPS in Table IX), but has limited control over the local area, while the latent perturbation does the opposite.

E. Discussions about DiffAttack's Relationship with Ensemble Attacks

1) DiffAttack as an "Implicit" Ensemble Attack: DiffAttack can be considered as an "Implicit" ensemble attack. The loss function $\mathcal{L}_{transfer}$ in Eq. 15 functions to divert the intermediate 2D cross-attention maps. This resembles the role of a zero-shot

CLIP classifier [82], which aims to align the image's features with its corresponding text embedding. From this perspective, *DiffAttack* can be viewed as an ensemble adversarial attack, targeting both a zero-shot CLIP classifier and a surrogate classifier.

However, it's essential to highlight that, unlike explicit ensemble attacks involving multiple surrogate models behind the final output adversarial examples [53], *DiffAttack*'s ensemble characteristic is "implicit". $\mathcal{L}_{transfer}$ is designed to perturb the intermediate 2D cross-attention maps of the diffusion model rather than attacking the final similarity results of an explicit CLIP classifier. This design avoids the need for an additional image classifier to generate adversarial examples, resulting in no additional memory overhead.

In summary, *DiffAttack* exhibits an **"implicit ensemble characteristic"** but differs significantly from typical explicit ensemble attacks.

2) Comparisons with Explicit Ensemble Attacks Using a Zero-shot CLIP Classifier: To ensure the comprehensiveness of our experiments, we have included comparisons with ensemble attacks employing an additional explicit zero-shot CLIP classifier. Also, we adapted the original *DiffAttack* into an explicit ensemble attack by substituting $\mathcal{L}_{transfer}$ with an explicit CLIP surrogate model.

We display the compared results in Table X. The base surrogate model is VGG-19 and we consider comparisons with three recent attack methods [75], [19], [29]. For the zero-shot CLIP classifier, we utilized the pretrained ViT-B/32 weights provided by OpenAI. Based on the results obtained, our original *DiffAttack* consistently outperforms other methods in terms of both transferability and imperceptibility, even when those methods attack an additional CLIP classifier. As for our adapted ensemble *DiffAttack*, which replaces $\mathcal{L}_{transfer}$ with an explicit CLIP classifier, we observed an improvement in transferability but a reduction in imperceptibility. It's worth noting again that, unlike the explicit CLIP classifier, $\mathcal{L}_{transfer}$ utilizes intermediate cross-attention maps during the denoising process, incurring no additional memory costs.

3) Leveraging DiffAttack for Ensemble Attacks: Here, we unveil another remarkable potential of diffusion models in crafting adversarial examples: **Ensemble attacks founded on diffusion models can significantly outperform conventional ensemble attacks** [53].

To demonstrate this, we conducted a comparison between *DiffAttack* and L_p-norm-based attacks involving multiple surrogate models, using S^2I-FGSM [19] as an example. Adversarial examples were generated to target various model structures.

The results in Table XI indicate that our original *DiffAttack*, which targets a single model structure, falls short when compared to ensemble attacks that target two model structures explicitly. The reason is evident: when more model structures are explicitly attacked, the generated adversarial examples exhibit superior transferability across these surrogate structures. It's important to note that the diffusion model [30] we employ, designed initially for image synthesis, fundamentally serves as an "implicit" recognition model. Therefore, our deception loss $L_{transfer}$ cannot be designed in the same manner as commonly used attack losses (See L_{attack} in Eq. 12) that directly target the classifier's decision (the ultimate goal of the attack). This limitation explains the original *DiffAttack*'s inability to outperform ensemble attacks in terms of transferability, although it still maintains superior imperceptibility.

However, when we employed an explicit ensemble attack based on *DiffAttack*, while also removing $L_{transfer}$ for fairness, *DiffAttack* achieved better (or competitive) results in both transferability and imperceptibility, as evident in Table XI. These findings underscore the potential of diffusion models as a promising platform also for constructing ensemble attacks.

V. Discussions and Outlooks

Besides the designs outlined in Section III, we have explored other strategies to enhance imperceptibility and transferability during the exploration of diffusion-based adversarial attacks. While these exploratory endeavors yielded limited success, we deem it valuable to provide an in-depth discussion, as they may contribute to future research. Detailed insights are presented in Appendices B and C.

We are also encouraged by the rapid growth of subsequent research in diffusion-based attacks, some of which emerged shortly after the initial public release of our work, highlighting the potential of this field. To help readers stay abreast of developments in this area, we briefly discuss these recent efforts. Unlike our work, which focuses on creating imperceptible and transferable adversarial attacks, Xue *et al.* [83] prioritizes controllability and stealthiness, proposing Diff-PGD, a method that combines PGD [11] with diffusion models to explore its applicability across different attack types, including style-guided and physical attacks. Wang *et al.* [84] also targets unrestricted attacks and introduces a semantic transformation and a latent masking technique to either fine-tune the diffusion model or modify the latent space. Additionally, Chen *et al.* [85] focuses on unrestricted adversarial attacks and integrates the momentum concept [16] to enhance attack performance.

Furthermore, we offer insights into potential future directions for diffusion-based adversarial attacks. One avenue for future research is to take diffusion models as a novel input augmentation. Recently, there are many works [19], [20] that enhance the attack's transferability by applying differentiable augmentations on the input image, in which way, the crafted adversarial examples gain robustness under different scenarios. In line with these approaches, we can also take diffusion models as novel augmentations. By directly adding noise (or applying DDIM Inversion), we first convert the input image into the latent space, then we conduct the diffusion denoising process to reconstruct images. This reconstruction process, with small differences from the input image every time, can be seen as an augmentation when we leverage stochastic sampling in each step (the way like DDPM [48] but not deterministic DDIM [47]). Therefore, we may expect good transferability in this way.

Moreover, as the adversarial example crafted by diffusion models has many semantic clues embedded in it (see Figure 1), it is also interesting and worth exploring whether the accuracy of clean images can be improved if we merge these examples in the training dataset and whether such an adversarial training

can enhance the robustness of the classifier without sacrificing the clean image accuracy compared with previous attacks [81].

Additionally, we identify three crucial aspects of diffusion-based attacks that merit further examination. First, the substantial computational cost, arising from the iterative nature and numerous parameters of diffusion models, potentially limits their practicality in real-time or resource-constrained settings (see Appendix G). Second, compared to pixel-based attacks, *DiffAttack* struggles to achieve a 100% white-box attack success rate, a phenomenon also observed in other generative-model-based (GAN-based) attacks [77] and unrestricted attacks [37], [29] (see Table I and Section IV-B3). Finally, in the transferable targeted attack task (see Appendix F), *DiffAttack*, along with other compared attacks, exhibits low transferability despite strong performance in the untargeted attack task. These findings also suggest promising avenues for future research.

VI. Conclusion

In this work, we explore the potential of diffusion models in crafting adversarial examples and propose a powerful transfer-based unrestricted attack. By leveraging the properties of diffusion models, our approach achieves both imperceptibility and transferability. Experiments across extensive black-box models, defenses, and datasets have demonstrated our method's superiority. Furthermore, we have comprehensively discussed the possible future work with diffusion models. We believe our work can pave the way for imperceptible and transferable adversarial attacks.

References

[1] S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, and H. X. Liu, "Dense reinforcement learning for safety validation of autonomous vehicles," *Nature*, vol. 615, no. 7953, pp. 620–627, 2023. 1

[2] Z. Zou, R. Zhang, S. Shen, G. Pandey, P. Chakravarty, A. Parchami, and H. X. Liu, "Real-time full-stack traffic scene perception for autonomous driving with roadside cameras," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 890–896. 1

[3] Y. Zhang, F. Xie, J. Chen, and J. Liu, "Tformer: A throughout fusion transformer for multi-modal skin lesion diagnosis," *Computers in Biology and Medicine*, p. 106712, 2023. 1

[4] Y. Zhang, F. Xie, X. Song, Y. Zheng, J. Liu, and J. Wang, "Dermoscopic image retrieval based on rotation-invariance deep hashing," *Medical Image Analysis*, vol. 77, p. 102301, 2022. 1

[5] J. Chen, K. Chen, H. Chen, W. Li, Z. Zou, and Z. Shi, "Contrastive learning for fine-grained ship classification in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022. 1

[6] J. Chen, K. Chen, H. Chen, Z. Zou, and Z. Shi, "A degraded reconstruction enhancement-based method for tiny ship detection in remote sensing images with a new large-scale dataset," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022. 1

[7] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013. 1, 2

[8] K. N. Kumar, C. K. Mohan, and L. R. Cenkeramaddi, "The impact of adversarial attacks on federated learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2023. 1

[9] M. Goldblum, D. Tsipras, C. Xie, X. Chen, A. Schwarzschild, D. Song, A. Madry, B. Li, and T. Goldstein, "Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 1563–1580, 2023. 1

[10] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," *arXiv preprint arXiv:1605.07277*, 2016. 1, 2

[11] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018. 1, 2, 14

[12] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014. 1, 2

[13] W. Brendel, J. Rauber, and M. Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models," in *International Conference on Learning Representations*, 2018. 1, 2

[14] N. Narodytska and S. P. Kasiviswanathan, "Simple black-box adversarial perturbations for deep networks," *arXiv preprint arXiv:1612.06299*, 2016. 1

[15] J. Lin, C. Song, K. He, L. Wang, and J. E. Hopcroft, "Nesterov accelerated gradient and scale invariance for adversarial attacks," in *International Conference on Learning Representations*, 2020. 1, 3, 10

[16] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193. 1, 2, 3, 10, 14

[17] Y. Lu, Y. Jia, J. Wang, B. Li, W. Chai, L. Carin, and S. Velipasalar, "Enhancing cross-task black-box transferability of adversarial examples with dispersion reduction," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2020, pp. 940–949. 1, 3

[18] N. Inkawhich, W. Wen, H. H. Li, and Y. Chen, "Feature space perturbations yield more transferable adversarial examples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7066–7074. 1, 3

[19] Y. Long, Q. Zhang, B. Zeng, L. Gao, X. Liu, J. Zhang, and J. Song, "Frequency domain model augmentation for adversarial attack," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*. Springer, 2022, pp. 549–566. 1, 3, 6, 7, 10, 14

[20] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2730–2739. 1, 3, 6, 10, 14

[21] Y. Dong, T. Pang, H. Su, and J. Zhu, "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4312–4321. 1, 3, 6, 10

[22] Z. Zhao, Z. Liu, and M. Larson, "Towards large yet imperceptible adversarial image perturbations with perceptual color distance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1039–1048. 1, 3

[23] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711. 1

[24] Y. Sharma, G. W. Ding, and M. A. Brubaker, "On the effectiveness of low frequency perturbations," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 3389–3396. 1

[25] W. Zhou, X. Hou, Y. Chen, M. Tang, X. Huang, X. Gan, and Y. Yang, "Transferable adversarial perturbations," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 452–467. 1

[26] S. Jia, B. Yin, T. Yao, S. Ding, C. Shen, X. Yang, and C. Ma, "Adv-attribute: Inconspicuous and transferable adversarial attack on face recognition," in *Advances in Neural Information Processing Systems*, 2022. 1, 3, 5

[27] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, "Diffusion models for adversarial purification," in *International Conference on Machine Learning*. PMLR, 2022, pp. 16 805–16 827. 1, 2, 3, 5, 6, 7

[28] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "A self-supervised approach for adversarial robustness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 262–271. 1, 6, 7

[29] S. Yuan, Q. Zhang, L. Gao, Y. Cheng, and J. Song, "Natural color fool: Towards boosting black-box unrestricted attacks," in *Advances in Neural Information Processing Systems*, 2022. 1, 3, 6, 7, 10, 14, 15

[30] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695. 1, 2, 3, 4, 6, 14

[31] J. Xu, S. Liu, A. Vahdat, W. Byeon, X. Wang, and S. De Mello, "Open-vocabulary panoptic segmentation with text-to-image diffusion models," *arXiv preprint arXiv:2303.04803*, 2023. 2, 3, 5

[32] K. Clark and P. Jaini, "Text-to-image diffusion models are zero-shot classifiers," in *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023. 2, 3, 5

[33] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *Proceedings of the 10th ACM workshop on artificial intelligence and security*, 2017, pp. 15–26. 2

[34] Y. Xiong, J. Lin, M. Zhang, J. E. Hopcroft, and K. He, "Stochastic variance reduced ensemble adversarial attack for boosting the adversarial transferability," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 983–14 992. 3

[35] X. Wang and K. He, "Enhancing the transferability of adversarial attacks through variance tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1924–1933. 3

[36] M. M. Naseer, S. H. Khan, M. H. Khan, F. Shahbaz Khan, and F. Porikli, "Cross-domain transferability of adversarial perturbations," *Advances in Neural Information Processing Systems*, vol. 32, 2019. 3, 8, 9

[37] A. Bhattad, M. J. Chong, K. Liang, B. Li, and D. Forsyth, "Unrestricted adversarial examples via semantic manipulation," in *International Conference on Learning Representations*, 2020. 3, 6, 15

[38] H. Qiu, C. Xiao, L. Yang, X. Yan, H. Lee, and B. Li, "Semanticadv: Generating adversarial examples via attribute-conditioned image editing," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer, 2020, pp. 19–37. 3

[39] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10 850–10 869, 2023. 3

[40] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 479–36 494, 2022. 3

[41] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, 2022. 3

[42] W. Li, X. Yu, K. Zhou, Y. Song, Z. Lin, and J. Jia, "Sdm: Spatial diffusion model for large hole image inpainting," *arXiv preprint arXiv:2212.02963*, 2022. 3

[43] S. Xie, Z. Zhang, Z. Lin, T. Hinz, and K. Zhang, "Smartbrush: Text and shape guided object inpainting with diffusion model," *arXiv preprint arXiv:2212.05034*, 2022. 3

[44] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[45] G. Parmar, K. K. Singh, R. Zhang, Y. Li, J. Lu, and J.-Y. Zhu, "Zero-shot image-to-image translation," *arXiv preprint arXiv:2302.03027*, 2023. 3, 4, 13

[46] R. Mokady, A. Hertz, K. Aberman, Y. Pritch, and D. Cohen-Or, "Null-text inversion for editing real images using guided diffusion models," *arXiv preprint arXiv:2211.09794*, 2022. 3, 4, 6, 13

[47] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021. 3, 4, 6, 14

[48] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020. 3, 14

[49] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International Conference on Machine Learning*. PMLR, 2015, pp. 2256–2265. 3

[50] G. Couairon, J. Verbeek, H. Schwenk, and M. Cord, "Diffedit: Diffusion-based semantic image editing with mask guidance," *arXiv preprint arXiv:2210.11427*, 2022. 4, 13

[51] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021. 4

[52] A. Hertz, R. Mokady, J. Tenenbaum, K. Aberman, Y. Pritch, and D. Cohen-Or, "Prompt-to-prompt image editing with cross attention control," *arXiv preprint arXiv:2208.01626*, 2022. 5, 13

[53] F. Tramèr, D. Boneh, A. Kurakin, I. Goodfellow, N. Papernot, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," in *6th International Conference on Learning Representations, ICLR 2018-Conference Track Proceedings*, 2018. 5, 6, 7, 14

[54] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8. 5

[55] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "Sdedit: Guided image synthesis and editing with stochastic differential equations," in *International Conference on Learning Representations*, 2021. 6

[56] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," California Institute of Technology, Tech. Rep. CNS-TR-2011-001, 2011. 6, 10

[57] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *Proceedings of the IEEE international conference on computer vision workshops*, 2013, pp. 554–561. 6, 10

[58] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 976–11 986. 6

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. 6, 10

[60] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 6

[61] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826. 6

[62] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520. 6

[63] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. 6

[64] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022. 6

[65] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International conference on machine learning*. PMLR, 2021, pp. 10 347–10 357. 6

[66] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24 261–24 272, 2021. 6

[67] A. Mustafa, S. H. Khan, M. Hayat, J. Shen, and L. Shao, "Image super-resolution as a defense against adversarial attacks," *IEEE Transactions on Image Processing*, vol. 29, pp. 1711–1724, 2019. 6, 7

[68] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, "Mitigating adversarial effects through randomization," in *International Conference on Learning Representations*, 2018. 6, 7

[69] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu, "Defense against adversarial attacks using high-level representation guided denoiser," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1778–1787. 6, 7

[70] A. Thomas and O. Elibol, "Defense against adversarial attacks-3rd place," https://github.com/anlthms/nips-2017/blob/master/poster/defense.pdf, 2017. 6, 7

[71] A. Kurakin, I. Goodfellow, S. Bengio, Y. Dong, F. Liao, M. Liang, T. Pang, J. Zhu, X. Hu, C. Xie *et al.*, "Adversarial attacks and defences competition," in *The NIPS'17 Competition: Building Intelligent Systems*. Springer, 2018, pp. 195–231. 6, 7

[72] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019. 6

[73] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017. 6

[74] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595. 6

[75] L. Gao, Q. Zhang, J. Song, X. Liu, and H. T. Shen, "Patch-wise attack for fooling deep neural network," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. Springer, 2020, pp. 307–322. 6, 14

[76] C. Laidlaw and S. Feizi, "Functional adversarial attacks," *Advances in neural information processing systems*, vol. 32, 2019. 6

[77] O. Poursaeed, I. Katsman, B. Gao, and S. Belongie, "Generative adversarial perturbations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4422–4431. 8, 9, 15
[78] Q. Zhang, X. Li, Y. Chen, J. Song, L. Gao, Y. He *et al.*, "Beyond imagenet attack: Towards crafting adversarial examples for black-box domains," in *International Conference on Learning Representations*, 2022. 8, 10
[79] Z. He, W. Wang, J. Dong, and T. Tan, "Transferable sparse adversarial attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 963–14 972. 8, 9
[80] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141. 10
[81] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," in *International Conference on Learning Representations*, 2017. 11, 15
[82] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763. 14
[83] H. Xue, A. Araujo, B. Hu, and Y. Chen, "Diffusion-based adversarial sample generation for improved stealthiness and controllability," *arXiv preprint arXiv:2305.16494*, 2023. 14
[84] C. Wang, J. Duan, C. Xiao, E. Kim, M. Stamm, and K. Xu, "Semantic adversarial attacks via diffusion models," *arXiv preprint arXiv:2309.07398*, 2023. 14
[85] Z. Chen, B. Li, S. Wu, K. Jiang, S. Ding, and W. Zhang, "Content-based unrestricted adversarial attack," *arXiv preprint arXiv:2305.10665*, 2023. 14

Jianqi Chen received his B.S. and M.S. degrees from the Image Processing Center, School of Astronautics, Beihang University, China in 2021 and 2024. He is currently pursuing his Ph.D. degree. His research interests lie in AISafety, image synthesis, and remote sensing image processing.

Hao Chen received his B.S. and Ph.D. degrees from the Image Processing Center, School of Astronautics, Beihang University in 2017 and 2023, respectively. He is currently a Junior Researcher at Shanghai AI Laboratory. His research interests include geospatial machine learning, remote sensing, earth monitoring, and prediction.

Keyan Chen received the B.S. and M.S. degrees from the School of Astronautics, Beihang University, Beijing, China, in 2019 and 2022, respectively. He is currently pursuing the Ph.D. degree with the Image Processing Center, School of Astronautics, Beihang University. His research interests include image processing, machine learning, and pattern recognition.

Yilan Zhang received her B.S. and M.S. degrees from the Image Processing Center, School of Astronautics, Beihang University in 2021 and 2024. She is currently pursuing her Ph.D. degree. Her research interests include deep learning, computer vision, and multimodal learning.

Zhengxia Zou is currently a Professor at the School of Astronautics, Beihang University. He received his B.S. degree and his Ph.D. degree from Beihang University in 2013 and 2018. During 2018-2021, he was a postdoc research fellow at the University of Michigan, Ann Arbor. His research interests include remote sensing image processing and computer vision. He has published more than 40 peer-reviewed papers in top-tier journals and conferences, including Nature Communications, Proceedings of the IEEE, IEEE Transactions on Image Processing, Transactions on Geoscience and Remote Sensing, and IEEE / CVF Computer Vision and Pattern Recognition. His personal website is https://zhengxiazou.github.io/.

Zhenwei Shi (Senior Member, IEEE) is currently a Professor and Dean of the Image Processing Center, School of Astronautics, Beihang University. He has authored or co-authored over 200 scientific articles in refereed journals and proceedings, including the IEEE Transactions on Pattern Analysis and Machine Intelligence, the IEEE Transactions on Image Processing, the IEEE Transactions on Geoscience and Remote Sensing, the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) and the IEEE International Conference on Computer Vision (ICCV). His current research interests include remote sensing image processing and analysis, computer vision, pattern recognition, and machine learning.

Prof. Shi serves as an Editor for IEEE Transactions on Geoscience and Remote Sensing, Pattern Recognition, ISPRS Journal of Photogrammetry and Remote Sensing, Infrared Physics and Technology, etc. His personal website is http://levir.buaa.edu.cn/.

Diffusion Models for Imperceptible and Transferable Adversarial Attack – Appendix –

Jianqi Chen, Hao Chen, Keyan Chen, Yilan Zhang, Zhengxia Zou, and Zhenwei Shi*, *Senior Member, IEEE*

APPENDIX A
OVERVIEW

In Appendix B and Appendix C, considering the possible help for future research, we display our further trials (with little success) on improving the imperceptibility and transferability of the attacks. We present *DiffAttack*'s performance on more surrogate models in Appendix D, as well as its application to additional dataset and model type in Appendix E. Besides, we give discussions about the *DiffAttack*'s performance on the transferable targeted attack in Appendix F. Limitations of computational cost are discussed in Appendix G. Finally, more visualizations are shown in Appendix H.

APPENDIX B
TRIAL FOR BETTER IMPERCEPTIBILITY WITH "PSEUDO" MASK

As mentioned in Section III-E in the main paper, for some specific images, the adversarial examples crafted by *DiffAttack* may distort a lot compared with the original ones. For better control of the changes, we try to generate "pseudo" masks with the cross attention. With these masks, we can then filter out the background regions and only perturb the foreground objects, thus achieving better human-imperception. However, we found that although the results could more easily evade the human eyes, their transferability dropped a lot. We infer this may be because background information is also beneficial for image recognition. More details about the implementation and experiments of the trial can be found as follows. In practice, we will weaken the inversion strength for overly distorted images.

A. Design Details

As mentioned in Section III-D in the main paper, there is a strong relationship in the cross-attention maps between the text prompt and the image pixels. Thus, we can make use of this property to generate the true label's "pseudo" mask:

$$P = \text{Average}(\text{Cross}(x_t, t, C; \text{SDM})) \tag{1}$$

$$M_{soft} = \text{Up}(\frac{P}{\text{Max}(P)}) \tag{2}$$

$$\text{(Optional)} \quad M_{hard} = \begin{cases} 1, & M_{soft} > 0.5 \\ 0, & M_{soft} \leq 0.5 \end{cases} \tag{3}$$

where $\text{Up}(\cdot)$ is an upsampling operation to resize the cross-attention map (due to the existing downsamplings in the encoder of the Autoencoder and U-Net). $\text{Max}(\cdot)$ is to extract the maximum value and normalize the cross-attention maps P. Since $P \geq 0$, the normalized $M_{soft} \in [0, 1]$. Eq. 3 is optional to get a hard mask. With the mask, we then filter out the background area and only apply perturbations on the foreground (area covered by true objects). The Eq. 12 in the main paper is then changed as follows:

$$\arg\min_{x_t} \mathcal{L}_{attack} = -J(x' \times M + x \times (1 - M), y; G_\phi) \tag{4}$$

The optimization details are the same as the implementation details in Section IV-A in the main paper.

B. Experiments and Analysis

Here we conduct experiments to see the impact of different upsampling strategies and different mask types. In Table I, we display the performance when the mask is applied. It can be perceived from the results that there is an obvious trade-off between transferability and imperceptibility. The use of masks lowers the FID and LPIPS value, yet also lowers the attack success by a large margin. We infer that it is because the recognition of an image is not only related to its foreground but also its background [1]. Thus the attack success rate will drop when the mask is applied. We also visualize the adversarial example crafted by leveraging the mask in Figure 1, from which we can see that the applied mask can better preserve words on hot air balloon skin, and the hard mask tends to generate blocky artifacts compared with soft-mask.

APPENDIX C
TRIAL ON FURTHER IMPROVING TRANSFERABILITY

We also explore further improving the transferability of *DiffAttack*. For image classification, the classifier will output each category's confidence, and *top1* is usually taken as the final decision. We here try to also make use of the following 4 categories in *top5* for better transferability. Specifically, different from Section III-D in the main paper where we only set the guided text to *top1* category's name, we here set the text to a stack of the 5 categories' names in *top5* (sort by confidence from largest to smallest). Then, we optimize x_t to reduce the intensity of cross attention between pixels and the first category text and increase that between pixels and the other four categories text. The motivation is that, the confidence denotes, to some extent, the amount of related information of the category in the image, thus it may be much easier to

TABLE I
COMPARISONS OF DIFFERENT MASK TYPES AND UPSAMPLING STRATEGIES. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED.

Mask Types	Upsampling Strategy	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
		Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
None	None	**59.5**	**55.6**	**55.4**	**13.9**	**76.9**	**75.2**	**72.8**	**74.0**	**71.0**	**58.9**	**54.7**	**65.4**	62.3	0.126
hard	nearest	71.4	67.9	64.8	17.8	85.2	80.9	82.9	80.3	81.0	68.2	60.2	74.2	59.1	**0.064**
hard	bilinear	68.8	66.8	65.6	18.3	84.0	79.0	81.4	79.9	79.5	66.0	61.8	73.3	59.3	0.065
soft	bilinear	73.9	69.4	66.9	18.4	88.2	82.7	86.5	84.8	82.0	68.5	62.0	76.5	**58.8**	**0.064**

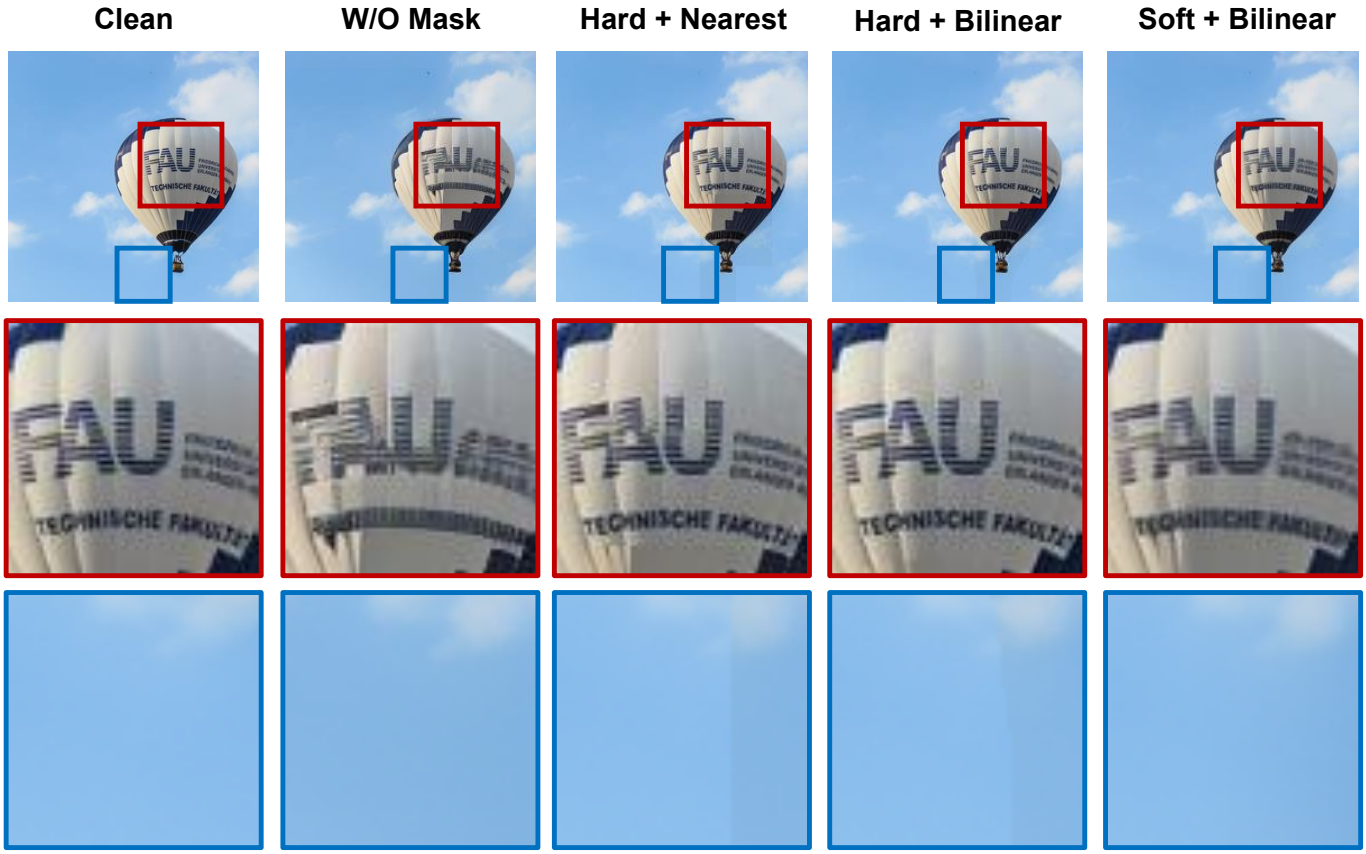

Fig. 1. **Visualization of the adversarial example crafted by leveraging the mask.** The second and third rows denote the scaled-up regions in the first row.

deceive the classifier to the nearest category on the decision plane. However, from our experiments, this trial fails to improve the transferability and even hurts it. We attribute this to the limitation of the search space. More details can be found as follows.

A. Design Details

In Eq. 15 in the main paper, $C =$ "{True Label / Category 1_{st}}" that the guided text can be either the true label or the *top1* predicted category. We here extend the text to leverage more categories:

$$C_{ext} = \text{"}\{\text{Category } 1_{\text{st}}\},\ \{\text{Category } 2_{\text{nd}}\},\ \cdots,\ \{\text{Category N}_{\text{th}}\}\text{"} \tag{5}$$

where {Category N_{th}} denotes the name of the N_{th} most possible category predicted by the classifier. Then, Eq. 12 in the main paper is modified to:

$$\mathcal{L}_{attack} = -J(x', \text{Category } 1_{\text{st}}; G_\phi) + \underbrace{(J(x', \text{Category } 2_{\text{nd}}; G_\phi) + \cdots + J(x', \text{Category N}_{\text{th}}; G_\phi))}_{N-1} \tag{6}$$

By minimizing the above equation, the adversarial examples are crafted to lead the classification results towards the most error-prone categories, which may have benefits on the transferability (but failed through our experiments). We further add an extra loss to force the perturbed x_t to have lower cross-attention intensity with {Category 1_{st}} and higher with other text prompts, which we expect can help deceive the diffusion models:

$$P_i = \text{Average}(\text{Cross}(x_t, t, C_i; \text{SDM})) \tag{7}$$

$$\arg\max_{x_t} \mathcal{L}_{ext} = \text{Average}(\underbrace{P_2 + \cdots + P_N}_{N-1}) - P_1 \tag{8}$$

TABLE II

EXPLORATIONS ON THE EFFECT OF DIFFERENT CATEGORIES AS TEXT PROMPTS. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED.

top-N	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
1	**59.5**	**55.6**	**55.4**	13.9	**76.9**	**75.2**	**72.8**	**74.0**	**71.0**	**58.9**	54.7	**65.4**	62.3	**0.126**
2	67.0	63.7	61.2	9.1	81.7	77.1	77.5	79.1	76.4	64.0	56.8	70.4	**60.9**	0.132
5	64.4	61.0	59.2	**5.9**	81.4	78.8	78.0	77.5	75.9	61.0	**54.0**	69.1	62.8	0.133

TABLE III

COMPARISONS ON MORE SURROGATE MODELS. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. "S." DENOTES SURROGATE MODELS WHILE "T." DENOTES TARGET MODELS. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED, AND THE SECOND-BEST RESULT IS UNDERLINED.

S. \ T.	Attacks	CNNs					Transformers				MLPs		AVG↓ (w/o self)	FID↓	LPIPS↓
		Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L			
	Clean	92.7	88.7	86.9	80.5	97.0	93.7	95.9	94.5	94.0	82.5	76.5	89.4	57.8	—
ViT-B	PI-FGSM	34.2	27.7	23.6	31.5	66.9	0	56.5	25.6	17.0	29.7	26.3	33.9	91.2	0.360
	S^2I-FGSM	45.0	39.6	38.6	38.1	63.1	0.2	45.2	10.7	5.5	18.1	20.2	32.4	70.2	0.177
	NCF	45.1	40.4	39.6	56.1	73.5	27.6	70.1	64.1	57.8	49.7	44.9	54.1	67.4	0.364
	DiffAttack(Ours)	39.4	40.5	36.1	34.7	41.7	4.7	30.3	22.4	19.9	27.2	30.0	**32.2**	**66.4**	**0.152**
DeiT-B	PI-FGSM	33.8	19.4	22.8	30.6	64.7	22.5	54.5	0	16.7	32.6	28.9	34.4	92.1	0.362
	S^2I-FGSM	39.8	34.5	29.3	32.6	50.4	6.7	28.0	0.4	3.9	13.6	17.9	**25.7**	75.8	0.166
	NCF	52.2	47.3	46.7	59.3	73.4	62.5	67.5	31.7	59.2	50.9	48.0	56.7	**65.3**	0.336
	DiffAttack(Ours)	39.9	40.4	36.8	37.0	37.5	22.4	25.9	3.1	18.2	26.2	27.9	31.2	67.6	**0.146**
Mix-B	PI-FGSM	47.5	37.0	39.2	40.8	72.1	49.9	70.9	56.6	45.1	0	13.9	47.3	85.5	0.344
	S^2I-FGSM	60.6	52.4	47.8	52.1	72.6	48.4	58.4	43.9	40.7	1.6	8.9	48.6	66.4	0.154
	NCF	55.0	47.5	49.6	61.1	81.8	71.1	77.5	75.6	71.1	10.0	35.0	62.5	65.2	0.326
	DiffAttack(Ours)	52.2	52.1	49.6	45.0	57.9	48.8	49.9	44.6	45.4	16.6	22.1	**46.8**	**64.2**	**0.143**

where C_i denotes Category $\mathrm{i_{th}}$, and P_i denotes the cross attention between image pixels and C_i. $\mathrm{Average}(\cdot)$ here represents the averaging operation in pixel space. We then add $\mathcal{L}_{ext}$ to Eq. 17 in the main paper with a weight factor set to 100.

B. Experiments and Analysis

We here analyze the impact of different numbers of categories leveraged as text prompts. From Table II, leveraging more guided category texts failed to improve the attack's transferability, and even damage the performance. We infer that it is because the search space of the attack is limited when we force the adversarial examples to be classified as some specific categories. When we set the category number from 2 to 5, we can observe a slight increase in the attack success, while when we set it to 1, we have no constraint on the predicted category, and thus gain a large increase in the attack success.

APPENDIX D
PERFORMANCE ON ADDITIONAL SURROGATE MODELS

Besides the results in Table I in the main paper, we supplement more experiments when the surrogate models are Transformers or MLPs in Table III. Here, we further consider ViT-B, DeiT-B, and Mix-B as the surrogate model. For brevity, we only compare DiffAttack with those more recent attack methods [2], [3], [4]. From the results, it is further verified that DiffAttack generalizes well on various model structures, achieving good performance on both imperceptibility and transferability.

APPENDIX E
PERFORMANCE ON ADDITIONAL DATASET AND MODEL TYPE

While the experiments presented in the main paper (Tables I and V) demonstrate the effectiveness of our method across various models and datasets, concerns may still arise regarding the potential risk of overfitting to specific dataset types or model architectures. This is because the data distribution of the ImageNet, CUB-200-2011, and Stanford Cars datasets closely resembles that of the Stable Diffusion training set. To more thoroughly assess how well the adversarial examples generated by our method perform against new models and datasets, we supplemented our experiments with the UC Merced [5] remote sensing scene classification dataset. Compared to the datasets validated in the main paper, UC Merced has a data distribution that diverges from the Stable Diffusion training data, making it an ideal candidate for testing DiffAttack's performance on a new dataset.

In terms of model type selection, besides ResNet50, ViT-B, and Swin-B, we included RSMamba [6], a model based on the recently introduced Mamba [7] architecture. This addition allows us to evaluate DiffAttack's performance on a novel model type. For this exploration, we randomly sampled 200 images from the UC Merced dataset and compared our results

TABLE IV

EXPERIMENTS ON ADDITIONAL DATASET AND MODEL TYPE. WE REPORT TOP-1 ACCURACY(%) OF EACH METHOD. "S." DENOTES SURROGATE MODELS WHILE "T." DENOTES TARGET MODELS. FOR THE WHITE-BOX ATTACKS (SURROGATE MODEL SAME AS TARGET ONE), WE SET THEIR BACKGROUND TO GRAY. "AVG(W/O SELF)" DENOTES THE AVERAGE ACCURACY ON ALL THE TARGET MODELS EXCEPT THE ONE THAT SAME AS THE SURROGATE ONE. THE BEST RESULT IS BOLDED.

S. \ T.	Attacks	Res-50	Swin-B	ViT-B	RSMamba-B	AVG(w/o self)↓	FID↓	LPIPS↓
	clean	96.0	89.5	83.5	96.5	91.4	97.4	-
Res-50	S^2I-FGSM	0.5	78.0	68.0	85.5	77.2	141.6	0.119
	DiffAttack(Ours)	44.5	60.5	59.0	67.0	**62.2**	**136.8**	**0.116**
Swin-B	S^2I-FGSM	89.0	0	64.0	90.0	81.0	143.6	0.121
	DiffAttack(Ours)	52.5	7.0	37.0	53.5	**47.7**	**137.7**	**0.117**
ViT-B	S^2I-FGSM	90.5	78.0	18.0	95.0	87.8	141.5	0.125
	DiffAttack(Ours)	90.0	76.5	40.5	93.5	**86.7**	**133.4**	**0.120**
RSMamba-B	S^2I-FGSM	75.0	51.5	64.5	0	63.7	148.1	0.117
	DiffAttack(Ours)	60.5	60.0	56.9	12.0	**59.1**	**145.6**	**0.115**

TABLE V

PERFORMANCE COMPARISONS ON TARGETED TRANSFERABLE ATTACKS. WE REPORT **ATTACK SUCCESS RATE**(%) OF EACH METHOD HERE. WE CRAFT ADVERSARIAL EXAMPLES ON VGG-19. "SUCCESS RATE AVG(W/O SELF)" DENOTES THE AVERAGE ATTACK SUCCESS RATE ON ALL THE TARGET MODELS EXCEPT THE ONES THAT HAVE A GRAY BACKGROUND. THE BEST RESULT IS BOLDED.

Targeted Attacks	CNNs					Transformers				MLPs		Success Rate	FID↓	LPIPS↓
	Res-50	VGG-19	Mob-v2	Inc-v3	ConvNeXt	ViT-B	Swin-B	DeiT-B	DeiT-S	Mix-B	Mix-L	AVG (w/o self)↑		
DI-FGSM	0.4	94.7	0.5	0.2	0.1	0.0	0.1	0.0	0.0	0.0	0.0	0.1	75.6	0.145
TI-FGSM	0.3	97.3	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	**70.0**	0.150
PI-FGSM	0.1	99.7	0.2	0.0	0.0	0.1	0.1	0.1	0.0	0.1	0.2	0.1	92.3	0.369
S^2I-FGSM	2.0	91.4	1.9	0.5	0.8	0.0	0.3	0.1	0.0	0.0	0.0	0.6	82.9	0.156
ReColorAdv	0.1	59.3	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	75.8	0.169
cAdv	0.0	95.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	79.7	0.221
DiffAttack($1e^{-2}$)	0.3	61.4	0.3	0.0	0.2	0.1	0.2	0.0	0.2	0.0	0.0	0.1	74.7	**0.143**
DiffAttack($1e^{-1}$)	6.1	99.8	5.6	3.5	6.3	4.1	3.8	2.6	3.2	0.9	1.1	**3.7**	147.2	0.508

TABLE VI

LIMITATION IN TERMS OF TIME AND GPU MEMORY CONSUMPTION. WE REPORT THE TIME FOR CRAFTING ADVERSARIAL EXAMPLES OF DIFFERENT ATTACK METHODS, TOGETHER WITH THE MAXIMUM MEMORY COST. FOR GAN-BASED METHODS (BIA), THE ADVERSARIAL EXAMPLES ARE CRAFTED BY INFERENCING THE TRAINED GENERATOR. WHILE FOR OTHER METHODS, THE ADVERSARIAL EXAMPLES ARE ITERATIVELY OPTIMIZED WITH RES-50 AS THE SURROGATE MODEL.

Attack	DI-FGSM	TI-FGSM	PI-FGSM	S^2I-FGSM	ReColorAdv	cAdv	NCF	BIA	DiffAttack
Mem(MB)	336	301	412	305	952	775	374	242	14083
Time(s)	0.2	0.2	0.6	5.5	2.9	5.3	18.6	0.01	29.9

with the S^2I-FGSM attack method. The comparison results are presented in Table IV.

From the experimental outcomes, it is evident that DiffAttack generalizes well to both the new dataset and the new model type, achieving satisfactory attack transferability and imperceptibility. This further verifies the robustness and generalization capability of our attack method.

APPENDIX F
DISCUSSIONS ON DIFFATTACK'S PERFORMANCE IN TRANSFERABLE TARGETED ATTACK

In this section, we assess the performance of DiffAttack when employed as a targeted attack method. Originally designed for the untargeted attack, we adapt DiffAttack for the targeted task by removing $L_{transfer}$ in Section III-D in the main paper directly. To transform all compared methods in the main paper into targeted attacks, we modify their loss functions by reversing the sign of the classification loss to maximize the logit for the target category. Notably, we exclude NCF due to its extremely low success rate in targeted attacks. For target categories, we employ the labels provided in the ImageNet-Compatible Dataset. Adversarial examples are crafted using VGG-19, and the results are presented in Table V. Differing from the results presented in other tables, here we present the *attack success rate* of the target attacks for clarity. The attack success rate is essentially the complement of the top-1 accuracy, calculated as 100% minus the top-1 accuracy.

From the results, we observe that all models struggle to achieve transferability to black models, a notable and promising avenue for future research. Additionally, when compared to pixel-based attacks, DiffAttack exhibits a lower success rate on the targeted model. We attribute this difference to the tendency of pixel-based attacks to overfit by introducing high-frequency noise. In contrast, unrestricted attacks like DiffAttack often emphasize large-scale patterns with high-level semantics, making it challenging to achieve a high success rate in white-

box attacks (this also occurs among those GAN-based attacks). It's worth noting that by increasing the learning rate from $1e^{-2}$ (our default setting) to $1e^{-1}$, DiffAttack can improve its white-box attack success rate and also its transferability. However, this enhancement comes at the cost of reduced fidelity which may be less meaningful.

APPENDIX G
DISCUSSIONS ABOUT LIMITATION OF TIME AND MEMORY COST

Due to the iterative characteristic and the substantial number of parameters in diffusion models, DiffAttack has a limitation in terms of time and memory consumption compared to other attack methods. In Table VI, we display a comprehensive comparison of computational cost and runtime among DiffAttack, pixel-based attacks, and GAN-based attacks.

The comparison is to process a 224×224 image on a single RTX 3090 GPU. The results reveal that DiffAttack consumes greater memory and generally takes longer to generate adversarial examples. This could hinder its deployment in resource-constrained settings, such as autonomous driving and edge models, or for targeting real-time systems.

Notably, this is a common drawback shared by all approaches relying on diffusion models. However, we hope to note that due to the popularity of diffusion models these years, famous communities such as PyTorch and Huggingface keep advancing the efficiency and memory optimization of diffusion models (like Pytorch 2.0 and Diffusers repository). Many recent works [8], [9] have also been dedicated to accelerating diffusion models and addressing memory costs. We firmly believe that these efforts will help bridge the computational gap between DiffAttack and other attack methods in the future, further fostering research on diffusion-based attacks.

To better highlight this bottleneck and provide readers with a clearer understanding of these constraints, as well as which aspects can be improved in the future, we explored possible strategies to speed up processing time. By testing on different software and hardware environments, we found significant improvements. The processing time of 29.9 seconds, shown in Table VI, was measured using a 3090 GPU with Torch version 1.13. By upgrading to Torch version 2+ (tested on version 2.2), which includes optimizations for attention calculations in diffusion models, the processing time was reduced to approximately 24 seconds. This time can be further reduced to less than 18 seconds when using the advanced 4090 GPU, without any observed impact on the attack's performance.

Another way to accelerate *DiffAttack* is by modifying the internal optimization iterations. As mentioned in Section IV-A, we use a default of 30 iterations for latent optimization. By reducing this number to 25 or 20 iterations, we can achieve faster processing speeds (from 29.9 seconds to approximately 24 and 20 seconds, respectively). Additionally, reducing the number of DDIM inversion steps can speed up processing (from 29.9 seconds to around 23 seconds) and lower memory usage (from 14GB to approximately 12GB) when reducing the steps from 5 to 4. However, these adjustments are double-edged swords, as they can reduce the attack success rate (but improve imperceptibility), as shown in Figure 7 in the main paper. Therefore, when considering such optimizations, the specific application scenarios should be carefully evaluated to determine whether speed is more critical than the attack success rate, unlike the more straightforward improvements offered by software and hardware upgrades.

We also explored the use of mixed precision for potential memory savings. However, due to the mechanism that stores both weight tensors in FP32 and their variants cast to FP16, and because *DiffAttack* processes a single sample at a time, the mixed precision strategy did not yield significant benefits. The memory consumed by the large weight tensors outweighed the advantages gained from intermediate low-precision feature maps.

APPENDIX H
MORE QUANTITIVE STUDIES AND VISUALIZATIONS

As a supplement to the experiments in Section IV in the main paper, we here display more visual comparisons in Figure 2 and Figure 3, from which it can be observed that the adversarial examples crafted by our attack are human-imperceptible and hard to be perceived.

REFERENCES

[1] Z. Zhu, L. Xie, and A. Yuille, "Object recognition with and without objects," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017, pp. 3609–3615. 1

[2] L. Gao, Q. Zhang, J. Song, X. Liu, and H. T. Shen, "Patch-wise attack for fooling deep neural network," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. Springer, 2020, pp. 307–322. 3

[3] Y. Long, Q. Zhang, B. Zeng, L. Gao, X. Liu, J. Zhang, and J. Song, "Frequency domain model augmentation for adversarial attack," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*. Springer, 2022, pp. 549–566. 3

[4] S. Yuan, Q. Zhang, L. Gao, Y. Cheng, and J. Song, "Natural color fool: Towards boosting black-box unrestricted attacks," in *Advances in Neural Information Processing Systems*, 2022. 3

[5] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, 2010, pp. 270–279. 3

[6] K. Chen, B. Chen, C. Liu, W. Li, Z. Zou, and Z. Shi, "Rsmamba: Remote sensing image classification with state space model," *IEEE Geoscience and Remote Sensing Letters*, 2024. 3

[7] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023. 3

[8] A. Ulhaq, N. Akhtar, and G. Pogrebna, "Efficient diffusion models for vision: A survey," *arXiv preprint arXiv:2210.09292*, 2022. 5

[9] T. Dao, D. Fu, S. Ermon, A. Rudra, and C. Ré, "Flashattention: Fast and memory-efficient exact attention with io-awareness," *Advances in Neural Information Processing Systems*, vol. 35, pp. 16 344–16 359, 2022. 5

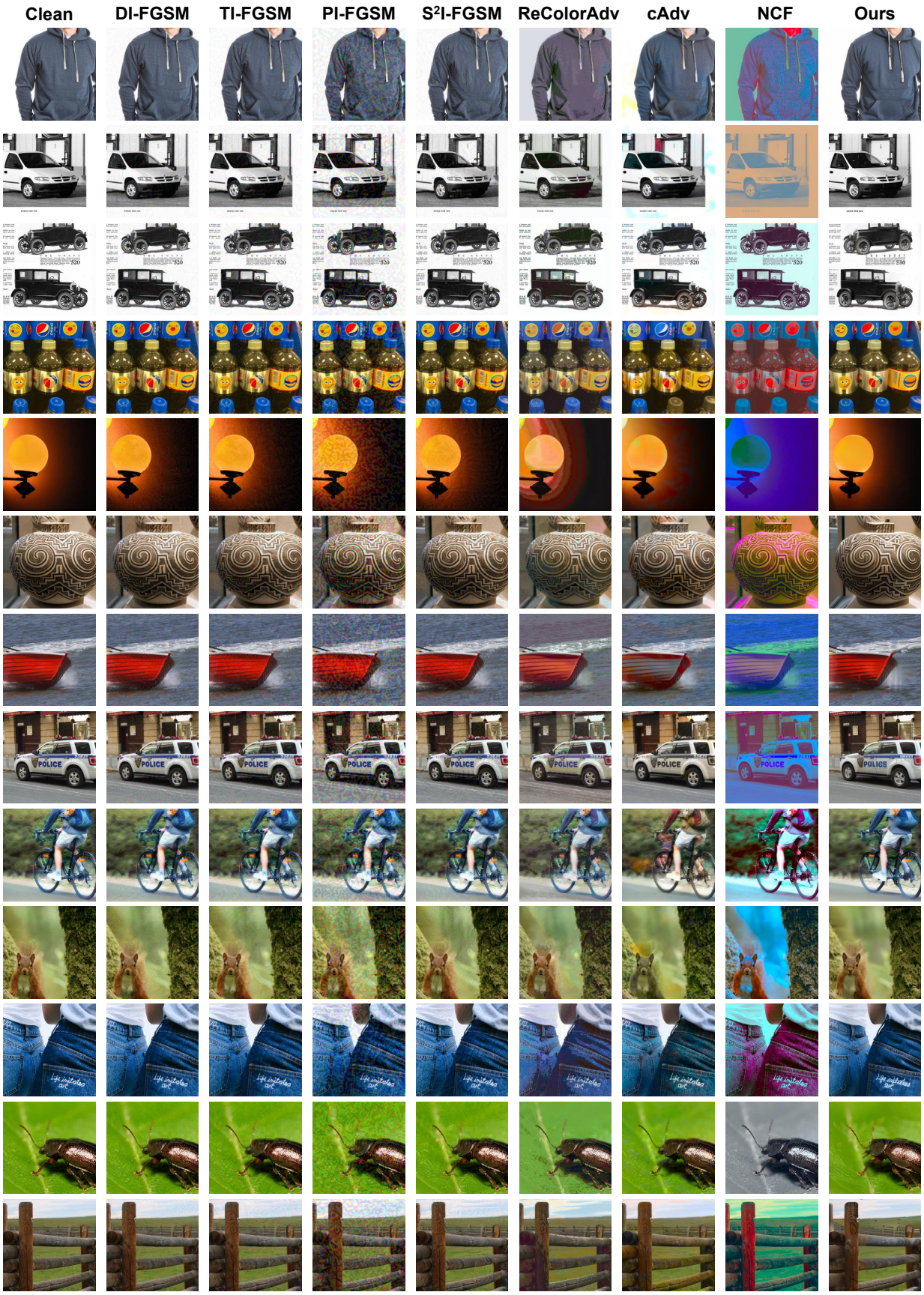

Fig. 2. **Supplement visualization of adversarial examples crafted by different attacks.** Please zoom in for a better view.

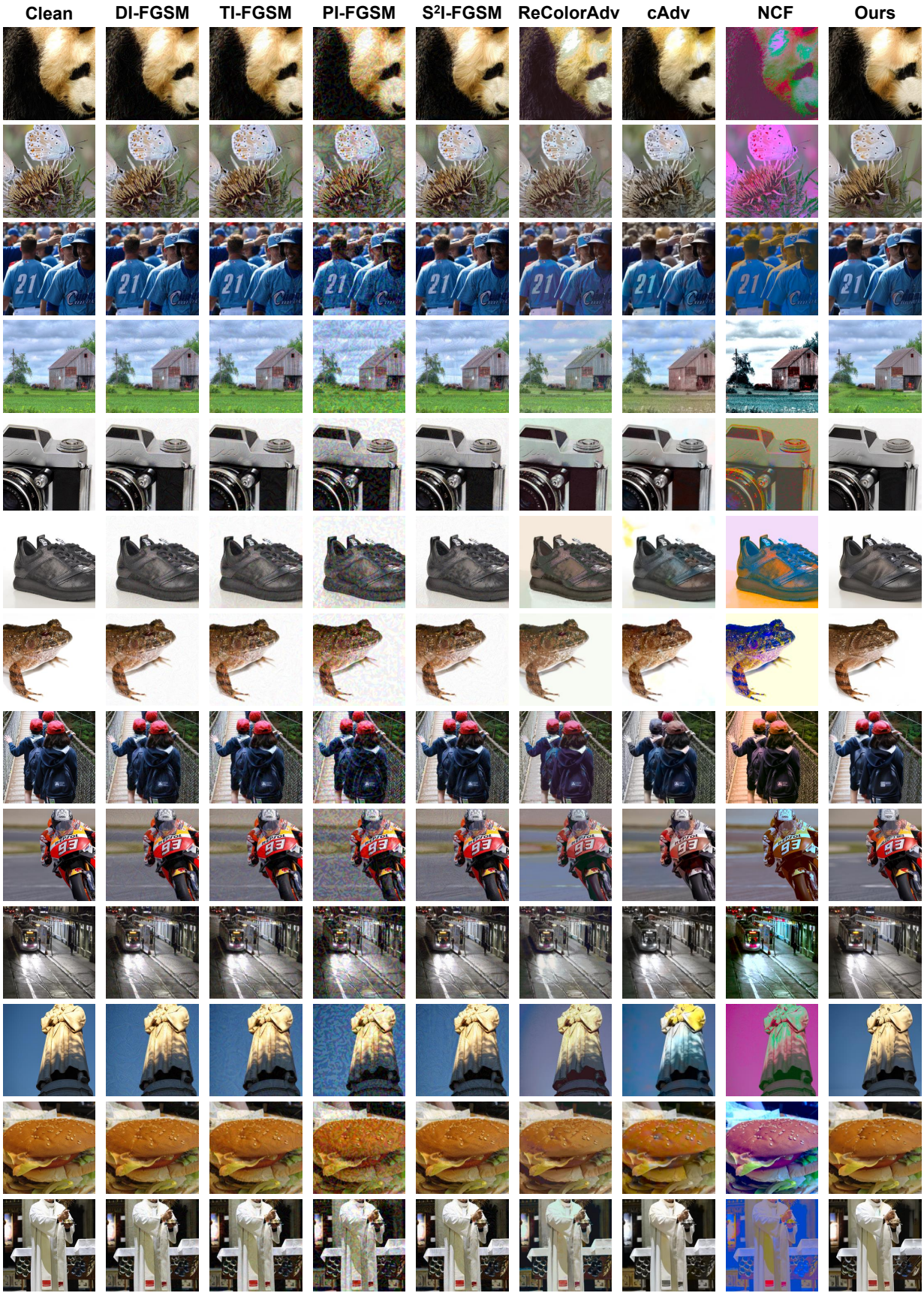

Fig. 3. **Supplement visualization of adversarial examples crafted by different attacks.** Please zoom in for a better view.