Efficient Generation of Targeted and Transferable Adversarial Examples for Vision-Language Models Via Diffusion Models

Qi Guo, Shanmin Pang, *Member, IEEE*, Xiaojun Jia, Yang Liu, *Senior Member, IEEE*, Qing Guo, *Member, IEEE*

***Abstract*—Adversarial attacks, particularly targeted transfer-based attacks, can be used to assess the adversarial robustness of large visual-language models (VLMs), allowing for a more thorough examination of potential security flaws before deployment. However, previous transfer-based adversarial attacks incur high costs due to high iteration counts and complex method structure. Furthermore, due to the unnaturalness of adversarial semantics, the generated adversarial examples have low transferability. These issues limit the utility of existing methods for assessing robustness. To address these issues, we propose AdvDiffVLM, which uses diffusion models to generate natural, unrestricted and targeted adversarial examples via score matching. Specifically, AdvDiffVLM uses Adaptive Ensemble Gradient Estimation to modify the score during the diffusion model's reverse generation process, ensuring that the produced adversarial examples have natural adversarial targeted semantics, which improves their transferability. Simultaneously, to improve the quality of adversarial examples, we use the GradCAM-guided Mask method to disperse adversarial semantics throughout the image rather than concentrating them in a single area. Finally, AdvDiffVLM embeds more target semantics into adversarial examples after multiple iterations. Experimental results show that our method generates adversarial examples 5x to 10x faster than state-of-the-art transfer-based adversarial attacks while maintaining higher quality adversarial examples. Furthermore, compared to previous transfer-based adversarial attacks, the adversarial examples generated by our method have better transferability. Notably, AdvDiffVLM can successfully attack a variety of commercial VLMs in a black-box environment, including GPT-4V.**

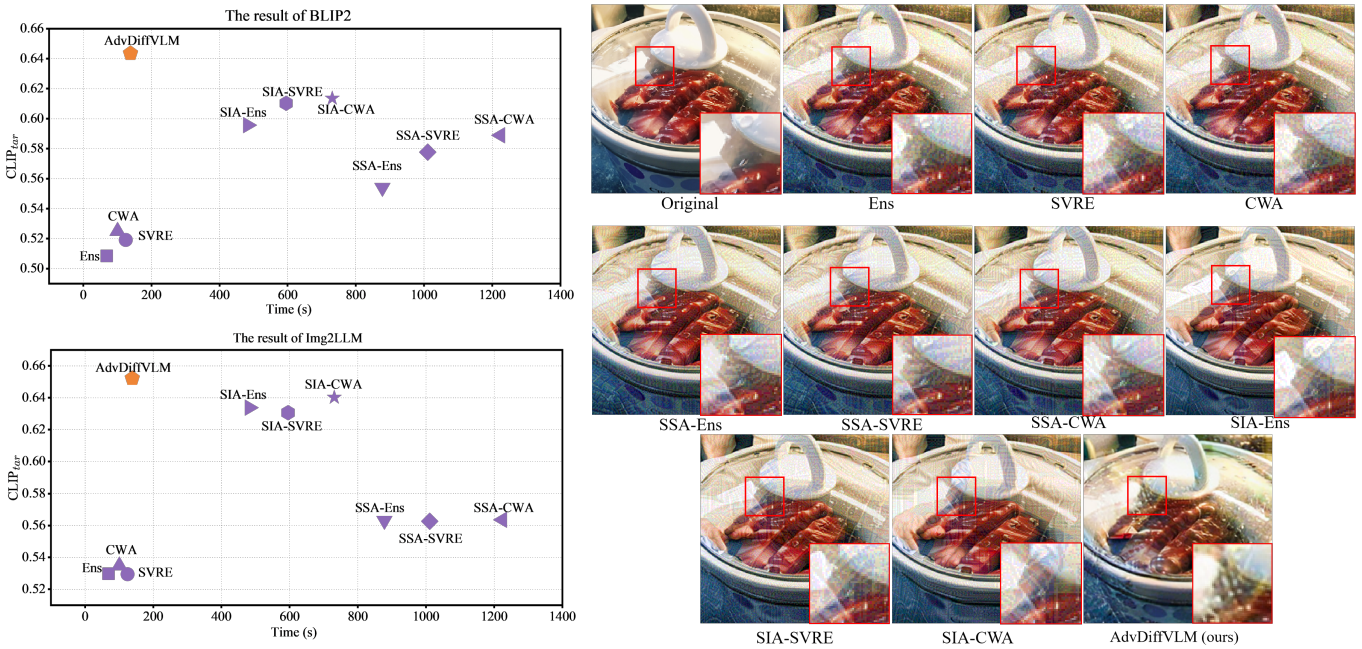
***Index Terms*—Adversarial Attack, Visual Language Models, Diffusion Models, Score Matching.**

I. Introduction

LARGE VLMs have shown great success in tasks like image-to-text [1]–[3] and text-to-image generation [4], [5]. Particularly in image-to-text generation, users can use images to generate executable commands for robot control [6], which has potential applications in autonomous driving systems [7], [8], visual assistance systems [9], and content moderation systems [10]. However, VLMs are highly susceptible to adversarial attacks [11], [12], which can result in life and property safety issues [13], [14]. As a result, it is critical to evaluate the adversarial robustness [15]–[18] of these VLMs before deployment.

The early research on assessing the adversarial robustness of VLMs concentrated on white-box and untargeted scenarios [19]–[21]. Black-box and targeted attacks can cause models to generate targeted responses without knowing the models' internal information, resulting in greater harm [22], [23]. Else, targeted attacks on black-box models present more challenges than untargeted attacks [24], [25]. As a result, when assessing the adversarial robustness of VLMs, it is critical to consider more threatening and challenging black-box and targeted attacks [26]. AttackVLM [26] is the first work to explore the adversarial robustness of VLMs in both black-box and targeted scenarios using query attacks with transfer-based priors. However, due to the large number of queries required and the complex method structure, this method is inefficient, which limits its usefulness in evaluating VLMs. Another attack method that can be used in black-box and targeted scenarios is the transfer-based attack [27]–[30]. However, this type of attack method is slow to generate adversarial examples due to its complex structure and numerous iterations. Else, because it adds unnatural adversarial semantics, the transferability of adversarial examples is poor. Unrestricted adversarial examples [31]–[34] can incorporate more natural adversarial targeted semantics into the image, thereby improving the image quality and transferability of adversarial examples. For example, AdvDiffuser [33] incorporates PGD [35] into the reverse process of the diffusion model to generate targeted adversarial examples with better transferability against classification models. However, we find that applying PGD to the latent image in the reverse process is not suitable for the more difficult task of attacking VLMs. At the same time, performing PGD on each step of the reverse process incurs high costs.

To attack VLMs effectively and efficiently, we consider using adaptive ensemble gradient to guide score matching [39] during latent image generation, which can naturally embed more adversarial target semantics than AdvDiffuser does. Specifically, we propose AdvDiffVLM, which employs diffusion models to generate natural, unrestricted and targeted adversarial examples based on score matching. We leverage and modify the pre-trained diffusion models' reverse process, using Adaptive Ensemble Gradient Estimation to modify the score and embed target semantics in adversarial examples. To improve the naturalness of the output, we propose the GradCAM-guided Mask, which distributes the adversarial target semantics across adversarial examples, thereby preventing the model from producing adversarial features in specific areas and improving image quality. In addition, we embed more target semantics into adversarial examples through multiple iterations. As shown in Figure 1, compared with current attacks, AdvDiffVLM generates targeted adversarial examples faster, and the generated adversarial examples have better transferability. In addition, the generated adversarial examples are more natural. Therefore, it can be used as a more effective



(a) Comparison of attack performance of different transfer-based attacks

(b) Comparison of visualization results of different transfer-based attacks

Fig. 1. Comparison of different transfer-based attacks and our method on VLMs. (a) Comparison of attack performance. We select BLIP2 [2] and Img2LLM [36] as the representation models of VLMs. We select existing transfer-based attacks in conjunction with AttackVLM [26] as comparison methods, including Ens [37], SVRE [28], CWA [27], SSA [38] and SIA [29]. We report the CLIP$_{tar}$ score, which is the similarity between the response generated by the input images. (b) Comparison of image quality. We enlarge the local area of the adversarial examples to enhance visual effects. It is evident that adversarial examples generated by transfer-based attacks exhibit notable noise. Our method has better visual effects. Magnify images for improved contrast.

tool to evaluate the adversarial robustness of VLMs.

We summarize our contributions as follows:

- We propose the AdvDiffVLM framework to efficiently generate targeted and transferable adversarial examples for VLMs.
- We propose the Adaptive Ensemble Gradient Estimation module that embeds adversarial target semantics into adversarial examples based on score matching, and the GradCAM-guided Mask Generation module that further improves the visual quality of adversarial examples.
- Extensive experiments show that our method generates targeted adversarial examples 5x to 10x faster than state-of-the-art adversarial attack methods in attacking VLMs, and the generated adversarial examples exhibit better transferability. Additionally, our method can successfully induce black-box commercial VLMs to output target responses.

II. Related Work

A. Visual-Language Models (VLMs)

Large language models (LLMs) [40]–[42] have demonstrated great success in a variety of language-related tasks. The knowledge contained within these powerful LLMs has aided the development of VLMs. There are several strategies and models for bridging the gap between text and visual modalities [43], [44]. Some studies [2], [45] extract visual information from learned queries and combine it with LLMs to enhance image-based text generation. Models like LLaVA [3] and MiniGPT-4 [46] learn simple projection layers to align visual encoder features with LLM text embeddings. Some works [5] train VLMs from scratch, which promotes better alignment of visual and textual modalities. In this paper, we focus on the adversarial robustness of these VLMs, with the goal of discovering security vulnerabilities and encouraging the development of more robust and trustworthy VLMs.

B. Adversarial Attacks in VLMs

Adversarial attack methods are classified as white-box or black-box attacks based on adversary knowledge, as well as targeted or untargeted attacks based on adversary objectives [47]–[49]. Studies have investigated the robustness of VLMs, focusing on adversarial challenges in visual question answering [26] and image captioning [19]. However, most studies focus on traditional CNN-RNN-based models, which make assumptions about white-box access or untargeted goals, limiting their applicability in real-world scenarios. Recently, AttackVLM [26] implemented both transfer-based and query-based attacks on large open-source VLMs with black-box access and targeted goals. Nonetheless, this approach is time-consuming due to its reliance on numerous VLM queries. In addition, [50] studied the adversarial robustness of VLMs using ensemble transfer-based attacks, assuming untargeted goals. In this paper, we investigate the adversarial robustness of VLMs against targeted transfer-based attacks. Initially, we evaluate VLM's robustness against current SOTA transfer-based attacks in conjunction with AttackVLM. We then examine the limitations of current methods and implement targeted improvements, culminating in the proposal of AdvDiffVLM.

TABLE I
HYPERPARAMETERS OF VARIOUS TRANSFER-BASED ATTACKS, WHERE EN REPRESENTS THE ENSEMBLE METHODS AND DA REPRESENTS THE DATA AUGMENTATION METHODS. PLEASE NOTE THAT THE SAME HYPERPARAMETER NAME IN DIFFERENT PAPERS MAY MEAN DIFFERENT MEANINGS, AND WE USE THE MEANING IN THEIR PAPERS.

Methods	Type	Hyperparameters
Ens [37]	EN	$\mu = 1$
SVRE [28]	EN	$M = 16, \alpha = 160/255$
CWA [27]	EN	$\alpha = 160/255, \beta = 250, r = 16/255/15$
SSA [38]	DA	$\alpha = 160/255, N = 20, \rho = 0.5, \sigma = 16/255$
SIA [29]	DA	$s = 3, N = 20, \alpha = 160/255$

C. Unrestricted Adversarial Examples

Researchers are increasingly interested in unrestricted adversarial examples, as the l_p norm distance fails to capture human perception [31]–[34], [51], [52]. Some approaches use generative methods to create unrestricted adversarial examples. For example, [31], [32] modified the latent representation of GANs to produce unrestricted adversarial examples. However, due to the limited interpretability of GANs, the generated adversarial examples are of poor quality. Diffusion models [53] are SOTA generative models based on likelihood and theoretical foundations, sampling data distribution with high fidelity and diversity. AdvDiffuser [33] incorporated the PGD [35] method into the diffusion model's reverse process, resulting in high-quality adversarial examples without restrictions. In this study, we explore using the diffusion model for generating unrestricted adversarial examples, focusing on modifying the score in the diffusion model's reverse process rather than adding noise to the latent image. We discuss the differences between our method and AdvDiffuser in more detail in Section IV-D.

III. PRELIMINARIES

A. Diffusion Models

In this work, we use diffusion models [4], [53], [54] to generate unrestricted and targeted adversarial examples. In a nutshell, diffusion models learn a denoising process from $x_T \sim \mathcal{N}(x_T; 0, \mathbf{I})$ to recover the data $x_0 \sim q(x_0)$ with a Markov chain and mainly include two processes: forward process and reverse process. Forward process defines a fixed Markov chain. Noise is gradually added to the image x_0 over T time steps, producing a series of noisy images $\{x_1, x_2, \cdots, x_T\}$. Specifically, noise is added by $q(x_t \mid x_0) := \sqrt{\bar{\alpha}_t} x_0 + \epsilon\sqrt{1-\bar{\alpha}_t}, \epsilon \sim \mathcal{N}(0,1)$, where $\alpha_t := 1 - \beta_t$, $\overline{\alpha_t} := \prod_{s=1}^{t} \alpha_s$ and β_t is a fixed variance to control the step sizes of the noise. The purpose of the reverse process is to gradually denoise from x_T to obtain a series of $\{\tilde{x}_{T-1}, \tilde{x}_{T-2}, \cdots, \tilde{x}_1\}$, and finally restore x_0. It learns the denoising process through a denoising model ε_θ, and the training objective is $\mathcal{L}_{simple} := E_{t\sim[1,T],\epsilon\sim\mathcal{N}(0,\mathbf{I})} \left\|\varepsilon_\theta\left(x_t, t\right) - \epsilon\right\|^2$.

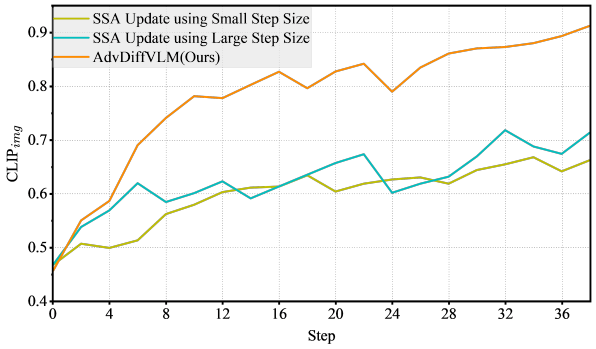


Fig. 2. The CLIP$_{img}$ score varies with the step sizes. Here, CLIP$_{img}$ is the similarity between the adversarial examples and the adversarial target images, which is calculated by the visual encoder of CLIP ViT-B/32. We choose SSA [38] as the representative of transfer-based attacks.

B. Problem Settings

Then we give the problem setting of this paper. We denote the victim VLM model as f_ξ, and aim to induce f_ξ to output the target response. This can be formalized as

$$\begin{aligned} &\max\ CS(g_\psi(f_\xi\left(\boldsymbol{x}_{\text{adv}}; \boldsymbol{c}_{\text{in}}\right)), g_\psi(\boldsymbol{c}_{\text{tar}})) \\ &\text{s.t. } D(\boldsymbol{x}, \boldsymbol{x}_{\text{adv}}) \leq \delta \end{aligned} \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^{3\times H\times W}$ represents the original image, $\boldsymbol{x}_{\text{adv}}$ and $\boldsymbol{c}_{\text{tar}}$ respectively refer to adversarial example and adversarial target text, and $g_\psi(\cdot)$ denotes the CLIP text encoder. Besides, $D(\boldsymbol{x}, \boldsymbol{x}_{\text{adv}}) \leq \delta$ places a bound on a distance metric, and $CS(\cdot,\cdot)$ refers to the cosine similarity metric. Finally, $\boldsymbol{c}_{\text{in}}$ denote the input text.

Since f_ξ is a black-box model, we generate adversarial examples on the surrogate model ϕ_ψ and subsequently transfers them to f_ξ. In addition, inspired by [26], matching image-image features can lead to better results, we define the problem as,

$$\begin{aligned} &\max\ CS(\phi_\psi(\boldsymbol{x}_{\text{adv}}), \phi_\psi(\boldsymbol{x}_{\text{tar}})) \\ &\text{s.t. } D(\boldsymbol{x}, \boldsymbol{x}_{\text{adv}}) \leq \delta \end{aligned} \tag{2}$$

where $\boldsymbol{x}_{\text{tar}}$ represents the target image generated by $\boldsymbol{c}_{\text{tar}}$. We use stable diffusion [4] to implement the text-to-image generation. ϕ_ψ refers to CLIP image encoder. Our study is the most realistic and challenging attack scenarios, i.e., targeted and transfer scenarios.

C. Rethinking Transfer-based Attacks

Transfer-based attacks can effectively solve Eq.2. In this context, we assess the robustness of VLMs against current SOTA transfer-based attacks, in conjunction with AttackVLM. Specifically, we consider ensemble methods, data augmentation methods, and combinations of both. We primarily employ the simple ensemble version of data augmentation attacks, as relying on a single surrogate model tends to yield poor performance. The hyperparameter settings are shown in Table I. Else, in all attacks, the value range of adversarial example pixels is [0,1]. We set the perturbation budget as $\delta = 16/255$ under the ℓ_∞ norm. The number of iterations for all attacks is set to 300. In addition, we use the MI-FGSM [37] method and set $\mu = 1$. Please note that the same hyperparameter name

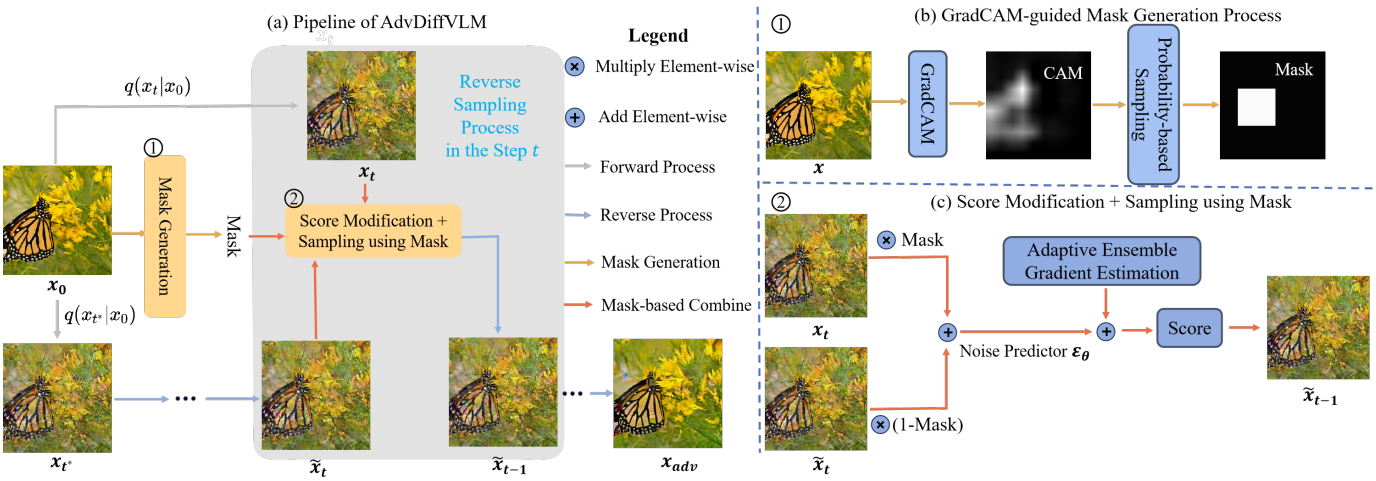


Fig. 3. The main framework of the AdvDiffVLM for efficiently generating transferable unrestricted adversarial examples. AdvDiffVLM mainly includes two components: Adaptive Ensemble Gradient Estimation and GradCAM-guided Mask. Details are respectively described in Secs. IV-B and IV-C. Please refer to Section IV for specific symbol meanings.

in different papers may mean different meanings, and we use the meaning in their papers.

The outcomes of these transfer-based attacks on VLMs are depicted in Figure 1. As illustrated, current transfer-based attacks face challenges such as slow adversarial example generation, noticeable noise within these examples, and limited transferability. The limitations of existing transfer-based attacks on VLMs are analyzed as follows: First, existing SOTA transfer-based attacks only access the original image during the optimization of Eq.2. Consequently, they employ small steps and strategies like data augmentation to tentatively approach the optimal solution, necessitating numerous iterations and resulting in high attack costs. As shown in Figure 2, using a larger step size results in pronounced fluctuations during the optimization process. This issue may be mitigated by leveraging score, which provides insights into the data distribution. By offering score guidance towards solving Eq.2, quicker convergence is expected. Therefore score information can be considered in the design of new improved attack method. Second, existing transfer-based attacks introduce unnatural adversarial noises with limited transferability. Unrestricted adversarial examples can introduce more natural adversarial targeted semantics, increasing transferability. This implies that new transfer-based targeted attacks can consider unrestricted adversarial attacks.

IV. Methodology

In this section, we first present the theoretical analysis of our method and then offer a comprehensive description of the proposed AdvDiffVLM. Finally, we delineate the distinctions between our method and AdvDiffuser. The main framework of AdvDiffVLM is illustrated in Figure 3.

A. Theoretical Analysis

We are focused on modeling adversarial attacks from a generative perspective, considering how to utilize the data distribution (score) of the generative model to produce natural, unrestricted and targeted adversarial examples. Additionally, as indicated in [55], learning to model the score function is equivalent to modeling the negative of the noise, suggesting that score matching and denoising are equivalent processes. Thus, our method derives from integrating diffusion models and score matching, positioning it as a novel approach for generating high-quality, unrestricted, transferable and targeted adversarial examples.

Formally, we want to obtain distribution meeting the condition that the adversarial example has target semantic information during the reverse generation process

$$p(x_{t-1}|x_t, f_\xi(\boldsymbol{x}_{\text{adv}}; \boldsymbol{c}_{\text{in}}) = \boldsymbol{c}_{\text{tar}}) \tag{3}$$

where x_t represents the latent image of the diffusion model. Next, we start from the perspective of score matching [39] and consider the score $\nabla \log p(x_{t-1}|x_t, \boldsymbol{c}_{\text{tar}})$ of this distribution, where ∇ is the abbreviation for ∇_{x_t}. According to Bayes theorem,

$$\begin{aligned}
&\nabla \log p\left(x_{t-1} \mid x_t, \boldsymbol{c}_{\text{tar}}\right) = \nabla \log \left(\frac{p(\boldsymbol{c}_{\text{tar}}|x_{t-1}, x_t) \cdot p(x_{t-1}|x_t)}{p(\boldsymbol{c}_{\text{tar}}|x_t)}\right) \\
&= \nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_{t-1}, x_t\right) + \nabla \log p\left(x_{t-1} \mid x_t\right) \\
&\quad -\nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_t\right) \\
&= \nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_{t-1}\right) + \nabla \log p\left(x_t \mid x_{t-1}, \boldsymbol{c}_{\text{tar}}\right) \\
&-\nabla \log p\left(x_t \mid x_{t-1}\right) + \nabla \log p\left(x_{t-1} \mid x_t\right) - \nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_t\right) \\
&= \nabla \log p\left(x_t \mid x_{t-1}, \boldsymbol{c}_{\text{tar}}\right) - \nabla \log p\left(x_t \mid x_{t-1}\right) \\
&\quad +\nabla \log p\left(x_{t-1} \mid x_t\right) - \nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_t\right)
\end{aligned} \tag{4}$$

$p\left(x_t \mid x_{t-1}, c_{tar}\right)$ and $p\left(x_t \mid x_{t-1}\right)$ respectively denote the add noise process with target text and the add noise process devoid of target semantics. From an intuitive standpoint, whether target text is present or not, the forward noise addition process follows a Gaussian distribution, and the added noise remains consistent, indicating that the gradient solely depends on x_t. The difference between x_t without target text and x_t with target text is minimal, as constraints are employed to ensure minimal variation of the adversarial sample from the original sample. Therefore, $\nabla \log p\left(x_t \mid x_{t-1}, \boldsymbol{c}_{\text{tar}}\right)$ and $\nabla \log p\left(x_t \mid x_{t-1}\right)$ are approximately equal. So the final score is $\nabla \log p\left(x_{t-1} \mid x_t\right) - \nabla \log p\left(\boldsymbol{c}_{\text{tar}} \mid x_t\right)$.

Because score matching and denoising are equivalent processes, that is, $\nabla \log p\left(x_t\right) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}}\epsilon_0$. Therefore we can get score $(\nabla \log p(x_{t-1}|x_t, \boldsymbol{c}_{\text{tar}}))$,

$$\text{score} = -(\frac{\varepsilon_\theta\left(x_t\right)}{\sqrt{1-\bar{\alpha}_t}} + \nabla \log p_{f_\xi}\left(c_{\text{tar}} \mid x_t\right)) \tag{5}$$

where ε_θ is denoising model, and $\bar{\alpha}_t$ is the hyperparameter.

Eq.5 demonstrates that the score of $p(x_{t-1}|x_t, \boldsymbol{c}_{\text{tar}})$ can be derived by incorporating gradient information into the inverse process of the diffusion model. Consequently, adversarial semantics can be incrementally embedded into adversarial examples based on the principle of score matching.

B. Adaptive Ensemble Gradient Estimation

Since f_ξ is a black-box model and cannot obtain gradient information, we use surrogate model to estimate $\nabla \log p_{f_\xi}\left(c_{\text{tar}} \mid x_t\right)$. As a scalable method for learning joint representations between text and images, CLIP [56] can leverage pre-trained CLIP models to establish a bridge between images and text. Therefore we use the CLIP model as the surrogate model to estimate the gradient.

Specifically, we first add noise to the original image $\boldsymbol{x}$ by t^* steps through the forward process $q(x_{t^*}|x_0)$ to obtain x_{t^*}, where $x_0 = \boldsymbol{x}$. Then, at each step of reverse process, we change score:

$$\text{score} = -(\frac{1}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta\left(\tilde{x}_t\right) + s\nabla_{\tilde{x}_t}(CS(\phi_\psi(\tilde{x}_t), \phi_\psi(\boldsymbol{x}_{\text{tar}})))) \tag{6}$$

where s is the adversarial gradient scale used to control the degree of score change and $\tilde{x}_t$ is the latent image in the reverse process.

We find that gradient estimation using only a single surrogate model is inaccurate. Therefore, we consider using a set of surrogate models $\left\{\phi_\Psi^i\right\}_{i=1}^{N_m}$ to better estimate the gradient. Specifically, we make the following improvements to Eq. 6:

$$\text{score} = -(\frac{\varepsilon_\theta(\tilde{x}_t)}{\sqrt{1-\bar{\alpha}_t}} + s\nabla_{\tilde{x}_t}(w_i \textstyle\sum_{i=1}^{N_m} CS(\phi_\psi^i(\tilde{x}_t), \phi_\psi^i(\boldsymbol{x}_{\text{tar}})))) \tag{7}$$

where $\mathbf{w} = (w_1, w_2, \cdots, w_{N_m})$ represents the weight of cosine loss of different models.

Since different images have different sensitivities to surrogate models, only using simple ensemble cannot obtain optimal solution. Inspired by [57], we propose a new adaptive ensemble method, and obtain $\mathbf{w}$ in Eq. 7 in the following way:

$$w_i(t) = \frac{\sum_{j=1}^{N_m} \exp(\tau \mathcal{L}_j(t+1)/\mathcal{L}_j(t+2))}{N_m \exp(\tau \mathcal{L}_i(t+1)/\mathcal{L}_i(t+2))} \tag{8}$$

where τ refers to the temperature. A larger τ makes all weights close to 1. $\mathcal{L}_i = CS(\phi_\psi^i(\tilde{x}_t), \phi_\psi^i(\boldsymbol{x}_{\text{tar}}))$. We initialize $\{w_i(t^*)\}_{i=1}^{N_m}$ and $\{w_i(t^*-1)\}_{i=1}^{N_m}$ to 1. Through Eq. 8, we reduce the weight of surrogate models with fast-changing losses to ensure that gradient estimations of different surrogate models are updated simultaneously.

Finally, we set the perturbation threshold δ, and then clip the adversarial gradient to ensure the naturalness of the synthesized adversarial examples.

Algorithm 1: The overall algorithm of AdvDiffVLM

Input: Original image $\boldsymbol{x}$, N_m surrogate models ϕ_θ^i, adversarial guidance scale s, reverse generation process timestep t^*, mask area size k, perturbation threshold ϵ, temperature τ, adversarial target image $\boldsymbol{x}_{\text{tar}}$, Number of iterations N.
Output: adversarial example $\boldsymbol{x}_{\text{adv}}$.

Initialize $\{w_i\}_{i=1}^{N_m} = 1$, CAM, $x_0 = \boldsymbol{x}$;
Sample $x_{t^*} \sim q(x_{t^*}|x_0)$, let $\tilde{x}_{t^*} = \bar{x}_{t^*} = x_{t^*}$;
for $n \leftarrow 1, \cdots, N$ **do**
  for $t \leftarrow t^*, \cdots, 1$ **do**
    Get mask m according to CAM;
    $x_t \sim q(x_t|x_0)$;
    $\hat{x}_t = \mathrm{m} \odot x_t + (1-\mathrm{m}) \odot \tilde{x}_t$;
    $w_i = \frac{\sum_{j=1}^{N_m} \exp(\tau \mathcal{L}_j(t+1)/\mathcal{L}_j(t+2))}{N_m \exp(\tau \mathcal{L}_i(t+1)/\mathcal{L}_i(t+2))}$;
    $g = \nabla_{\tilde{x}_t}(w_i \sum_{i=1}^{N_m} CS(\phi_\psi^i(\tilde{x}_t), \phi_\psi^i(\boldsymbol{x}_{\text{tar}})))$;
    $g = clip(g, -\delta, \delta)$;
    score $= \varepsilon_\theta\left(\bar{x}_t\right)/\sqrt{1-\bar{\alpha}_t} + s \cdot g$;
    $\tilde{x}_{t-1} = -\sqrt{1-\bar{\alpha}_t} \times$ score;
  end
end
Return $\boldsymbol{x}_{\text{adv}} = \tilde{x}_0$

C. GradCAM-guided Mask Generation

We detail Adaptive Ensemble Gradient Estimation in the previous section. However, we note that only relying on adaptive ensemble gradient estimation leads to the generation of obvious adversarial features in specific areas, resulting in poor visual effects. To achieve a balance between the natural visual effects and attack capabilities of adversarial examples, we introduce GradCAM-guided Mask, which utilizes a mask to combine the forward noisy image x_t and the generated image $\tilde{x}_t$. Through the combination, the adversarial semantics concentrated in the adversarial examples across the entire image is distributed, thereby enhancing natural visual effect of the adversarial examples.

First, we utilize GradCAM [58] to derive the class activation map CAM of $\boldsymbol{x}$ with respect to ground-truth label $\boldsymbol{y}$. CAM assists in identifying important and non-important areas in the image. Subsequently, we clip the CAM values to the range $[0.3, 0.7]$ and normalize them to obtain the probability matrix P. We sample according to the P to obtain the coordinate (x, y), and then set the $\mathrm{k} \times \mathrm{k}$ area around (x, y) to be 1 and the remaining areas to be 0 to obtain mask m. Here, m has the same shape as $\tilde{x}_t$. This approach disperses more adversarial features in non-important areas and less in important areas of adversarial examples, improving the natural visual effect of adversarial examples.

At each step t, we combine x_t and $\tilde{x}_t$ as following:

$$\hat{x}_t = \mathrm{m} \odot x_t + (1-\mathrm{m}) \odot \tilde{x}_t \tag{9}$$

where $\odot$ refers to Hadamard Product. Afterwards, we can obtain new score by integrating $\varepsilon_\theta\left(\hat{x}_t\right)$ with the estimated gradient and then use $\tilde{x}_{t-1} = -\sqrt{1-\bar{\alpha}_t} \times$ score for sampling.

Finally, we take the generated adversarial example as x_0, and iterate N times to embed more target semantics into it. We provide a complete algorithmic overview of AdvDiffVLM in Algorithm 1.

TABLE II
COMPARISON WITH EXISTING SOTA ATTACK METHODS, WHERE THE BEST RESULT IS **BOLDED**. NOTE THAT WE USE FOUR VERSIONS OF THE CLIP VISUAL ENCODER, INCLUDING RESNET50, RESNET101, VIT-B/16 AND VIT-B/32, AS SURROGATE MODELS. SINCE UNIDIFFUSER USES VIT-B/32 AS THE VISUAL ENCODER, IT IS A GRAY BOX SCENARIO, WHICH WE INDICATE WITH *. ELSE, WE PROVIDE THE AVERAGE TIME (S) FOR EACH STRATEGY TO CRAFT A SINGLE x_{adv}. THE SHADED PARTS REPRESENT OUR PROPOSED METHOD.

	Unidiffuser*		BLIP2		MiniGPT-4		LLaVA		Img2LLM		Time(s)
	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	
Original	0.4770	0.0%	0.4931	0.0%	0.4902	0.0%	0.5190	0.0%	0.5288	0.0%	/
Ens	0.7353	99.1%	0.5085	0.9%	0.4980	1.8%	0.5366	3.5%	0.5297	4.5%	69
SVRE	0.7231	100.0%	0.5190	2.4%	0.5107	2.2%	0.5385	4.6%	0.5292	3.8%	125
CWA	0.7568	100.0%	0.5249	5.2%	0.5211	3.8%	0.5493	7.1%	0.5346	5.4%	101
SSA-Ens	0.7275	100.0%	0.5539	9.2%	0.5175	10.1%	0.6098	37.5%	0.5629	19.6%	879
SSA-SVRE	0.7217	100.0%	0.5776	18.7%	0.5395	16.5%	0.6005	40.2%	0.5625	18.4%	1012
SSA-CWA	0.7485	100.0%	0.5888	23.3%	0.5407	20.6%	0.6152	40.7%	0.5634	20.4%	1225
SIA-Ens	0.7377	100.0%	0.5956	49.6%	0.5605	40.4%	0.7158	84.7%	0.6337	27.0%	483
SIA-SVRE	0.7302	100.0%	0.6102	50.1%	0.5782	46.4%	0.7122	88.3%	0.6305	35.4%	596
SIA-CWA	0.7498	100.0%	0.6135	51.8%	0.5810	47.8%	0.7194	89.5%	0.6401	40.6%	732
AdvDiffuser_{ens}	0.6774	86.7%	0.5396	8.6%	0.5371	8.2%	0.5507	25.3%	0.5395	11.5%	574
$\text{AdvDiffuser}_{adaptive}$	0.6932	88.9%	0.5424	10.4%	0.5391	9.6%	0.5595	27.4%	0.5502	14.8%	602
AdvDiffVLM	0.7502	100.0%	**0.6435**	**66.7%**	**0.6145**	**58.6%**	**0.7206**	**91.2%**	**0.6521**	**43.8%**	139

TABLE III
THE DETAILS OF VICTIM VLMS, INCLUDE CODE AND CONFIGURATION.

Models	Code	Version
Unidifusser	https://github.com/thu-ml/unidiffuser	/
BLIP2	https://github.com/salesforce/LAVIS	(blip2_opt, pretrain_opt2.7b)
MiniGPT-4	https://github.com/Vision-CAIR/MiniGPT-4	(Vicuna 7B)
LLaVA	https://github.com/haotian-liu/LLaVA	(Vicuna, llava-v1.5-7b)
Img2LLM	https://github.com/salesforce/LAVIS	(img2prompt_vqa, base)

D. Differences from AdvDiffuser

Both our method and AdvDiffuser [33] produce unrestricted adversarial examples using the diffusion model. Here, we discuss the distinctions between them, highlighting our contributions.

Tasks of varying difficulty levels: AdvDiffuser is oriented towards classification models, while our research targets the more intricate Vision-Language Models (VLMs). Initially, within the realm of classification tasks, each image is associated with a singular label. Conversely, in the image-to-text tasks, images may be linked to numerous text descriptions. When faced with an attack targeting a singular description, VLMs have the capability to generate an alternate description, thereby neutralizing the attack's effectiveness. As a result, our task presents a greater challenge.

Different theoretical foundations: AdvDiffuser posits that PGD can introduce adversarial noise. It begins with Gaussian noise, subsequently incorporating high-frequency adversarial perturbations into the latent image in a sequential manner. Given that the diffusion model's reverse process inherently constitutes a denoising procedure, it necessitates numerous iterations to introduce sufficient perturbations, leading to heavy computation. In contrast, our method derives from score matching, where we employ CLIP to estimate gradient, subsequently altering the score rather than adding it into latent image. Through score matching, the adversarial gradient can be perfectly integrated into the reverse generation process without being weakened. In summary, AdvDiffuser applies PGD to the latent image in the reverse process of the diffusion model. In contrast, we incorporate score matching into the generation process of the latent image, which can embed more adversarial target semantics more naturally than AdvDiffuser. Furthermore, our approach obviates the need for initiating with Gaussian noise, initially introducing noise to $\boldsymbol{x}$ through t^* steps, followed by the application of adversarial gradient to modify score, thereby facilitating more efficient generation of adversarial examples.

Distinct schemes of GradCAM utilization: The GradCAM mask utilized by AdvDiffuser leads to restricted modification of crucial image areas, rendering it inadequate for image-based attacks. Addressing this issue, we have introduced the GradCAM-guided Mask. Contrary to utilizing GradCAM results directly as a mask, we employ them as a directive to generate the mask further. This not only guarantees a likelihood of modification across all image areas but also secures minimal alteration of significant areas, striking a balance between image quality and attack ability.

V. EXPERIMENTS

A. Experimental Setup

Datasets and victim VLMs: Following [50], we use NeurIPS'17 adversarial competition dataset, compatible with ImageNet, for all the experiments. Else, we select 1,000 text descriptions from the captions of the MS-COCO dataset as our adversarial target texts and then use Stable Diffusion [4] to generate 1,000 adversarial targeted images. For the victim VLMs, SOTA open-source models are evaluated, including Unidiffuser [5], BLIP2 [2], MiniGPT-4 [46], LLaVA [3] and Img2LLM [36]. The details are shown in Table III. Among them, Unidiffuser is a gray-box model, and the others are black-box models.

Baselines: We compare with AdvDiffuser [33] and other SOTA transfer-based attackers described in Section III-C. Since AdvDiffuser is used for classification models, we use

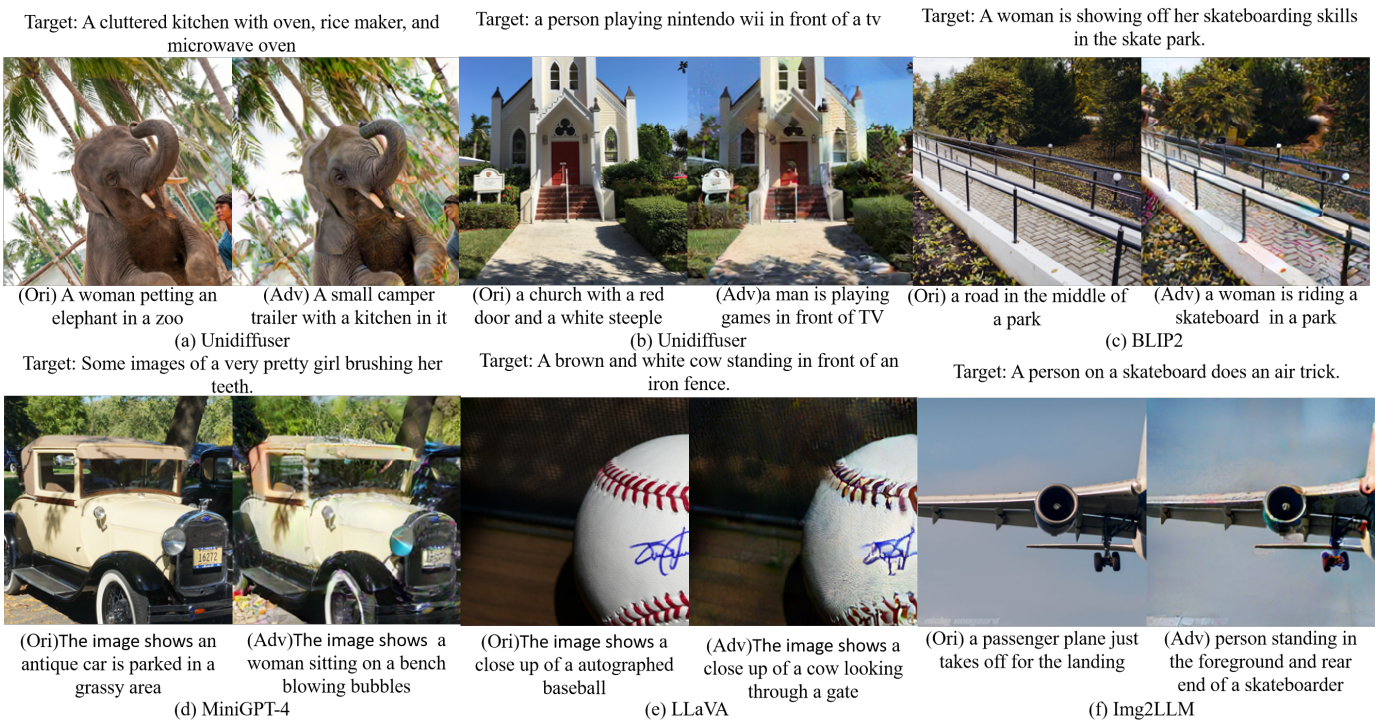


Fig. 4. Visualization of the attack results of our method on various open-source VLMs. We show the adversarial target text above the image, and display the image caption results of original image and adversarial example below the image.

cosine similarity loss instead of classification loss for adversarial attacks on VLMs. For a fair comparison, we implement the ensemble version of AdvDiffuser, including simple ensemble and adaptive ensemble, which are denoted as AdvDiffuser$_{ens}$, AdvDiffuser$_{adaptive}$ respectively. For hyperparameters (in AdvDiffuser), we choose $T = 200, \sigma = 0.4, I = 25$.

Evaluation metrics: Following [26], we adopt CLIP score between the generated responses from victim models and predefined targeted texts, as computed by ViT-B/32 text encoder, refered as CLIP$_{tar}$. We adopt the method of calculating the attack success rate (ASR) in [50], positing that an attack is deemed successful solely if the image description includes the target semantic main object. In order to measure the quality of adversarial examples and the perceptibility of applied perturbations, we use four evaluation metrics: SSIM [59], FID [60], LPIPS [61] and BRISQUE [62].

Implementation details: Since our adversarial diffusion sampling does not require additional training to the original diffusion model, we use the pre-trained diffusion model in our experiment. We adapt LDM [4] with DDIM sampler [54] (the number of diffusion steps $T = 200$). For surrogate models, we select four versions of CLIP [56], namely Resnet50, Resnet101, ViT-B/16 and ViT-B/32. For other hyperparameters, we use $s = 35, \delta = 0.0025, t^* = 0.2, \mathrm{k} = 8, \tau = 2$ and $N = 10$. All the experiments are conducted on a Tesla A100 GPU with 40GB memory.

B. Main Experiments

We first explore the effectiveness of our method in targeted and transferable scenarios. Specifically, we fix the attack scenario as the targeted scenario, and then quantitatively compare the transferability of our method and baselines on open source and commercial VLMs. We also give the time taken by different methods to generate adversarial examples. Finally, we give the qualitative results of our method on open source and commercial VLMs.

TABLE IV
THE RESULT OF ATTACKING COMMERCIAL VLMS. WE REPORT ASR AND PROVIDE THE AVERAGE TIME (S) FOR EACH STRATEGY TO CRAFT A SINGLE x_{adv}

. The best result is **bolded**.

	GPT-4V	Gemini	Copilot	ERNIE Bot	Time(s)
No attack	0%	0%	0%	0%	/
SIA-CWA	35%	12%	25%	50%	732
AdvdiffVLM	**37%**	**17%**	**26%**	**58%**	**139**

Quantitative results on open source VLMs. To validate the effectiveness of AdvDiffVLM, we quantitatively evaluate the transferability of adversarial examples generated by AdvDiffVLM and baseline methods on various open source VLMs. As shown in Table II, all methods demonstrate favorable attack results in gray box scenarios. In the transfer attack scenario, our method yields the best results. For example, on BLIP2, our method improves CLIP$_{tar}$ and ASR by 0.0200 and 10.9% , respectively, when compared to SIA-CWA. Furthermore, our method generates adversarial examples much faster than baselines. Specifically, when compared to AdvDiffuser, SIA and SSA methods, our method generates adversarial examples 5x to 10x faster. Experimental results show that our method generates adversarial examples with better transferability at a faster rate, demonstrating its superiority.

Additionally, it has been observed that AdvDiffuser exhibits suboptimal performance in challenging attack scenarios, particularly against VLMs. This is attributed to its direct application of GradCAM as the mask, which restricts the modifiable

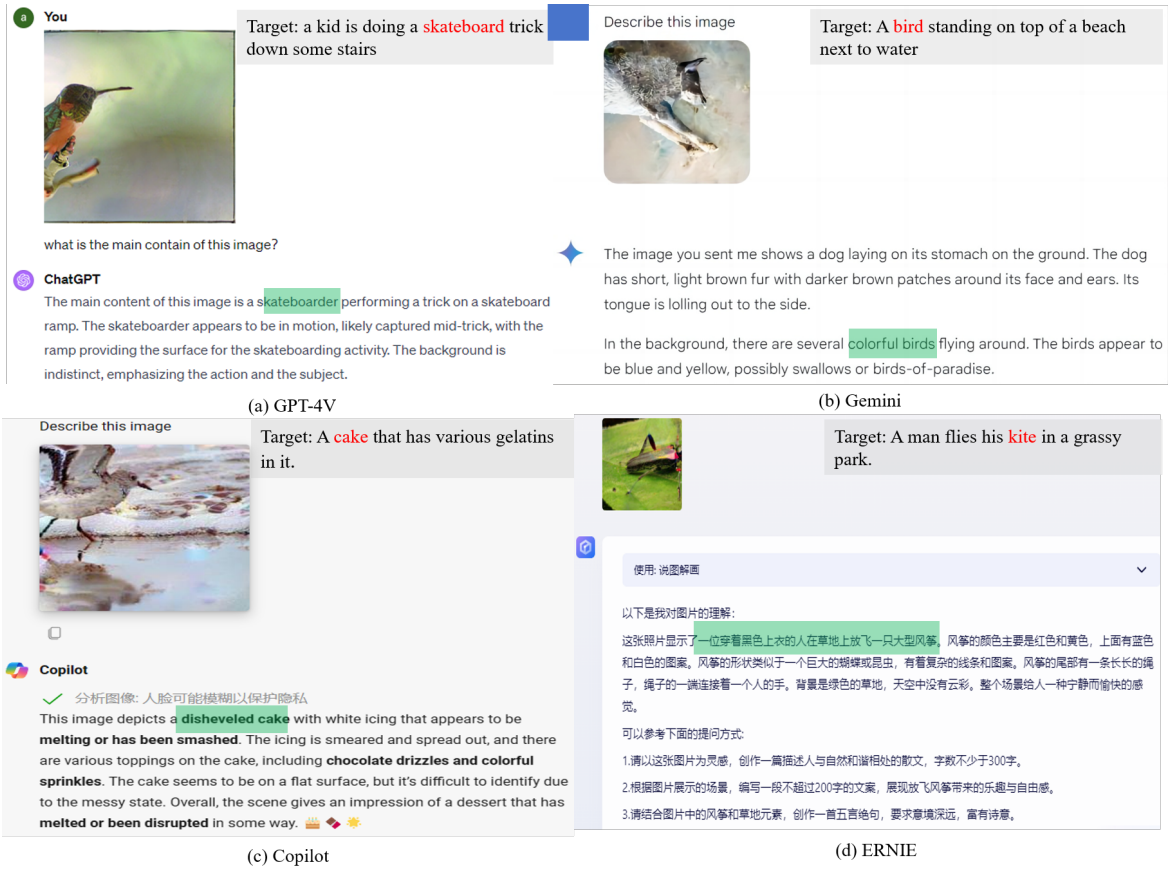


Fig. 5. Screenshots of successful attacks against various commercial VLMs API's image description. We give the adversarial target text on the right side of the image. Else, we mark the main objects of the adversarial target in red and the main objects in the API's response in green.

area for adversarial examples in demanding tasks, thereby diminishing attack effectiveness. Simultaneously, AdvDiffuser employs high-frequency adversarial noise to alter semantics. This adversarial noise, being inherently fragile, is significantly mitigated during the diffusion model's reverse process, further diminishing its attack potential on complex tasks. These observations validate the advantages of our GradCAM-guided Mask and score matching idea.

Quantitative results on commercial VLMs. We conduct a quantitative evaluation of commercial VLMs such as OpenAI's GPT-4V[1], Google's Gemini[2], Microsoft's Copilot[3], and Baidu's ERNIE Bot[4]. We choose SIA-CWA to represent baselines and ASR as an evaluation metric. We chose 100 images from the NeurIPS'17 adversarial competition dataset and 100 text descriptions from the MS-COCO dataset as target texts. Table IV presents the experimental results. Our method outperforms SIA-CWA in terms of attack success rate, demonstrating its superior transferability.

Qualitative results on open source VLMs. We then present visualizations depicting the outcomes of our method's attacks on open source VLMs, as illustrated in Figure 4. Considering the image caption task, we focus on two models: Unidiffuser and BLIP2. Considering the VQA task, we focus on MiniGPT-4, LLaVA and Img2LLM. In the case of MiniGPT-4, the input text is configured as "What is the image showing?". For LLaVA, the input text is set to "What is the main contain of this image?", and the prefix "The main contain is" is omitted in the output. For Img2LLM, the input text is configured as "What is the content of this image?". Our method demonstrates the capability to effectively induce both gray-box and black-box VLMs to produce adversarial target semantics. For example, in the case of LLaVA's attack, we define the adversarial target text as "A cake that has various gelatins in it." LLaVA generate the response "The main contain is a close-up view of a partially eaten cake with chocolate and white frosting." as the target output, while the original image's content is described as "The main contain is a bird, specifically a seagull, walking on the beach near the water.".

Qualitative results on commercial VLMs. We finally show screenshots of successful attacks on various commercial VLMs image description tasks, including Google's Gemini, Microsoft's Copilot, Baidu's ERNIE Bot, and OpenAI's GPT-4V, as shown in Figure 5. These models are large-scale visual language models deployed commercially, and their model configurations and training datasets have not been made public. Moreoever, compared with open source VLMs, these models are equipped with more complex defense mechanisms, making them more difficult to attack. However, as shown in Figure 5, our method successfully induces these commercial VLMs to generate target responses. For example, in GPT-4V, we define the adversarial target text as "a kid is doing a skateboard trick down some stairs." GPT-4V generates the response "The main

[1]https://chat.openai.com/
[2]https://gemini.google.com/
[3]https://copilot.microsoft.com/
[4]https://yiyan.baidu.com/

TABLE V
COMPARISON RESULTS OF DEFENSE EXPERIMENTS WITH SOTA METHOD SIA. WE USE CLIP_{tar} EVALUATION METRIC AND REPORT THE REDUCTION RESULTS OF CLIP_{tar} WHERE THE BEST RESULT IS **BOLDED**. ELSE, THE PARENTHESES REPRESENT THE HYPERPARAMETERS (IN THEIR PAPER).

Defense models	Attack methods	Unidiffuser	BLIP2	MiniGPT-4	LLaVA	Img2LLM
Bit Reduction (4)	SIA-Ens	$0.7204_{\downarrow 0.0173}$	$0.5602_{\downarrow 0.0454}$	$0.5273_{\downarrow 0.0432}$	$0.7034_{\downarrow 0.0124}$	$0.6284_{\downarrow 0.0053}$
	SIA-CWA	$0.7281_{\downarrow 0.0217}$	$0.5798_{\downarrow 0.0435}$	$0.5442_{\downarrow 0.0468}$	$0.7063_{\downarrow 0.0131}$	$0.6375_{\downarrow 0.0026}$
	AdvDiffVLM	$\mathbf{0.7397_{\downarrow 0.0105}}$	$\mathbf{0.6320_{\downarrow 0.0115}}$	$\mathbf{0.6261_{\downarrow 0.0084}}$	$\mathbf{0.7168_{\downarrow 0.0038}}$	$\mathbf{0.6501_{\downarrow 0.0020}}$
STL (k=64, s=8, λ=0.2)	SIA-Ens	$0.7192_{\downarrow 0.0185}$	$0.5571_{\downarrow 0.0485}$	$0.5192_{\downarrow 0.0513}$	$0.6968_{\downarrow 0.0190}$	$0.6230_{\downarrow 0.0107}$
	SIA-CWA	$0.7233_{\downarrow 0.0265}$	$0.5733_{\downarrow 0.0500}$	$0.5385_{\downarrow 0.0525}$	$0.7001_{\downarrow 0.0193}$	$0.6314_{\downarrow 0.0087}$
	AdvDiffVLM	$\mathbf{0.7329_{\downarrow 0.0173}}$	$\mathbf{0.6267_{\downarrow 0.0168}}$	$\mathbf{0.5997_{\downarrow 0.0148}}$	$\mathbf{0.7145_{\downarrow 0.0061}}$	$\mathbf{0.6471_{\downarrow 0.0050}}$
JPEG Compression (p=50)	SIA-Ens	$0.6734_{\downarrow 0.0642}$	$0.5345_{\downarrow 0.0711}$	$0.5002_{\downarrow 0.0703}$	$0.6542_{\downarrow 0.0616}$	$0.6020_{\downarrow 0.0317}$
	SIA-CWA	$0.6801_{\downarrow 0.0697}$	$0.5525_{\downarrow 0.0708}$	$0.5273_{\downarrow 0.0637}$	$0.6550_{\downarrow 0.0644}$	$0.6088_{\downarrow 0.0313}$
	AdvDiffVLM	$\mathbf{0.6896_{\downarrow 0.0606}}$	$\mathbf{0.6218_{\downarrow 0.0217}}$	$\mathbf{0.5865_{\downarrow 0.0380}}$	$\mathbf{0.6983_{\downarrow 0.0223}}$	$\mathbf{0.6354_{\downarrow 0.0167}}$
DISCO (s=3, k=5)	SIA-Ens	$0.6087_{\downarrow 0.1290}$	$0.5134_{\downarrow 0.0922}$	$0.4986_{\downarrow 0.0719}$	$0.6274_{\downarrow 0.0884}$	$0.5771_{\downarrow 0.0566}$
	SIA-CWA	$0.6114_{\downarrow 0.1384}$	$0.5290_{\downarrow 0.0943}$	$0.5114_{\downarrow 0.0796}$	$0.6331_{\downarrow 0.0863}$	$0.5842_{\downarrow 0.0559}$
	AdvDiffVLM	$\mathbf{0.6215_{\downarrow 0.1287}}$	$\mathbf{0.5892_{\downarrow 0.0543}}$	$\mathbf{0.5727_{\downarrow 0.0418}}$	$\mathbf{0.6728_{\downarrow 0.0478}}$	$\mathbf{0.6093_{\downarrow 0.0428}}$
DISCO+JPEG	SIA-Ens	$0.5642_{\downarrow 0.1735}$	$0.5025_{\downarrow 0.1031}$	$0.4878_{\downarrow 0.0827}$	$0.6067_{\downarrow 0.1091}$	$0.5681_{\downarrow 0.0656}$
	SIA-CWA	$0.5735_{\downarrow 0.1763}$	$0.5176_{\downarrow 0.1057}$	$0.5074_{\downarrow 0.0836}$	$0.6106_{\downarrow 0.1088}$	$0.5692_{\downarrow 0.0709}$
	AdvDiffVLM	$\mathbf{0.5924_{\downarrow 0.1578}}$	$\mathbf{0.5859_{\downarrow 0.0576}}$	$\mathbf{0.5650_{\downarrow 0.0495}}$	$\mathbf{0.6724_{\downarrow 0.0482}}$	$\mathbf{0.6081_{\downarrow 0.0440}}$
DiffPure (t* = 0.15)	SIA-Ens	$0.4921_{\downarrow 0.2456}$	$0.5048_{\downarrow 0.1008}$	$0.4919_{\downarrow 0.0786}$	$0.5356_{\downarrow 0.1802}$	$0.5372_{\downarrow 0.0965}$
	SIA-CWA	$0.4942_{\downarrow 0.2556}$	$0.5099_{\downarrow 0.1136}$	$0.5025_{\downarrow 0.0885}$	$0.5360_{\downarrow 0.1835}$	$0.5388_{\downarrow 0.1013}$
	AdvDiffVLM	$\mathbf{0.5837_{\downarrow 0.1665}}$	$\mathbf{0.5527_{\downarrow 0.0908}}$	$\mathbf{0.5506_{\downarrow 0.0639}}$	$\mathbf{0.5857_{\downarrow 0.1349}}$	$\mathbf{0.5711_{\downarrow 0.0810}}$

content of this image is a skateboarder performing a trick on a skateboard ramp...", while the semantics of the original image is "A bird standing on a branch." Moreover, our method is also applicable to various languages. For example, we use English to generate adversarial examples but successfully attack ERNIE Bot, which operates in Chinese.

Qualitative and quantitative experimental results show that our method can generate targeted adversarial examples with better transferability 5x to 10x faster, demonstrating the superiority of our method.

C. More Experiments

Against adversarial defense models. Our method achieves superior attack performance on both open source and commercial VLMs. In recent years, various adversarial defense methods have been proposed to mitigate the threat of adversarial examples. To verify the superiority of our method against these defense methods, we conduct various experiments on Bit Reduction [63], STL [64], JPEG Compression [65], DISCO [66], JPEG+DISCO, and DiffPure [67]. We report the CLIP_{tar} metric. At the same time, we report the CLIP_{tar} reduction results, which more accurately reflect the ability of adversarial examples to resist defense methods. The experimental results are shown in Table V. It can be observed that, for all defense methods, both CLIP_{tar} and CLIP_{tar} reduction results of our methods outperform the baselines. This demonstrates the superiority of our method against defense methods compared to baselines.

To better evaluate the resistance of our method against adversarial defense methods, we further in detail show the results of the SOTA defense method, namely DiffPure, in Table VI. It can be found that our method outperforms baselines in both gray-box and black-box settings. For example, on Unidiffuser, for CLIP_{tar} score, our method is 0.0895 higher than SIA-CWA. On BLIP2, for CLIP_{tar} score, our method is 0.0428 higher than SIA-CWA. Furthermore, in all cases, the attack success rate of our methods is higher than the baselines. These experimental results demonstrate that our method outperforms baselines in evading the DiffPure defense method.

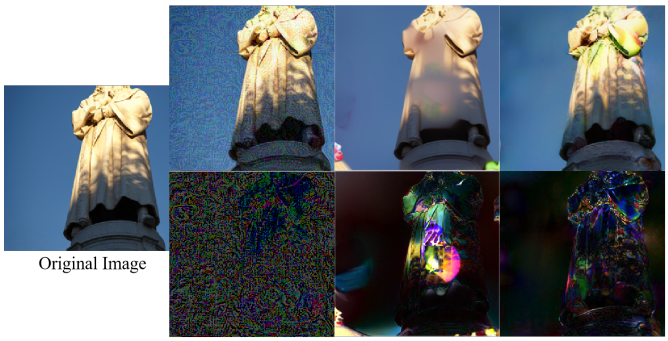


Fig. 6. Visualization of adversarial perturbations generated by different attack methods. Note that the first row represents adversarial examples, and the second row represents adversarial perturbations. We choose SIA-CWA and $\text{AdvDiffuser}_{adaptive}$ as representatives of baselines. We amplify the perturbation values for better visualization.

We can break the SOTA defense method Diffpure with an attack success rate of more than 10% in a completely black-box scenario, exposing the flaws in current defense methods and raising new security concerns for designing more robust deep learning models.

Image quality comparison. The image quality of adversarial examples is also particularly important. Adversarial examples with poor image quality can be easily detected. We further evaluate the image quality of the generated adversarial examples using four evaluation metrics: SSIM, FID, LPIPS, and BRISQUE. As shown in Table VII, compared to baselines, the adversarial examples generated by our method exhibit higher image quality. Specifically, our results are significantly better than the baselines in terms of SSIM, LPIPS, and FID evaluation metrics. For the BRISQUE metric, AdvDiffuser outperforms our method. This is because BRISQUE is a reference-free image quality assessment algorithm and

TABLE VI
DEFENSE RESULTS WITH DIFFPURE. THE SETTING ARE THE SAME AS TABLE II EXCEPT THE ADVERSARIAL EXAMPLES ARE PURIFIED BY DIFFPURE. IN THIS TABLE, CLIP_{tar} EVALUATES THE SIMILARITY BETWEEN THE RESULTS OF PURIFIED EXAMPLES AND THE TARGET TEXTS.

	Unidiffuser*		BLIP2		MiniGPT-4		LLaVA		Img2LLM	
	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑	$\text{CLIP}_{tar}\uparrow$	ASR ↑
Original	0.4802	0.0%	0.4924	0.0%	0.4831	0.0%	0.5253	0.0%	0.5302	0.0%
Ens	0.4833	0.0%	0.4929	0.0%	0.4840	0.0%	0.5263	0.0%	0.5332	0.0%
SVRE	0.4846	0.7%	0.4953	0.0%	0.4852	0.0%	0.5264	0.0%	0.5312	0.0%
CWA	0.4873	2.1%	0.4973	0.0%	0.4901	1.0%	0.5272	0.8%	0.5307	0.0%
SSA-Ens	0.4914	0.9%	0.5024	0.0%	0.4916	0.0%	0.5280	1.2%	0.5322	0.0%
SSA-SVRE	0.4899	2.1%	0.4984	0.2%	0.4918	0.0%	0.5273	1.2%	0.5356	0.0%
SSA-CWA	0.4868	2.5%	0.4997	0.0%	0.4997	0.0%	0.5283	2.8%	0.5367	0.7%
SIA-Ens	0.4921	3.7%	0.5048	1.2%	0.4919	1.1%	0.5356	2.5%	0.5372	1.6%
SIA-SVRE	0.4930	3.9%	0.5012	1.8%	0.5011	1.6%	0.5349	4.2%	0.5380	2.5%
SIA-CWA	0.4942	5.8%	0.5099	2.6%	0.5025	2.2%	0.5360	4.0%	0.5388	1.5%
AdvDiffuser_{ens}	0.4920	4.2%	0.4933	2.6%	0.4906	2.4%	0.5325	3.7%	0.5310	2.7%
$\text{AdvDiffuser}_{adaptive}$	0.4922	4.5%	0.5001	3.2%	0.5001	3.2%	0.5336	3.4%	0.5325	2.8%
AdvDiffVLM	**0.5837**	**22.4%**	**0.5527**	**10.2%**	**0.5506**	**12.6%**	**0.5857**	**18.0%**	**0.5711**	**10.5%**

TABLE VII
QUALITY COMPARISON OF ADVERSARIAL EXAMPLES UNDER FOUR EVALUATION METRICS. THE BEST RESULT IS **BOLDED**.

Method	SSIM ↑	LPIPS ↓	FID ↓	BRISQUE ↓
SSA-Ens	0.6687	0.3320	110.5	66.89
SSA-SVRE	0.6610	0.3325	112.6	70.05
SSA-CWA	0.6545	0.3673	123.4	67.67
SIA-Ens	0.6925	0.2990	117.3	55.61
SIA-SVRE	0.6920	0.3042	120.0	57.42
SIA-CWA	0.6892	0.3306	125.3	56.02
AdvDiffuser_{ens}	0.6520	0.3074	115.5	**14.61**
$\text{AdvDiffuser}_{adaptive}$	0.6471	0.3096	126.7	15.32
AdvDiffVLM	**0.6992**	**0.2930**	**107.4**	32.96

TABLE VIII
COMPARISON OF IMAGE QUALITY OF ADVERSARIAL EXAMPLES BEFORE AND AFTER USING THE GRADCAM-GUIDED MASK MODULE. THE BEST RESULT IS **BOLDED**.

Method	SSIM ↑	LPIPS ↓	FID ↓	BRISQUE ↓
w/o mask	0.7129	0.2687	111.9	16.92
Ours	**0.7188**	**0.2358**	**96.1**	**16.80**

is sensitive to blur, noise, color change, etc. As shown in Figure 6, the adversarial examples generated by AdvDiffuser lack obvious abnormalities in these elements, so its results are marginally better than our method. However, as shown in Figure 6, the perturbation introduced by our method is semantic, while AdvDiffuser significantly alters the non-salient area, resulting in poor visual effects. This shows that the adversarial examples generated by AdvDiffuser are unsuitable for more complex scenarios, such as attacking VLMs. In addition, it can be seen that the adversarial examples generated by the transfer-based methods exhibit significant noise, indicating that our method has obvious superiority in terms of image quality.

D. Ablation Experiments

To further understand the effectiveness of the proposed method, we discuss the role of each module. We set $N = 1$ to more conveniently discuss the impact of each module. We consider three cases, including using only a single ViT-B/32 to calculate the loss, using a simple ensemble strategy, and not using the GradCAM-guided Mask module, named Single, Ens, and w/o mask respectively.

Is Adaptive Ensemble Gradient Estimation module beneficial for boosting the attack capability? We first explore whether the Adaptive Ensemble Gradient Estimation module could help improve the transferability and robustness of adversarial examples. We divide the Adaptive Ensemble Gradient Estimation module into two approaches, Single and Ens, and maintain all other conditions constant. The results are shown in Figure 7(a) and (b). It is observable that the ensemble method exhibits better performance in transferability and robustness compared to the single loss method. Furthermore, the performance of the adaptive ensemble method is enhanced compared to the basic ensemble method. The experimental results demonstrate that the Adaptive Ensemble Gradient Estimation module enhances the transferability and robustness of adversarial examples.

Does GradCAM-guided Mask module help trade-off image quality and attack capability? Next, we explore the role of the GradCAM-guided Mask module in balancing image quality and transferability. We compare this with the w/o mask method, and the results are presented in Figure 7. As shown in Figure 7(a) and (b), the use of the GradCAM-guided Mask module results in a slight decrease in the transferability and robustness of the adversarial examples. However, as shown in Figure 7(c), the absence of the GradCAM-guided Mask module leads to the adversarial examples exhibiting obvious target features, and the use of the GradCAM-guided Mask module enhances the visual quality of the adversarial example. In addition, Table VIII further shows that the GradCAM-guided Mask module can improve the visual quality of adversarial examples. The experimental results demonstrate that the GradCAM-guided Mask module effectively balances the visual quality and attack capability of the adversarial examples.

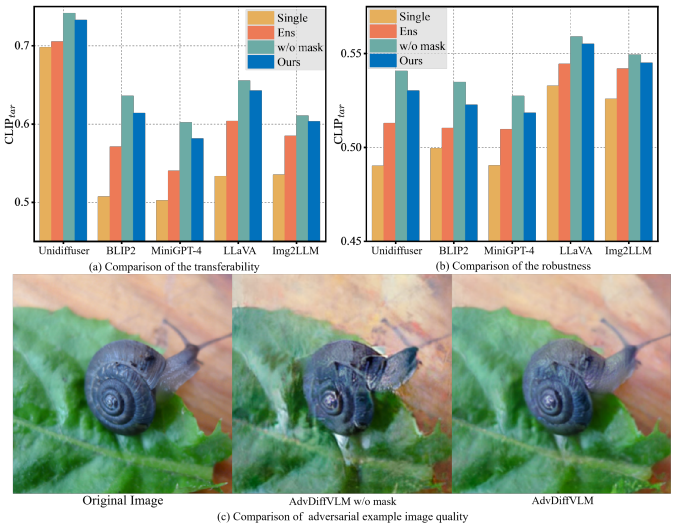


Fig. 7. Comparison results of different ablation methods. Here, "Single" means using a single ViT-B/32 to calculate the loss, "Ens" means using the simple ensemble strategy, and "w/o mask" means not using GradCAM-guided Mask module.

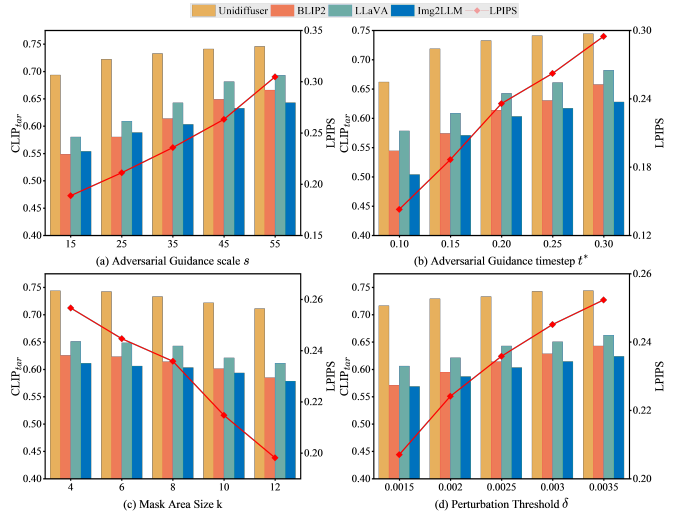


Fig. 8. Ablation study of the impact of inner loop hyperparameters. We adopt the CLIP$_{tar}$ and LPIPS scores to show the impact of transferability and image quality with four VLMs. A higher CLIP$_{tar}$ value indicates better performance, whereas a lower LPIPS value signifies better results. We only vary one of the hyperparameters at a time, and then fix the other three hyperparameters to the preset values shown in Section V-A. Note: the results of CLIP$_{tar}$ are presented using bar graphs, while LPIPS results are depicted using dot-line graphs.

E. Hyperparameter Studies

In this subsection, we conduct hyperparameter studies to explore the impact of hyperparameters, including inner loop hyperparameters s, t^*, k, and δ and outer loop hyperparameter N.

The impacts of inner loop hyperparameters. We first discuss the impacts of inner loop hyperparameters (including the s, t^*, k, and δ). We set $N = 1$ and conducting tests on Unidiffuser, BLIP2, LLaVA and Img2LLM. The experimental results are shown in Figure 8. It is evident that all parameters influence the trade-off between transferability and image quality. Increasing values for parameters s, t^*, and δ enhance transferability but diminish the visual quality of adversarial examples. This is because larger values for these parameters result in a greater perturbation, allowing for the embedding of more adversarial semantics into the image. Conversely, increasing the value of k produces adversarial examples with improved visual effects but reduces transferability. The reason is that larger values of k result in a larger generated mask, making it more challenging to modify the important areas in the image. To achieve an optimal trade-off between transferability and image quality, we empirically select $s = 35, t^* = 0.2, \mathrm{k} = 8$ and $\delta = 0.0025$.

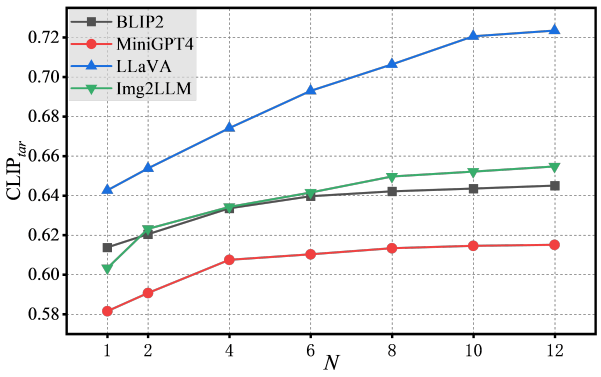


Fig. 9. Transferability of adversarial examples on various black-box VLMs as N changes from 1 to 12.

The impact of outer loop hyperparameter. Next, we investigate the impact of the outer hyperparameter N on the transferability of adversarial examples. We conduct experiments on BLIP2, MiniGPT4, LLaVA, and Img2LLM with $s = 35, t^* = 0.2, \mathrm{k} = 8$, and $\delta = 0.0025$. The results show that N improves the transferability of adversarial examples, but the improvement gradually fades. Specifically, the increase in transferability is limited after $N = 6, 6, 8, 10$ for BLIP2, MiniGPT4, Img2LLM, and LLaVA. Given that increasing N increases the computational cost, we choose $N = 10$ to strike a balance between transferability and cost.

VI. Conclusion

In this work, we propose AdvDiffVLM, an unrestricted and targeted adversarial example generation method for VLMs. We design the Adaptive Ensemble Gradient Estimation based on the idea of score matching. It embeds the target semantics into adversarial examples, which can generate targeted adversarial examples with better transferability faster. At the same time, in order to achieve a trade-off between adversarial example quality and attack capabilities, we proposed the GradCAM-guided Mask method. Finally, we embed more target semantics into adversarial examples using multiple iterations. Extensive experiments demonstrate that our method can generate targeted adversarial examples 5x to 10x faster than baselines, while also achieving better transferability. Our research can discover vulnerabilities in open-source VLMs and commercial VLMs, providing insights for developing more robust and trustworthy VLMs.

Impact Statements

Our research mainly aims to discover vulnerabilities in open-source large VLMs and commercial VLMs such as GPT-

4V, providing insights for developing more robust and trustworthy VLMs. However, our attack methods can be abused to evade actual deployed commercial systems, causing potential negative social impacts. For example, criminals may use our methods to cause GPT-4V APIs to output target responses, causing serious harm.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grant 61972312.

REFERENCES

[1] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International Conference on Machine Learning*. PMLR, 2022, pp. 12 888–12 900.

[2] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models," in *International Conference on Machine Learning*, 2023, pp. 1–13.

[3] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: https://openreview.net/forum?id=w0H2xGHlkw

[4] 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.

[5] F. Bao, S. Nie, K. Xue, C. Li, S. Pu, Y. Wang, G. Yue, Y. Cao, H. Su, and J. Zhu, "One transformer fits all distributions in multi-modal diffusion at scale," in *International Conference on Machine Learning*, 2023, pp. 1692–1717.

[6] S. H. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "Chatgpt for robotics: Design principles and model abilities," *IEEE Access*, 2024.

[7] G. Liao, J. Li, and X. Ye, "Vlm2scene: Self-supervised image-text-lidar learning with foundation models for autonomous driving scene understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 4, 2024, pp. 3351–3359.

[8] C. Cui, Y. Ma, X. Cao, W. Ye, Y. Zhou, K. Liang, J. Chen, J. Lu, Z. Yang, K.-D. Liao *et al.*, "A survey on multimodal large language models for autonomous driving," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 958–979.

[9] P. Nagesh, B. Prabha, S. B. Gole, G. Rao, and N. V. Ramana, "Visual assistance for visually impaired people using image caption and text to speech," in *AIP Conference Proceedings*, vol. 2512, no. 1. AIP Publishing, 2024.

[10] W. Wang, J. Huang, J.-t. Huang, C. Chen, J. Gu, P. He, and M. R. Lyu, "An image is worth a thousand toxic words: A metamorphic testing framework for content moderation software," in *2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 2023, pp. 1339–1351.

[11] D. Han, X. Jia, Y. Bai, J. Gu, Y. Liu, and X. Cao, "Ot-attack: Enhancing adversarial transferability of vision-language models via optimal transport optimization," *arXiv preprint arXiv:2312.04403*, 2023.

[12] S. Gao, X. Jia, X. Ren, I. Tsang, and Q. Guo, "Boosting transferability in vision-language attacks via diversification along the intersection region of adversarial trajectory," *arXiv preprint arXiv:2403.12445*, 2024.

[13] X. Gu, X. Zheng, T. Pang, C. Du, Q. Liu, Y. Wang, J. Jiang, and M. Lin, "Agent smith: A single image can jailbreak one million multimodal llm agents exponentially fast," *arXiv preprint arXiv:2402.08567*, 2024.

[14] J. Zheng, C. Lin, J. Sun, Z. Zhao, Q. Li, and C. Shen, "Physical 3d adversarial attacks against monocular depth estimation in autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 24 452–24 461.

[15] M. T. West, S.-L. Tsang, J. S. Low, C. D. Hill, C. Leckie, L. C. Hollenberg, S. M. Erfani, and M. Usman, "Towards quantum enhanced adversarial robustness in machine learning," *Nature Machine Intelligence*, vol. 5, no. 6, pp. 581–589, 2023.

[16] Y. Zhao, T. Pang, C. Du, X. Yang, C. Li, N.-M. M. Cheung, and M. Lin, "On evaluating adversarial robustness of large vision-language models," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[17] X. Jia, Y. Chen, X. Mao, R. Duan, J. Gu, R. Zhang, H. Xue, Y. Liu, and X. Cao, "Revisiting and exploring efficient fast adversarial training via law: Lipschitz regularization and auto weight averaging," *IEEE Transactions on Information Forensics and Security*, 2024.

[18] X. Jia, Y. Zhang, X. Wei, B. Wu, K. Ma, J. Wang, and X. Cao, "Improving fast adversarial training with prior-guided knowledge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[19] N. Aafaq, N. Akhtar, W. Liu, M. Shah, and A. Mian, "Language model agnostic gray-box adversarial attack on image captioning," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 626–638, 2022.

[20] Y. Xu, B. Wu, F. Shen, Y. Fan, Y. Zhang, H. T. Shen, and W. Liu, "Exact adversarial attack to image captioning via structured output learning with latent variables," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4135–4144.

[21] R. Lapid and M. Sipper, "I see dead people: Gray-box adversarial attack on image-to-text models," in *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2023.

[22] A. E. Baia, V. Poggioni, and A. Cavallaro, "Black-box attacks on image activity prediction and its natural language explanations," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3686–3695.

[23] L. Zhu, T. Wang, J. Li, Z. Zhang, J. Shen, and X. Wang, "Efficient query-based black-box attack against cross-modal hashing retrieval," *ACM Transactions on Information Systems*, vol. 41, no. 3, pp. 1–25, 2023.

[24] P. N. Williams and K. Li, "Black-box sparse adversarial attack via multi-objective optimisation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 291–12 301.

[25] H. Zhu, X. Sui, Y. Ren, Y. Jia, and L. Zhang, "Boosting transferability of targeted adversarial examples with non-robust feature alignment," *Expert Systems with Applications*, vol. 227, p. 120248, 2023.

[26] Y. Zhao, T. Pang, C. Du, X. Yang, C. Li, N.-M. Cheung, and M. Lin, "On evaluating adversarial robustness of large vision-language models," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[27] H. Chen, Y. Zhang, Y. Dong, and J. Zhu, "Rethinking model ensemble in transfer-based adversarial attacks," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=AcJrSoArlh

[28] 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.

[29] X. Wang, Z. Zhang, and J. Zhang, "Structure invariant transformation for better adversarial transferability," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4607–4619.

[30] Y. Wang, Y. Wu, S. Wu, X. Liu, W. Zhou, L. Zhu, and C. Zhang, "Boosting the transferability of adversarial attacks with frequency-aware perturbation," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 6293–6304, 2024.

[31] Y. Song, R. Shu, N. Kushman, and S. Ermon, "Constructing unrestricted adversarial examples with generative models," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 8322–8333.

[32] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=H1BLjgZCb

[33] X. Chen, X. Gao, J. Zhao, K. Ye, and C.-Z. Xu, "Advdiffuser: Natural adversarial example synthesis with diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4562–4572.

[34] A. S. Shamsabadi, R. Sanchez-Matilla, and A. Cavallaro, "Colorfool: Semantic adversarial colorization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1151–1160.

[35] 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. [Online]. Available: https://openreview.net/forum?id=rJzIBfZAb

[36] J. Guo, J. Li, D. Li, A. M. H. Tiong, B. Li, D. Tao, and S. Hoi, "From images to textual prompts: Zero-shot visual question answering with frozen large language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 867–10 877.

[37] 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 Con-*

ference on Computer Vision and Pattern Recognition, 2018, pp. 9185–9193.

[38] Y. Long, Q. Zhang, B. Zeng, L. Gao, X. Liu, J. Zhang, and J. Song, "Frequency domain model augmentation for adversarial attack," in *European Conference on Computer Vision*. Springer, 2022, pp. 549–566.

[39] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.

[40] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[41] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[42] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez *et al.*, "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023.

[43] S. Yin, C. Fu, S. Zhao, K. Li, X. Sun, T. Xu, and E. Chen, "A survey on multimodal large language models," *arXiv preprint arXiv:2306.13549*, 2023.

[44] J. Wu, W. Gan, Z. Chen, S. Wan, and S. Y. Philip, "Multimodal large language models: A survey," in *2023 IEEE International Conference on Big Data (BigData)*. IEEE, 2023, pp. 2247–2256.

[45] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds *et al.*, "Flamingo: a visual language model for few-shot learning," *Advances in neural information processing systems*, vol. 35, pp. 23 716–23 736, 2022.

[46] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592*, 2023.

[47] J. C. Costa, T. Roxo, H. Proença, and P. R. Inácio, "How deep learning sees the world: A survey on adversarial attacks & defenses," *IEEE Access*, 2024.

[48] S. Han, C. Lin, C. Shen, Q. Wang, and X. Guan, "Interpreting adversarial examples in deep learning: A review," *ACM Computing Surveys*, vol. 55, no. 14s, pp. 1–38, 2023.

[49] J. Gu, X. Jia, P. de Jorge, W. Yu, X. Liu, A. Ma, Y. Xun, A. Hu, A. Khakzar, Z. Li *et al.*, "A survey on transferability of adversarial examples across deep neural networks," *arXiv preprint arXiv:2310.17626*, 2023.

[50] Y. Dong, H. Chen, J. Chen, Z. Fang, X. Yang, Y. Zhang, Y. Tian, H. Su, and J. Zhu, "How robust is google's bard to adversarial image attacks?" *arXiv preprint arXiv:2309.11751*, 2023.

[51] Q. Li, Q. Hu, H. Fan, C. Lin, C. Shen, and L. Wu, "Attention-sa: Exploiting model-approximated data semantics for adversarial attack," *IEEE Transactions on Information Forensics and Security*, pp. 1–1, 2024.

[52] D.-T. Peng, J. Dong, M. Zhang, J. Yang, and Z. Wang, "Csfadv: Critical semantic fusion guided least-effort adversarial example attacks," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 5940–5955, 2024.

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

[54] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[55] C. Luo, "Understanding diffusion models: A unified perspective," *arXiv preprint arXiv:2208.11970*, 2022.

[56] 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.

[57] Z. Cai, Y. Tan, and M. S. Asif, "Ensemble-based blackbox attacks on dense prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4045–4055.

[58] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.

[59] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[60] 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," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 6629–6640.

[61] 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.

[62] A. Mittal, A. K. Moorthy, and A. C. Bovik, "Blind/referenceless image spatial quality evaluator," in *2011 conference record of the forty fifth asilomar conference on signals, systems and computers (ASILOMAR)*. IEEE, 2011, pp. 723–727.

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

[64] B. Sun, N.-h. Tsai, F. Liu, R. Yu, and H. Su, "Adversarial defense by stratified convolutional sparse coding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 11 447–11 456.

[65] G. K. Dziugaite, Z. Ghahramani, and D. M. Roy, "A study of the effect of jpg compression on adversarial images," *arXiv preprint arXiv:1608.00853*, 2016.

[66] C.-H. Ho and N. Vasconcelos, "Disco: Adversarial defense with local implicit functions," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 818–23 837, 2022.

[67] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, "Diffusion models for adversarial purification," in *International Conference on Machine Learning*, 2022, pp. 1–23.