This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

PA-Attack: Guiding Gray-Box Attacks on LVLM Vision Encoders with Prototypes and Attention

Hefei Mei[1] Zirui Wang[1] Chang Xu[2] Jianyuan Guo[1] Minjing Dong[1]*
[1]City University of Hong Kong [2]The University of Sydney
{hefeimei2-c, zrwang23-c}@my.cityu.edu.hk
c.xu@sydney.edu.au, {jianyguo, minjdong}@cityu.edu.hk

Abstract

Large Vision–Language Models (LVLMs) are foundational to modern multimodal applications, yet their susceptibility to adversarial attacks remains a critical concern. Prior white-box attacks rarely generalize across tasks, and black-box methods depend on expensive transfer, which limits efficiency. The vision encoder, standardized and often shared across LVLMs, provides a stable gray-box pivot with strong cross-model transfer. Building on this premise, we introduce ***PA-Attack*** *(Prototype-Anchored Attentive Attack). PA-Attack begins with a prototype-anchored guidance that provides a stable attack direction towards a general and dissimilar prototype, tackling the attribute-restricted issue and limited task generalization of vanilla attacks. Building on this, we propose a two-stage attention enhancement mechanism: (i) leverage token-level attention scores to concentrate perturbations on critical visual tokens, and (ii) adaptively recalibrate attention weights to track the evolving attention during the adversarial process. Extensive experiments across diverse downstream tasks and LVLM architectures show that PA-Attack achieves an average* 75.1% *score reduction rate (SRR), demonstrating strong attack effectiveness, efficiency, and task generalization in LVLMs. Code is available at https://github.com/hefeimei06/PA-Attack.*

1. Introduction

Large Vision–Language Models (LVLMs) integrate visual and linguistic modalities, enabling a new generation of multimodal applications [4, 23, 31, 37, 43]. As LVLMs are increasingly deployed in real-world applications, it is imperative to understand their robustness and security vulnerabilities [22, 36, 41]. Adversarial attacks, which induce model failures by adding imperceptible perturbations to inputs, pose a significant threat to LVLMs [5, 21, 29, 38, 40].

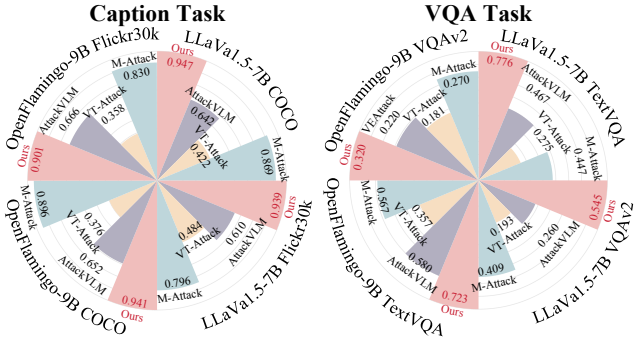
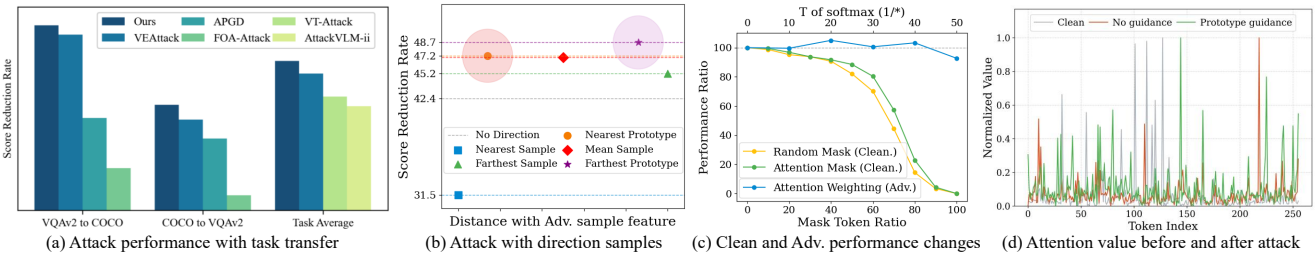


Figure 1. Adversarial performance on captioning and VQA tasks. The perturbation of black-box M-attack is $\epsilon = 16/255$ while that of other gray-box methods is $\epsilon = 2/255$.

Two stubborn challenges for attacking LVLMs lie in efficiency and task generalization [22], especially as the complexity of large language models (LLMs) and task diversity continue to grow. Existing white-box attacks [9, 10, 25, 29] assume full access to massive LVLM parameters but cannot generate adversarial examples that generalize to different LVLM tasks. The strong assumption and low efficiency reduce the attacking practicability. Meanwhile, black-box attacks [16, 18, 40, 42] explored various transfer strategies to achieve high attack transferability among LVLM tasks. However, these strategies always lead to high computational cost and assume large perturbation sizes that reduce stealthiness. As shown in Fig. 1, black-box M-Attack [18] struggles to achieve a high score reduction rate (SRR) of LVLM performance, despite employing larger perturbations.

By contrast, a gray-box attack targeting a foundational LVLM module provides better trade-offs among efficiency, transferability, and practicality since only partial parameters are required and overfitting to a specific LVLM task is relaxed. In particular, attacking the vision encoder offers several advantages. First, most LVLMs, such as LLaVA, Yi-VL, DeepSeek-VL [23, 24, 39], are built by pairing a shared vision backbone (e.g., CLIP [28]) with different LLMs, *i.e.*,

*Corresponding author



(a) Attack performance with task transfer (b) Attack with direction samples (c) Clean and Adv. performance changes (d) Attention value before and after attack

Figure 2. (a) **Comparison of task transfer performance score reduction rate (SRR) on LLaVa1.5-7B.** Slash columns compare the task transfer with white-box and black-box attack, while dotted columns represent the multi-task SRR of gray-box. (b) **Attack SRR w/ and w/o different direction samples.** The red diamond is the baseline without direction samples, and the light orange and purple circles represent their centers as prototypes. (c) **The ratio of clean and adversarial performance changes.** The blue line has the horizontal axis above (T in softmax) as a variable, and the other lines have the horizontal axis below (the proportion of mask tokens) as a variable. (d) **Attentions before and after a 50-step attack w/ and w/o prototype guidance.** For comparison, values are normalized with the maximum value.

Vicuna, Yi-LLM, DeepSeek-LLM [6, 8, 39], which makes the vision encoder a common component across diverse LVLMs and an ideal target for attack. Second, the vision encoder typically contains significantly fewer parameters than LLM modules, thereby boosting efficiency. Third, perturbations crafted to disrupt the vision encoder could be more generalizable across LVLM tasks since all downstream multimodal tasks rely on fundamental visual representations.

However, existing gray-box methods suffer from trade-offs between efficiency and effectiveness. VT-Attack [35] requires auxiliary text information from LVLM captions and high attack iterations, leading to significant costs. Meanwhile, AttackVLM-ii [42] simply utilizes cosine similarity supervision targeting the class token, which becomes ineffective. As shown in Fig. 1 and Fig. 2 (a), with lower attack iterations and perturbations, it is difficult for these gray-box attacks to attack LVLMs in different tasks. We mainly attribute it to the attack generalization. Existing gray-box attacks typically maximize the discrepancy between adversarial features and clean ones. However, these attacks always overfit to limited visual attributes without guidance. As shown in the red line of Fig. 2 (d), a few specific tokens dominate the attack. Fig. 4 also shows that the generated adversarial example can hardly generalize to tasks that focus on different visual attributes. In addition, Figure 2 (c) indicates significant token redundancy in performance impact. Existing approaches treat all tokens uniformly, which expands the search space and wastes effort on redundant tokens, thereby becoming ineffective.

In this paper, we propose a novel gray-box attack, PA-Attack (Prototype-Anchored Attentive Attack), to explore the potential of vision encoders for generalized attack in various LVLM tasks. To address the attack generalization issue, we first introduce prototype-anchored guidance to provide a more general attack direction targeting more visual attributes. Simply maximizing an objective without guidance could easily lead to overfitting to a few attributes, limiting the attack's generalization across diverse LVLM tasks. Instead of merely maximizing dissimilarity, we provide a stable directional guide by optimizing the adversarial features towards a pre-computed, highly dissimilar prototype derived from a diverse guidance dataset. Furthermore, to overcome high-dimensional feature redundancy that hinders effective search for adversarial visual features, we then employ token attention enhancement. This mechanism uses class token attention scores as weights to focus the attack on the most critical patch tokens. Because attention patterns shift during the attack, as shown in Fig. 2 (d), we design a two-stage attention refinement framework. This approach recalculates attention weights from the first-stage adversarial sample, adaptively aligning the optimization with the evolving state of the adversarial process to improve disruption. Extensive experiments demonstrate that PA-Attack achieves superior attack effectiveness with 75.1% score reduction rate (SRR) across diverse downstream tasks and exhibits strong generalization when evaluated on multiple LVLMs. Our results highlight the value of vision-encoder-centric attacks and advance gray-box LVLM attack designs.

2. Related work

Adversarial attacks with full model access on LVLMs. White-box attacks on LVLMs assume full access to model components and typically adapt classic gradient-based methods [7, 9, 12, 25] to optimize inputs with imperceptible budgets. APGD [29] shows that imperceptible perturbations can steer captions to misinformation or malicious links even with tiny perturbation budgets. Cui et al. [10] investigate the robustness of perturbations across prompts, concluding that LVLMs are non robust to visual perturbations and using context in the prompt mitigates failures. UMK [34] jointly learns an adversarial image prefix and a text suffix, outperforming uni-modal baselines. Due to their reliance on complete gradient information, the adversarial examples they generate often struggle to achieve general-

ization across the diverse tasks of LVLMs.

Adversarial attacks with limited model access on LVLMs. Research on generalizable LVLM attacks is increasingly focused on limited-access black-box and gray-box settings. In black-box attacks, AttackVLM [42] establishes an effective transfer strategy that aligns image-to-image features from surrogate models. SSA-CWA [11] demonstrates attacks against commercial, closed-source MLLMs like Google's Bard [13]. AnyAttack [40] introduces a self-supervised framework, pre-training a noise generator to create attacks independent of target labels. M-Attack [18] proposes an effective baseline using random cropping to embed local semantic details. FOA-Attack [16] jointly aligns global features with local patch-tokens using clustering and Optimal Transport. However, their reliance on complex transfer strategies can reduce efficiency. Gray-box attacks have emerged to balance generalization and efficiency by targeting accessible vision encoders. For instance, MIX.Attack [32] misleads LVLMs by aligning with a mix of irrelevant textual concepts in the CLIP embedding space. VT-Attack [35] leverages text information to generate more effective adversarial samples. VEAttack [26] attacks the patch tokens by minimizing their cosine similarity, achieving high efficiency. This work follows the gray-box setting and further explores the potential of the vision encoder in achieving efficient and generalized LVLM attacks.

3. Methodology

To achieve an efficient and generalizable attack on the vision encoder of LVLMs, we introduce PA-Attack, a two-stage optimization framework. As shown in Figure 3, we first employ a prototype-anchored objective that drives adversarial features toward a dissimilar prototype, encouraging attacks to cover diverse visual attributes and providing a stable and general attack direction. We further concentrate the limited attacking budget on important tokens by weighting tokens with attention and refreshing these weights with a two-stage framework, where the evolving attention during the adversarial optimization is tracked.

3.1. Vision Encoder Attack

Let g denote a pre-trained Large Vision-Language Model (LVLM), and let f be its vision encoder. The loss function $\mathcal{L}$ acts as the core attack objective to guide the generation of adversarial samples. Given an evaluated image-text dataset $\mathbb{D}_{eval} \subset \mathcal{X} \times \mathcal{T}$, which contains pairs $(\boldsymbol{x}_i, t_i)$, where $\boldsymbol{x}_i$ is an clean image, and t_i is the constructions for LVLMs. The vision encoder $f : \mathcal{X} \mapsto \mathbb{R}^{N\times d}$ encodes the images to visual tokens $\boldsymbol{v} \in \mathbb{R}^{N\times d}$, which are aligned with text tokens from $\mathcal{T}$. All the tokens are then fed into the LLM to generate the response in an autoregressive manner. During adversarial attacking, let $\mathcal{B}_\epsilon(\boldsymbol{x}) = \{\boldsymbol{x}' : \|\boldsymbol{x}' - \boldsymbol{x}\|_\infty \leq \epsilon\}$ denote as an ℓ_∞-norm ball for adversarial samples $\boldsymbol{x}'$, where ϵ is a perturbation budget. For t_q in multiple constructions with a single image $\boldsymbol{x}$, the normal optimization problem of white-box can be formulated as

$$\max_{\boldsymbol{x}+\boldsymbol{\delta}_q \in \mathcal{B}_\epsilon(\boldsymbol{x})} \mathcal{L}\left(g(\boldsymbol{x}, t_q), g(\boldsymbol{x}+\boldsymbol{\delta}, t_q)\right), \tag{1}$$

where the optimized $\boldsymbol{\delta}_q$ denotes an adversarial perturbation for construction t_q. While white-box attacks could disrupt LVLMs successfully, they still struggle with poor task generalization, such as the APGD attack [9] shown in Fig. 2 (a). Existing black-box attacks [16, 18, 42] explore the generalized attacks, but primarily focus on expensive transfer strategies and rely on large semantical perturbations. Addressing these gaps, we focus on a gray-box setting, targeting only the shared and fundamental vision encoder in LVLMs. As the goal of our work is solely attacking the vision encoder f to disrupt the performance of LVLMs, the optimization problem can be reformulated from Eq. (1) as

$$\max_{\boldsymbol{x}+\boldsymbol{\delta} \in \mathcal{B}_\epsilon(\boldsymbol{x})} \mathcal{L}\left(f(\boldsymbol{x}), f(\boldsymbol{x}+\boldsymbol{\delta})\right), \tag{2}$$

Following the core optimization goal defined in Eq. (2), we use dissimilarity between the visual features derived from clean images and those perturbed by adversarial perturbations, which can be expressed as

$$\mathcal{L}_{vision} = -\frac{1}{N}\sum\nolimits_j \cos\left(f(\boldsymbol{x})_j, f(\boldsymbol{x}+\boldsymbol{\delta})_j\right), \tag{3}$$

where $j \in \{1, 2, \ldots, N\}$ is the index of tokens.

3.2. Prototype-anchored Guidance

Our goal is to fully explore the potential of attacking only the vision encoder f, which enhances the attack effectiveness and generalization in a gray-box setting. Prior objectives in vision encoder attack (*e.g.*, Eq. (3) and AttackVLM-ii [42]) primarily maximize the feature discrepancy from clean samples $\boldsymbol{x}$. However, without a proper general guidance, the optimization tends to overfit to a few specific tokens as shown in Fig. 2 (d), which concentrate on specific attributes and hinder the generalization across diverse tasks in Fig. 4. To address this, we introduce prototype-anchored guidance to the loss function, leading generated adversarial examples to cover diverse visual attributes.

Given a guidance dataset $\mathbb{D}_{guide} \subset \mathcal{X}$, which does not overlap with the evaluation set as $\pi_1(\mathbb{D}_{eval}) \cap \mathbb{D}_{guide} = \varnothing$, where $\pi_1\left(\mathbb{D}_{eval}\right) = \{\boldsymbol{x} \in \mathcal{X} \mid \exists t \in \mathcal{T}, (\boldsymbol{x}, t) \in \mathbb{D}_{eval}\}$ is the set of all images in $\mathbb{D}_{eval}$. For all images in guidance set $\{\boldsymbol{x}^1_{guide}, \boldsymbol{x}^2_{guide}, \cdots, \boldsymbol{x}^m_{guide}\}$, we extract their features using the vision encoder and save them into a feature memory $\mathcal{M} = \{\boldsymbol{v}^1_{guide}, \boldsymbol{v}^2_{guide}, \cdots, \boldsymbol{v}^m_{guide}\}$ as shown in Fig. 3, where m is the number of images in guidance dataset $\mathbb{D}_{guide}$. As the visual space $\mathbb{R}^{N\times d}$ is typically large, *e.g.*, 256×1024 in CLIP-L/14 with 224×224 inputs, the

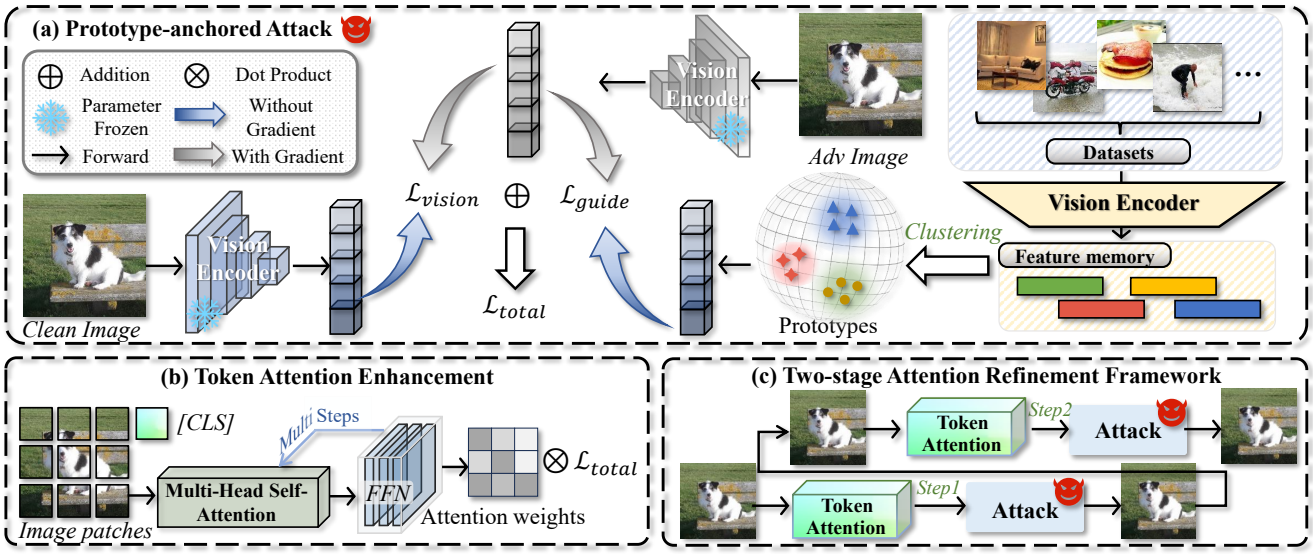


Figure 3. **Overview of PA-Attack.** (a) The prototype-anchored guidance includes a vision encoder attack loss and a guidance loss for general degradation across diverse tasks. (b) Attention obtained by averaging the attention of the class token to each patch in Self-Attention across different Heads. (c) The attention weights are adjusted to align with the adversarial image through a two-stage process.

features will retain the top-m principal components using Principal Component Analysis [1] as $\text{PCA}(\mathcal{M}) \mapsto \mathbb{R}^w$. Subsequently, the K-Means clustering [15] is applied to $\text{PCA}(\mathcal{M})$ to group similar spatial token features into K disjoint clusters. A connection between feature memory $\mathcal{M}$ and K clusters can be established by mapping each feature to a specific cluster. For all m features in the memory, their corresponding cluster indices can be formulated as $\mathbf{I} = [i_1, i_2, \ldots, i_m] \in \mathbb{R}^m$, where $i_t \in \{1, 2, \ldots, K\}$ denotes the cluster index of a specific feature $\boldsymbol{v}^t_{guide}$. Then the features that have the same cluster index can be aggregated to form a prototype that contains diverse visual attributes as

$$\mathbf{p}^k = \frac{1}{|\mathcal{S}_k|} \sum_{t \in \mathcal{S}_k} \boldsymbol{v}^t_{guide}, \text{where } \mathcal{S}_k = \{t \mid i_t = k\}. \quad (4)$$

With Eq. (4), K different prototypes can be constructed as $\mathcal{P} = \{\mathbf{p}^1, \mathbf{p}^2, \cdots, \mathbf{p}^K\}$. In Fig. 2 (b), we demonstrate that different guidance samples have different effects on vision encoder attacks, where more distant features could enhance the attack effect to varying degrees. Among them, the furthest prototype achieves the best attack effect by generating more general guidance. Formally, given an input sample $\boldsymbol{x}$ with its feature $\boldsymbol{v} = f(\boldsymbol{x})$, we compute the cosine similarity between $\boldsymbol{v}$ and each $\mathbf{p}^k$ in $\mathcal{P}$ to measure the distance between prototypes and feature $\boldsymbol{v}$. The prototype with the minimum cosine similarity is selected as

$$k^* = \arg \min_{k \in \{1,2,\ldots,K\}} \cos\left(\boldsymbol{v}, \mathbf{p}^k\right), \quad (5)$$

where k^* denotes the index of the prototype selected for attacking $\boldsymbol{v}$. To guide the attack toward maximizing the dissimilarity, we define the prototype-anchored guidance loss as the cosine similarity between $\boldsymbol{v}'$ and prototype $\mathbf{p}^{k^*}$ as

$$\mathcal{L}_{guide} = \frac{1}{N} \sum\nolimits_j \cos\left(f(\boldsymbol{x})_j, \mathbf{p}^{k^*}_j\right). \quad (6)$$

Here, the subscript j in $\mathbf{p}^{k^*}_j$ serves to index and select the feature corresponding to the j-th token within the prototype $\mathbf{p}^{k^*}$. Through integrating the vision encoder attack loss in Eq. (3) and prototype-anchored guidance loss in Eq. (6), the loss function can then be reformulated as

$$\mathcal{L}_{total} = \frac{1}{N} \sum\nolimits_j [-\cos(\boldsymbol{v}_j, \boldsymbol{v'}_j) + \lambda \cdot \cos(\boldsymbol{v'}_j, \mathbf{p}^{k^*}_j)], \quad (7)$$

where $\boldsymbol{v} = f(\boldsymbol{x})$, $\boldsymbol{v}' = f(\boldsymbol{x} + \boldsymbol{\delta})$ are visual features of clean and adversarial images, and λ is a hyperparameter that balances the loss terms.

3.3. Token Attention Enhancement

While the prototype-anchored guidance provides a more reliable optimization path, it still suffers from the high dimensionality and structural redundancy of visual features $\boldsymbol{v} \in \mathbb{R}^{N \times d}$, which makes the perturbation optimization difficult and inefficient. As shown in Fig. 2 (c), when evaluating token mask proportion versus LVLM performance, the model retains substantial functionality even at a 50% mask proportion, indicating that most tokens are redundant and each token makes an unequal contribution to downstream tasks. Since the maximum budget is limited, we need to focus on more important features during the attack rather than wasting effort on less critical token features to guarantee attack generalization. Observing that preserving higher-

attention tokens (via Attention Mask) maintains better performance than random preservation, we use attention scores as weights to prioritize critical tokens in perturbation.

A vision encoder f splits the input image into n patches and concatenates a class token to form the initial token sequence. This token sequence is subsequently processed through a stack of L layers, each consisting of a multi-head self-attention (MHSA) module and a feed-forward network (FFN). For the MHSA of l-th layer and i-th head, the corresponding key and features could be $\boldsymbol{K}_i^l, \boldsymbol{V}_i^l \in \mathbb{R}^{(N+1)\times d}$. The class token feature $\boldsymbol{y}_i^l$ is calculated by

$$\boldsymbol{y}_i^l = \boldsymbol{a}_i^l \boldsymbol{V}_i^l = \text{softmax}\left(\frac{\boldsymbol{q}_i^l \boldsymbol{K}_i^{lT}}{\sqrt{d}}\right)\boldsymbol{V}_i^l, \tag{8}$$

where $\boldsymbol{q}_i^l$ is the query feature of the class token. As the class token aggregates global image information, its attention to patch tokens serves as a reliable indicator of patch contribution [19]. To obtain a single attention weight for each patch token, we average the attention scores between the class token and each patch token across all h heads as $\boldsymbol{a}^l = \frac{1}{H}\sum_i \boldsymbol{a}_i^l$. So the final weights can be denoted as

$$\boldsymbol{w}_j = \text{softmax}(\boldsymbol{a}^l) = \frac{e^{\boldsymbol{a}_j^l/T}}{\sum_{j=1}^n e^{\boldsymbol{a}_j^l/T}}, \tag{9}$$

where T is a temperature parameter for softmax, and we show its effects in Fig. 2 (c). We choose the middle layer as layer l for the computation. Finally, these attention weights $\boldsymbol{w} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_N\}$ are incorporated into the optimization objective, and the loss can be formulated as

$$\mathcal{L} = -\frac{1}{N}\sum\nolimits_j \boldsymbol{w}_j \cdot [-\cos(\boldsymbol{v}_j, \boldsymbol{v'}_j) + \lambda \cdot \cos(\boldsymbol{v'}_j, \mathbf{p}_j^{k^*})]. \tag{10}$$

During the attack process, the attention scores $\boldsymbol{a}^l$ exhibit dynamic behavior as illustrated in Fig. 2 (d), where the attention map of the adversarial image $\boldsymbol{a}^l(\boldsymbol{x'})$ (green distribution) shows a shift compared to that of the clean image $\boldsymbol{a}^l(\boldsymbol{x})$ (gray distribution). This shift guides the model to gradually focus on less robust features, such as background elements or distracting objects. To dynamically identify and prioritize the optimization for tokens truly critical to the adversarial process, we design a two-stage attention refinement framework, which adjusts attention weights $\boldsymbol{w}$ based on the evolutionary state of adversarial samples $\boldsymbol{x'}$.

Specifically, we first extract the token weights $\boldsymbol{w}_{s1}$ from the attention $\boldsymbol{a}^l(\boldsymbol{x})$ during the forward propagation of the clean image $\boldsymbol{x}$ through the vision encoder. Then, we take Eq. (10) as the optimization objective $\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}), \boldsymbol{w}_{s1}, \mathbf{p}^{k^*})$ in Eq. (10) to prioritize perturbing and perform S_1 attack steps in the first stage. Therefore, the optimization of the first stage can be formulated as

$$\boldsymbol{x'}_{i+1} \leftarrow \boldsymbol{x'}_i + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x'}_i}\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}_i), \boldsymbol{w}_{s1}, \mathbf{p}^{k^*})), \tag{11}$$

Algorithm 1 PA-Attacking Procedure

Require: clean image $\boldsymbol{x}$, perturbation budget ϵ, stage-one iterations S_1, stage-two iterations S_2, loss function $\mathcal{L}$, prototype $\mathcal{P}$ from guidance dataset $\mathbb{D}_{guide}$, step size α, random initialization range $\boldsymbol{\eta} \in [0, \epsilon]$.

$\boldsymbol{x'}_0 \leftarrow \boldsymbol{x} + \text{Uniform}(-\boldsymbol{\eta}, \boldsymbol{\eta})$ ▷ Random start
$k^* = \arg\min_k \cos\left(\boldsymbol{v}, \mathbf{p}^k\right), \mathbf{p}^k \in \mathcal{P}$ ▷ Index in Eq. (5)
$\boldsymbol{w}_{s1} = \text{softmax}(\boldsymbol{a}^l \leftarrow f(\boldsymbol{x}))$ ▷ Token weights Eq. (9)
for $i = 0$ to $S_1 - 1$ **do**
  Compute $\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}_i), \boldsymbol{w}_{s1}, \mathbf{p}^{k^*})$ in Eq. (10)
  $\boldsymbol{x'}_{i+1} \leftarrow \boldsymbol{x'}_i + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x'}_i}\mathcal{L})$ ▷ Eq. (11)
  $\boldsymbol{x'}_{i+1} \leftarrow \text{clip}(\boldsymbol{x'}_{i+1}, \boldsymbol{x} - \epsilon, \boldsymbol{x} + \epsilon)$ ▷ To ℓ_∞-ball
end for
$\boldsymbol{w}_{s2} = \text{softmax}(\boldsymbol{a}^l \leftarrow f(\boldsymbol{x'}_{S_1-1}))$ ▷ Token weights
for $r = S_1$ to $S_1 + S_2 - 1$ **do**
  Compute $\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}_r), \boldsymbol{w}_{s2}, \mathbf{p}^{k^*})$ in Eq. (10)
  $\boldsymbol{x'}_{r+1} \leftarrow \boldsymbol{x'}_r + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x'}_r}\mathcal{L})$ ▷ Eq. (12)
  $\boldsymbol{x'}_{r+1} \leftarrow \text{clip}(\boldsymbol{x'}_{r+1}, \boldsymbol{x} - \epsilon, \boldsymbol{x} + \epsilon)$ ▷ To ℓ_∞-ball
end for
return $\boldsymbol{x}_{\text{adv}}$; ▷ $\boldsymbol{x'}_{S_1+S_2-1} \rightarrow \boldsymbol{x}_{\text{adv}}$

where i is from 0 to $S_1 - 1$, serving as each step in the first-stage attack, and α is the step size. After generating the adversarial image $\boldsymbol{x'}_{s1-1}$, the second stage focuses on refining the attention weights to the adversarial process. We re-input $\boldsymbol{x'}_{s1-1}$ into the vision encoder and recalculate the attention weights $\boldsymbol{w}_{s2}$ in the second stage using the attention $\boldsymbol{a}^l(\boldsymbol{x'}_{s1-1})$ and Eq. (9). Then we reuse Eq. (10) as the objective function $\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}), \boldsymbol{w}_{s2}, \mathbf{p}^{k^*})$ in Eq. (10) to perform another S_2 attack steps, which can be denoted as

$$\boldsymbol{x'}_{r+1} \leftarrow \boldsymbol{x'}_r + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x'}_r}\mathcal{L}(f(\boldsymbol{x}), f(\boldsymbol{x'}_r), \boldsymbol{w}_{s2}, \mathbf{p}^{k^*})). \tag{12}$$

The attack step r is from S_1 to $S_1 + S_2 - 1$, and final adversarial sample in the second stage could be $\boldsymbol{x'}_{S1+S2-1}$. The attacking process is concluded in Alg. 1. And more attention results are presented in the supplementary materials.

4. Experiments

4.1. Experimental settings

Datasets. To demonstrate the generalization of our proposed method across diverse downstream tasks of LVLMs, we select three representative tasks, which are image captioning, visual question answering (VQA), and hallucination detection. For the image captioning task, we adopt COCO [20] and Flickr30k [27] datasets for evaluation. For the VQA task, we choose two extensively utilized datasets, namely TextVQA [30] and VQAv2 [14], to evaluate the attack performance. Finally, for the hallucination task, we construct our evaluation by selecting images from the COCO [20] dataset and leveraging the annotation labels

Table 1. Comparison of PA-Attack with different gray-box attacks across different LVLMs and tasks. For each dataset, we present the performance (↓) after the gray-box attack, and finally, we calculate the average score reduction rate (SRR) (↑) across all datasets.

Model	Attack Perturbation (ϵ)	COCO ↓ $2/255$	COCO ↓ $4/255$	Flickr30k ↓ $2/255$	Flickr30k ↓ $4/255$	TextVQA ↓ $2/255$	TextVQA ↓ $4/255$	VQAv2 ↓ $2/255$	VQAv2 ↓ $4/255$	POPE ↓ $2/255$	POPE ↓ $4/255$	Average SRR ↑ $2/255$	Average SRR ↑ $4/255$
LLaVa1.5-7B	Clean	115.5		77.5		37.1		74.5		84.5		0.0	
	MIX.Attack [32]	67.5	55.4	42.1	35.9	24.6	19.1	59.4	57.9	72.0	69.1	31.2	38.9
	VT-Attack [35]	66.8	29.5	40.0	19.9	26.9	16.1	60.1	**28.3**	67.1	58.5	31.6	59.6
	AttackVLM-ii [42]	41.3	26.3	30.2	20.3	19.7	14.0	55.1	50.4	69.0	61.6	43.3	54.5
	VEAttack [26]	10.8	7.1	10.7	6.3	13.8	10.1	42.9	38.4	47.5	42.8	65.2	71.2
	PA-Attack (ours)	**6.1**	**4.1**	**4.7**	**3.3**	**8.3**	**5.1**	**33.9**	32.5	**29.6**	**33.6**	**77.1**	**79.0**
OF-9B	Clean	79.7		60.1		23.8		48.5		65.7		0.0	
	MIX.Attack [32]	45.9	25.4	33.7	18.0	13.4	8.8	39.8	36.0	59.0	53.3	31.6	49.2
	VT-Attack [35]	49.7	31.6	38.6	23.0	16.1	13.2	39.7	39.5	65.1	62.7	25.0	37.9
	AttackVLM-ii [42]	27.7	10.1	20.1	9.3	10.0	7.4	37.8	33.3	53.3	48.1	46.1	59.8
	VEAttack [26]	7.5	3.7	8.7	**3.2**	12.5	5.7	34.0	32.8	60.6	59.6	52.3	61.5
	PA-Attack (ours)	**4.7**	**3.6**	**5.4**	3.7	**6.6**	**4.8**	**33.0**	**32.6**	**47.6**	**45.9**	**63.4**	**66.4**
LLaVa1.5-13B	Clean	119.2		77.1		39.0		75.6		84.1		0.0	
	MIX.Attack [32]	73.8	60.1	41.0	32.2	22.8	19.7	59.8	58.0	74.2	68.2	31.8	39.9
	VT-Attack [35]	68.6	34.0	40.7	25.2	28.7	18.4	59.8	**27.6**	70.9	67.9	30.5	54.9
	AttackVLM-ii [42]	46.6	25.3	27.5	17.9	19.6	14.7	55.0	49.9	63.8	57.8	45.3	56.6
	VEAttack [26]	11.2	6.5	9.1	5.7	12.4	8.6	41.5	37.6	47.6	44.7	67.1	72.4
	PA-Attack (ours)	**5.4**	**3.9**	**4.1**	**2.0**	**7.0**	**6.5**	**36.5**	34.4	**31.4**	**27.9**	**77.3**	**79.8**

provided by the POPE [17]. Following the setting in [26], we randomly sampled 500 samples in each task for conducting adversarial attacks and evaluating.

Implementation Settings. In prototype-anchored guidance, we randomly select the guidance images from the COCO dataset. The number of images in the guidance dataset is set to $m = 3000$, and we use Principal Component Analysis (PCA) to retain the top-$w = 1024$ principal components of the visual tokens. Subsequently, we apply K-Means clustering to get $K = 20$ disjoint prototypes from the guidance of PCA features. The balance parameter of two losses is $\lambda = 1.0$. In the attention enhancement, we choose the attention from the middle layer while the temperature for softmax is set to $T = 1/20$. The number of attack iterations is set to $S_1 = 50$ in the first stage, and $S_2 = 100$ in the second stage. The methods AttackVLM-ii [42] and VEAttack [26] that we use for comparison set the iterations as $S = 100$, and VT-Attack [35] set as $S = 150$. Following the setting in VEAttack [26], we set the perturbation budges as $\epsilon = 2/255$ and $\epsilon = 4/255$ to ensure the imperceptible of attacks, and the attack step size is set to $\alpha = 1/255$. We evaluate our effectiveness across three LVLMs namely LLaVa1.5-7B [23], LLaVa1.5-13B [23] and OpenFlamingo-9B (OF-9B) [3]. All experiments were conducted on a NVIDIA-A6000 GPU.

Evaluation metrics. To clearly demonstrate the performance changes of each task, we first show the performance of different tasks after the attack. For image captioning tasks, we use the CIDEr [33] score, while for VQA tasks, we measure them by VQA accuracy [2]. Following POPE [17], the hallucination benchmark is evaluated by F1-score. As a gray-box attack, we aim to disrupt the performance of LVLMs more generally and effectively. Therefore, we use the score reduction rate (SRR) to represent the effectiveness of the attack, which can be calculated as:

$$\text{SRR} = 1 - (\text{Score}_{adv}/\text{Score}_{clean}) \tag{13}$$

where Score_{clean} and Score_{adv} denote the performance before and after the attack. The overall performance across tasks and the ablations is expressed using SRR.

4.2. Comparison results

Table 1 shows the comparison of our proposed PA-Attack with different gray-box attacks, including MIX.Attack, VT-Attack, AttackVLM-ii, and VEAttack, where AttackVLM-ii follows the black-box attack paradigm targeting vision encoders, treating the accessible vision encoder as a naive baseline. To visually demonstrate performance, we report the corresponding benchmark metrics for each single dataset. The results show large absolute drops across models, such as 115.5 to 4.1 in the COCO dataset and LLaVa1.5-7B and 60.1 to 3.7 in the Flickr30k dataset and OpenFlamingo-9B. For task generalization, we summarize effectiveness with the score reduction rate (SRR), and our PA-Attack delivers the best average SRR on every model across all tasks. With perturbation budgets as $2/255$ and $4/255$, PA-Attack reaches 77.1% and 79% on LLaVA-1.5-7B, 63.4% and 66.4% on OF-9, which surpasses the strongest gray-box VEAttack by 11.1% and 6.7% in average, and also outperforms the black-box naive baseline AttackVLM-ii by 27.7% and 18.1% in average. Although VT-Attack shows occasional advantages on VQAv2 and

Table 2. Ablation study on the components of PA-Attack with SRR as the metric, where PG denotes the Prototype-anchored Guidance, AE is the Attention Enhancement, and TS is a two-stage attention refinement framework.

PG	AE	TS	Step	COCO	Flickr30k	TextVQA	VQAv2
			100	93.8	91.9	72.8	48.4
✓			100	95.5	94.8	76.3	52.2
	✓		100	93.7	92.6	72.3	50.9
✓	✓		100	95.3	94.7	76.6	54.2
			150	94.8	93.7	77.9	48.8
✓	✓		150	96.2	95.5	81.1	54.4
✓	✓	✓	150	**96.5**	**95.7**	**86.3**	**56.4**

LLaVA models, its performance degrades markedly under a small perturbation budget $\epsilon = 2/255$ and the advantage does not generalize across tasks, whereas our method achieves a consistently more general and effective attack. Finally, PA-Attack remains highly effective under small perturbations, since even at $\epsilon = 2/255$ it already drives all captioning metrics to low single digits and surpasses all models and tasks.

4.3. Ablation study

Ablation of each component in PA-Attack. In Table 2, we show the contribution of the prototype-anchored guidance (PG), attention enhancement (AE), and the two-stage refinement (TS) using SRR on LLaVa1.5-7B and four datasets with $\epsilon = 4/255$. Starting from the 100-step baseline, introducing PG yields consistent gains, lifting SRR from 93.8% to 95.5% on COCO, and from 72.8% to 76.3% on TextVQA. AE alone produces mixed changes at 100 steps, improving Flickr30k and VQAv2 while leaving COCO and TextVQA essentially unchanged. Combining PG and AE at 100 steps delivers a clear improvement in SRR over the baseline across all datasets, with the largest increase on VQAv2, where SRR rises from 48.4% to 54.2%, which proves the effectiveness of the two modules we designed. To ensure a fair comparison of the two-stage framework, we also set the baseline to 150 iterations. Increasing the optimization budget to 150 steps improves the baseline, but algorithmic components remain the dominant factor. With PG and AE at 150 steps, SRR further reaches 96.2% on COCO, 81.1% on TextVQA, which are higher than the 150-step baseline. Introducing the two-stage refinement produces the best results, reaching 96.5% on COCO and 86.3% on TextVQA, which corresponds to improvements over the 150-step baseline and shows the effectiveness of our attention refinement module.

Ablation of loss balance weight. In Table 3, we study the balance weight λ between the vision encoder attack loss and the prototype-directed term. It can be seen that moderate weight with $\lambda = 1.0$ achieves the best overall SRR. A smaller weight $\lambda = 0.5$ weakens prototype guidance, which could hurt the QA datasets that have a variety of instructions and may require a more general perturbation. Increasing λ to 2.0 overemphasizes the guidance term, which may affect the degree of damage caused by the vision encoder attack and causes small drops on each dataset. Thus, we adopt the weight $\lambda = 1.0$ for all experiments.

Table 3. Ablation study on the weight λ of the Prototype-anchored Guidance loss with SRR as the metric.

λ	COCO	Flickr30k	TextVQA	VQAv2	Average
0.5	**96.6**	95.1	83.8	51.3	81.7
1.0	96.5	**95.7**	**86.3**	**56.4**	**83.7**
2.0	95.5	94.7	84.1	55.8	82.5

Table 4. Impact of prototype construction parameters, where m is the number of images in the guidance image dataset, w is the dimension after PCA, and K is the number of prototypes.

m	w	K	Flickr30k	TextVQA	VQAv2	Average
baseline			91.9	72.8	48.4	71.0
1000	512	20	92.8	77.1	47.6	72.5
2000	1024	20	92.6	**79.8**	47.5	73.3
3000	1024	20	**94.8**	76.3	**52.2**	**74.4**
3000	1024	10	94.2	77.6	51.2	74.3
3000	1024	30	92.3	79.5	49.0	73.6
3000	2048	20	89.9	74.7	47.5	70.7

Ablation of the parameters in prototype construction. Table 4 presents the impact of the key parameters for prototype construction. We first observe that all parameter variations outperform the baseline, demonstrating the general effectiveness of prototype-anchored guidance. Our method achieves its peak average performance of 74.4% with $m = 3000$. Fixing this optimal m, we analyze the PCA dimension w. Setting $w = 1024$ proves optimal, while increasing the dimension to 2048 leads to a significant performance degradation, resulting in a score of 70.7%. This indicates that a more compact dimension with $w = 1024$ effectively filters out irrelevant variance. Finally, varying the number of prototypes K reveals that $K = 20$ yields the highest score, slightly outperforming both fewer clusters $K = 10$ and more clusters $K = 30$, suggesting $K = 20$ provides the best granularity. Therefore, we adopt $m = 3000$, $w = 1024$, and $K = 20$ as our default configuration.

Ablation of the number of stages. We analyze the impact of the number of refinement stages in Fig. 5 (a). The score reduction rate steadily improves as the number of stages increases, which validates that refining attention allows the attack to progressively focus on more critical features of the adversarial process. However, there is a trade-off between performance and efficiency. Notably, the most significant performance leap is observed when transitioning from one to two stages. In contrast, the gains from subsequent stages become marginal. Therefore, we adopt two stages as our

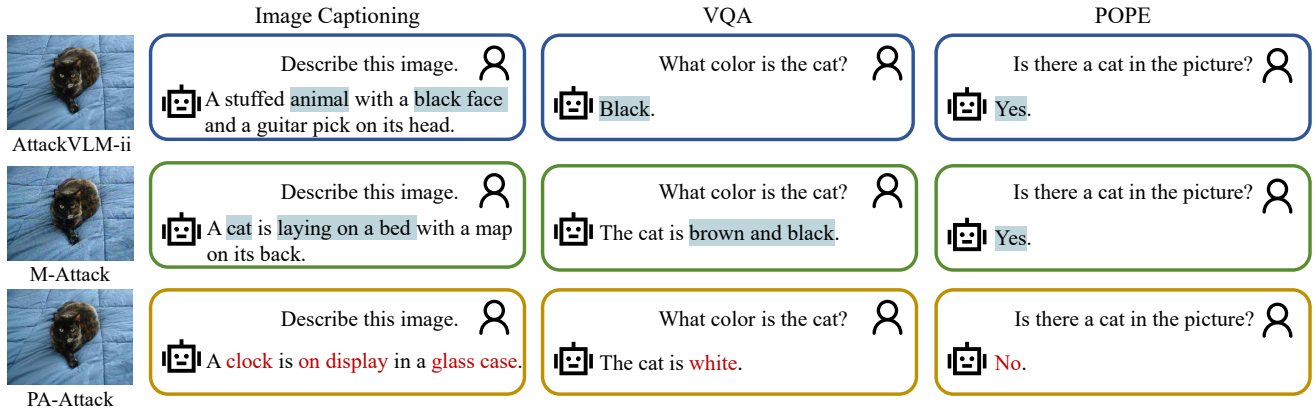
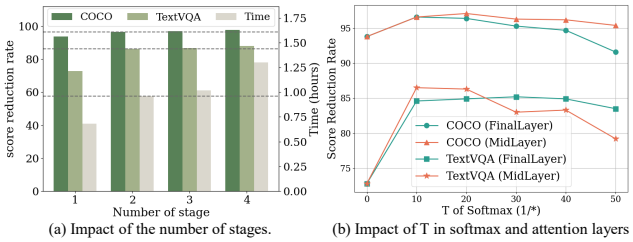


Figure 4. **Comparison of the responses of LLaVa1.5-7B with different attacks.** The attributes with a blue background remain unchanged, while the red texts indicate that the attributes have changed.



(a) Impact of the number of stages. (b) Impact of T in softmax and attention layers

Figure 5. (a) **Ablation of the number of stages in adaptively attention refinement.** Slash columns compare the score reduction rate, while dotted columns represent the attack time. (b) **Ablation of T in softmax and layers l in attention enhancement.**

default configuration to achieve an optimal balance between attack effectiveness and efficiency.

Ablation of the T in softmax and layers in attention. As shown in Fig. 5 (b), we analyze the impact of the source layer for attention extraction and the softmax temperature T in Eq. (9), which is plotted as $1/T$. The performance on both COCO and TextVQA peaks when T is set around $1/20$. This indicates that a large T flattens the distribution, failing to prioritize critical tokens and thus yielding lower scores. Conversely, a small T creates a sharp distribution, which also degrades performance, focusing too narrowly and ignoring other relevant features. Furthermore, at $T = 1/20$, using the MidLayer outperforms the FinalLayer. We hypothesize this is because mid-level features retain richer object information, which is beneficial for guiding the attack. Therefore, we adopt the MidLayer for attention extraction and set $T = 1/20$ as our default.

Ablation of different guidance samples. We validate the effectiveness of prototype-anchored guidance by comparing it against several guidance samples in Table 5. It can be seen that naively selecting the Nearest sample (the highest cosine similarity) from the guidance set $\mathcal{M}$ will cause the score to significantly decrease. This confirms that optimizing toward a highly similar feature provides a conflicting adversarial signal. Using a Farthest sample or Mean sample yields better results, while the Nearest also shows effective, demonstrating the benefit of using a generalized sample as a stable guidance. Crucially, selecting the Farthest prototype as defined in Eq. (5) achieves the highest average score of 82.9%, indicating that the guidance is both semantically general and maximally divergent, providing the strongest and most stable optimization for the adversarial process.

Table 5. Impact of different samples for the guidance objective.

Guidance	COCO	Flickr30k	VQAv2	Average
baseline	93.8	91.9	48.4	78.1
Mean sample	95.1	94.1	56.2	81.8
Nearest sample	64.6	86.2	33.4	61.4
Farthest sample	96.3	95.2	50.2	80.6
Nearest prototype	95.2	**96.1**	53.4	81.6
Farthest prototype	**96.5**	95.7	**56.4**	**82.9**

5. Conclusion

In this paper, we propose **PA-Attack**, a novel gray-box adversarial attack that effectively targets the shared vision encoders of Large Vision-Language Models. Our method addresses the critical challenges of attack generalization and efficiency. By introducing a prototype-anchored guidance, PA-Attack establishes a stable and general attack direction, preventing overfitting to limited visual attributes. Furthermore, our two-stage attention enhancement mechanism strategically focuses perturbations on the most critical visual tokens and adaptively calibrates to the evolving attention patterns during the attack. Extensive experiments demonstrate that PA-Attack achieves a state-of-the-art 75.1% average score reduction rate, showing superior effectiveness and strong generalization across diverse downstream tasks and LVLM architectures. Our findings underscore a significant vulnerability of sharing vision backbones, highlighting the urgent need to develop more robust defenses for foundational multi-modal systems.

Acknowledgments

This work was supported in part by Young Scientist Fund (No. 62406265) of NSFC, Start-up Grant (No. 9610680) of the City University of Hong Kong, and the Australian Research Council under Projects DP240101848 and FT230100549.

References

[1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010. 4

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 6

[3] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 6

[4] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 1

[5] Rishika Bhagwatkar, Shravan Nayak, Reza Bayat, Alexis Roger, Daniel Z Kaplan, Pouya Bashivan, and Irina Rish. Towards adversarially robust vision-language models: Insights from design choices and prompt formatting techniques. *arXiv preprint arXiv:2407.11121*, 2024. 1

[6] Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024. 2

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

[8] Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph 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)*, 2(3):6, 2023. 2

[9] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020. 1, 2, 3

[10] Xuanming Cui, Alejandro Aparcedo, Young Kyun Jang, and Ser-Nam Lim. On the robustness of large multimodal models against image adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24625–24634, 2024. 1, 2

[11] Yinpeng Dong, Huanran Chen, Jiawei Chen, Zhengwei Fang, Xiao Yang, Yichi Zhang, Yu Tian, Hang Su, and Jun Zhu. How robust is google's bard to adversarial image attacks? *arXiv preprint arXiv:2309.11751*, 2023. 3

[12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 2

[13] Google. Bard. https://bard.google.com/, 2023. 3

[14] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 5

[15] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108, 1979. 4

[16] Xiaojun Jia, Sensen Gao, Simeng Qin, Tianyu Pang, Chao Du, Yihao Huang, Xinfeng Li, Yiming Li, Bo Li, and Yang Liu. Adversarial attacks against closed-source mllms via feature optimal alignment. *arXiv preprint arXiv:2505.21494*, 2025. 1, 3

[17] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. 6

[18] Zhaoyi Li, Xiaohan Zhao, Dong-Dong Wu, Jiacheng Cui, and Zhiqiang Shen. A frustratingly simple yet highly effective attack baseline: Over 90% success rate against the strong black-box models of gpt-4.5/4o/o1. *arXiv preprint arXiv:2503.10635*, 2025. 1, 3

[19] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *arXiv preprint arXiv:2202.07800*, 2022. 5

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 5

[21] Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Xiang Fang, Keke Tang, Yao Wan, and Lichao Sun. Pandora's box: Towards building universal attackers against real-world large vision-language models. *Advances in Neural Information Processing Systems*, 37:52127–52158, 2024. 1

[22] Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Yu Cheng, and Wei Hu. A survey of attacks on large vision–language models: Resources, advances, and future trends. *IEEE Transactions on Neural Networks and Learning Systems*, 2025. 1

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in neural information processing systems*, pages 34892–34916, 2023. 1, 6

[24] Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. Deepseek-vl: towards real-world vision-

language understanding. *arXiv preprint arXiv:2403.05525*, 2024. 1

[25] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International conference on learning representations*, 2018. 1, 2

[26] Hefei Mei, Zirui Wang, Shen You, Minjing Dong, and Chang Xu. Veattack: Downstream-agnostic vision encoder attack against large vision language models. *arXiv preprint arXiv:2505.17440*, 2025. 3, 6

[27] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 5

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 1

[29] Christian Schlarmann and Matthias Hein. On the adversarial robustness of multi-modal foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3677–3685, 2023. 1, 2

[30] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019. 5

[31] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[32] Haoqin Tu, Chenhang Cui, Zijun Wang, Yiyang Zhou, Bingchen Zhao, Junlin Han, Wangchunshu Zhou, Huaxiu Yao, and Cihang Xie. How many unicorns are in this image? a safety evaluation benchmark for vision llms. *arXiv preprint arXiv:2311.16101*, 2023. 3, 6

[33] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 6

[34] Ruofan Wang, Xingjun Ma, Hanxu Zhou, Chuanjun Ji, Guangnan Ye, and Yu-Gang Jiang. White-box multimodal jailbreaks against large vision-language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6920–6928, 2024. 2

[35] Yubo Wang, Chaohu Liu, Yanqiu Qu, Haoyu Cao, Deqiang Jiang, and Linli Xu. Break the visual perception: Adversarial attacks targeting encoded visual tokens of large vision-language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1072–1081, 2024. 2, 3, 6

[36] Mang Ye, Xuankun Rong, Wenke Huang, Bo Du, Nenghai Yu, and Dacheng Tao. A survey of safety on large vision-language models: Attacks, defenses and evaluations. *arXiv preprint arXiv:2502.14881*, 2025. 1

[37] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 13040–13051, 2024. 1

[38] Shen You, Wei Jiang, Hefei Mei, Danei Gong, Zhongshen Li, Jixiang Yu, Junkai Ji, Qiuzhen Lin, Xiangtao Li, and Ka-Chun Wong. Unibreak: A unified evolutionary token-level jailbreaking framework for large language models. *IEEE Transactions on Evolutionary Computation*, 2026. 1

[39] Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Guoyin Wang, Heng Li, Jiangcheng Zhu, Jianqun Chen, et al. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*, 2024. 1, 2

[40] Jiaming Zhang, Junhong Ye, Xingjun Ma, Yige Li, Yunfan Yang, Jitao Sang, and Dit-Yan Yeung. Anyattack: Towards large-scale self-supervised generation of targeted adversarial examples for vision-language models. *arXiv preprint arXiv:2410.05346*, 2024. 1, 3

[41] Yichi Zhang, Yao Huang, Yitong Sun, Chang Liu, Zhe Zhao, Zhengwei Fang, Yifan Wang, Huanran Chen, Xiao Yang, Xingxing Wei, et al. Multitrust: A comprehensive benchmark towards trustworthy multimodal large language models. *Advances in Neural Information Processing Systems*, 37:49279–49383, 2024. 1

[42] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Chongxuan Li, Ngai-Man Man Cheung, and Min Lin. On evaluating adversarial robustness of large vision-language models. *Advances in Neural Information Processing Systems*, 36:54111–54138, 2023. 1, 2, 3, 6

[43] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1