Transferable Targeted Attacks on Vision–Language Models via Multi-View Feature Optimization

1st Baolin Yan
Institute of Software Chinese Academy of Sciences
Beijing, China
yanbaolin2023@iscas.ac.cn

2nd Xiaotian Ai
Shenyang Aircraft Design and Research Institute
Shenyang, China
240468702@qq.com

3rd Yuxi Ma
Institute of Software Chinese Academy of Sciences
Beijing, China
mayuxi@iscas.ac.cn

4th Lingzhong Meng
Institute of Software Chinese Academy of Sciences
Beijing, China
lingzhong@iscas.ac.cn

5th Youdi Gong
Institute of Software Chinese Academy of Sciences
Beijing, China
gongyoudi@iscas.ac.cn

6th Guang Yang*
Institute of Software Chinese Academy of Sciences
Beijing, China
yangguang@iscas.ac.cn

***Abstract*—Vision–Language Models (VLMs) have achieved substantial progress across tasks such as image captioning, visual question answering, and multimodal reasoning, and are increasingly deployed in real-world applications. Despite these advances, recent studies show that VLMs remain highly susceptible to transferable adversarial examples even in black-box settings. Existing transfer-based attack methods typically optimize adversarial perturbations under a single or limited number of views, which tends to cause overfitting to surrogate models and consequently restricts cross-model transferability. To address these limitations, we propose MVM-Attack, a transferable targeted attack framework that performs multi-view feature optimization in the visual embedding space of VLMs. Our method leverages diverse geometric view transformations to enforce consistent adversarial representations, thereby enhancing robustness against feature variations. In addition, we introduce a memory-guided ensemble weighting strategy that adaptively emphasizes shared vulnerable regions across surrogate models while stabilizing optimization using historical gradients. Extensive experiments on both open-source and closed-source VLMs show that, under the same perturbation constraints, MVM-Attack achieves substantially higher targeted attack success rates compared to multiple strong baselines.**

***Index Terms*—Vision language models, adversarial attacks, targeted attacks, transferability, surrogate models**

I. Introduction

Vision–Language Models (VLMs) have become a core component of modern multimodal systems. By aligning visual and textual representations into a shared embedding space, these models [1], [2] enable powerful cross-modal understanding capabilities. Through large-scale vision–language pretraining [3], [4], [5], VLMs have demonstrated outstanding performance across a wide range of tasks, including image captioning [6], visual question answering [7], and multimodal reasoning [8]. Beyond numerous open-source models, closed-source commercial systems [9], [10], built upon VLMs have also been widely deployed in real-world applications, further highlighting the practical impact of VLMs.

Despite their remarkable success, recent studies [11], [12], [13] have shown that VLMs remain vulnerable to adversarial examples [14], inheriting the adversarial fragility of their underlying visual encoders. In particular, targeted adversarial attacks in black-box settings pose a serious security threat, where attackers aim to induce specific semantic outputs without access to the internal architecture of the target model. Due to the inaccessibility of closed-source models, the effectiveness of such attacks largely depends on the transferability of adversarial examples—namely, whether adversarial samples generated on surrogate models can maintain their attack effectiveness across different model architectures and training distributions. Enhancing the transferability of targeted adversarial attacks has become a central yet highly challenging research problem.

Challenges. Existing transfer-based attacks against VLMs typically generate perturbations by aligning adversarial samples with target samples in the embedding space of surrogate models [15]. While such methods achieve a certain level of effectiveness at the global semantic level, their optimization processes are often restricted to a single or limited set of input views. This limitation tends to cause overfitting to surrogate models, resulting in degraded robustness and transferability when facing view variations or unseen models. Furthermore, transfer attacks commonly rely on ensembles of multiple surrogate models to improve robustness. However, existing approaches lack fine-grained gradient selection and guidance mechanisms, making it difficult to focus on vulnerability regions shared across multiple surrogate models, which in turn reduces attack effectiveness. These issues are further amplified in targeted attack scenarios, rendering precise and transferable semantic manipulation particularly challenging.

Proposal. We revisit the problem of transferable targeted adversarial attacks on VLMs from the perspective of multi-

*Corresponding author: Guang Yang.

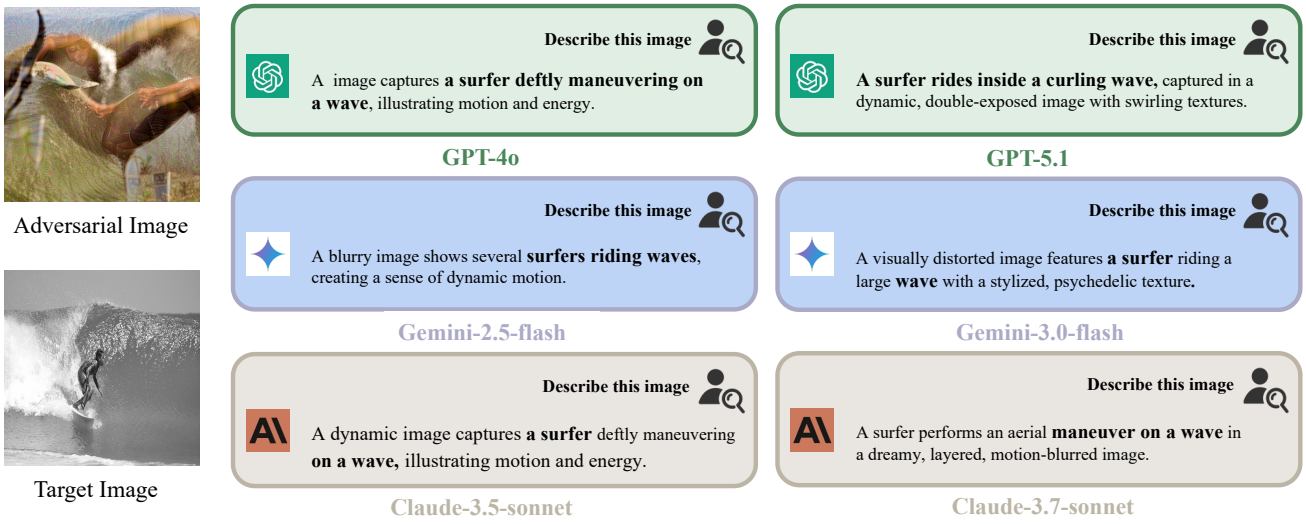

Fig. 1. Examples of MVM-Attack against mainstream closed-source commercial models on the image captioning task, where adversarial images generated by MVM-Attack induce the models to output the semantics of the target image under the prompt "Describe this image."

view feature optimization and propose **MVM-Attack (Multi-View Memory-guided Attack)**. Unlike prior optimization strategies [16], [17] that rely on local feature alignment, we introduce a multi-view constraint by applying diverse block-wise geometric transformations to the adversarial sample during optimization. This encourages the perturbation to induce consistent feature shifts across different views, thereby improving its robustness to view variations and enhancing cross-model transferability. Moreover, we propose a memory-guided ensemble weighting strategy that further enhances the transferability of adversarial examples. As illustrated in Fig. 1, adversarial samples crafted by MVM-Attack successfully transfer to and attack multiple mainstream commercial VLMs.

Overall, our contributions are threefold:

- We propose MVM-Attack, a transferable targeted adversarial attack framework for VLMs. By incorporating multi-view feature optimization with view-invariance constraints, our method effectively mitigates overfitting to surrogate models and substantially improves cross-model transferability.
- A memory-guided ensemble weighting mechanism is proposed to adaptively reweight surrogate model gradients toward shared vulnerability regions, thereby further improving attack transferability.
- Through extensive experiments on a broad spectrum of representative open-source and closed-source VLMs, we show that MVM-Attack consistently outperforms state-of-the-art targeted attack methods under the same perturbation constraints.

II. Related Works

Under black-box settings, adversarial attacks are generally categorized into query-based [18], [19] and transfer-based [20], [21] approaches. Query-based attacks typically require a large number of model queries, leading to prohibitively high costs, whereas transfer-based attacks construct adversarial examples on accessible surrogate models and rely on their cross-model generalization to attack unknown target models. Owing to their stronger practicality and lower cost, transfer-based attacks are more suitable for real-world black-box scenarios.

From a methodological perspective, existing transferable targeted attacks against VLMs can be broadly divided into two categories. The first category comprises **discriminative optimization–based methods**, which directly optimize adversarial perturbations on surrogate models through iterative gradient-based procedures. AttackVLM [22] was among the first to systematically study targeted transfer attacks in VLM scenarios and demonstrated that imposing target constraints in the image feature space is more effective for cross-model transferability. Building upon this insight, SSA-CWA [15] enhances transferability to closed-source models by combining Spectral Simulation Attack (SSA) with Common Weakness Attack (CWA). M-Attack [16] further improves cross-model generalization by incorporating input augmentation strategies such as random cropping and scale resizing. On this basis, FOA-Attack [17] introduces both global and local feature constraints in the visual embedding space and adopts a dynamic multi-model ensemble strategy to further enhance transferability.

The second category consists of **generation-based methods**, which construct adversarial examples by training independent generative models to directly produce adversarial perturbations. AdvDiffVLM [23] employs a diffusion-based framework combined with adaptive ensemble gradient estimation and saliency-guided mechanisms to generate targeted adversarial samples. AnyAttack [24] adopts a self-supervised learning paradigm to train a noise generator on large-scale image–text data, enabling targeted attacks on multimodal mod-

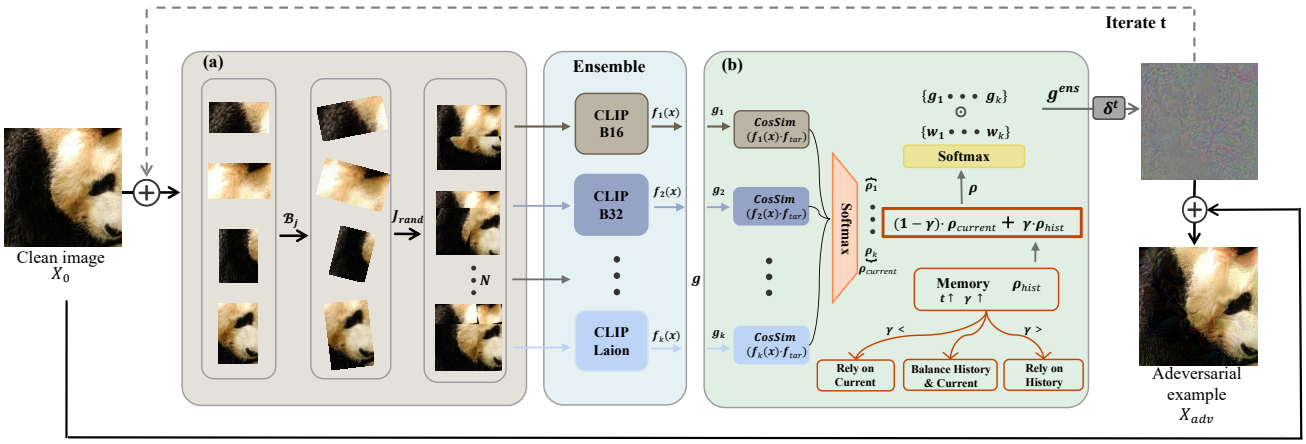

Fig. 2. Overview of the proposed MVM-Attack framework. (a) Multi-view feature optimization: diverse block-wise geometric transformations are applied to generate multiple views, whose features are jointly optimized for view-consistent adversarial manipulation. (b) Memory-guided ensemble weighting: surrogate models are adaptively reweighted using difficulty scores to emphasize shared vulnerable regions during optimization.

els without manual annotations. These approaches typically require substantial additional training, and the generalization capability of the learned generators plays a critical role in determining their transfer effectiveness.

III. Methodology

This section introduces the proposed MVM-Attack framework for transferable targeted attacks on VLMs. As shown in Fig. 2, the framework is composed of two core components: multi-view feature optimization and a memory-guided ensemble weighting strategy.

A. Problem Definition

Given a clean image $x \in \mathbb{R}^{H\times W\times 3}$ and a target image x_t, our goal is to generate an adversarial example $x^{adv} = x + \delta$, such that it induces visual-semantic representations similar to those of x_t on unseen target models, while keeping the perturbation δ imperceptible.

Under the black-box setting, the parameters and gradients of the target model are inaccessible. We therefore assume that the attacker has access to a set of K surrogate visual encoders $\mathcal{F} = \{f_k(\cdot)\}_{k=1}^{K}$, where each surrogate model maps an input image into a visual embedding space. Let $f_k(x) \in \mathbb{R}^d$ denote the normalized visual feature representation extracted by the k-th surrogate model.

Following the common setting of transfer-based adversarial attacks, we formulate a targeted optimization objective in the visual embedding space of VLMs. Specifically, the adversarial perturbation is optimized to minimize the feature distance between the adversarial image and the target image across multiple surrogate models:

$$\min_{\delta} \sum_{k=1}^{K} \mathcal{L}(f_k(x+\delta),\, f_k(x_t)), \quad \text{s.t.} \quad \|\delta\|_\infty \le \epsilon, \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ denotes a similarity-based loss function defined in the visual embedding space, and ϵ constrains the perturbation magnitude under the ℓ_∞ norm.

B. Multi-View Feature Optimization

Optimizing adversarial perturbations under a single input view often leads to overfitting to the specific feature representations of surrogate models, thereby limiting transferability to unseen target models. To address this issue, we propose a multi-view feature optimization mechanism, which jointly optimizes adversarial perturbations across diverse geometric views to explicitly enhance the robustness of feature manipulation under view variations.

Multi-View Transformation Space. Given a clean image x and the perturbation estimate δ at the current iteration, we form the intermediate adversarial input $x^{adv} = x + \delta$. We construct a set of N randomized geometric transformations $\mathcal{T}_i(\cdot)$, each parameterized by transformation parameters θ_i sampled from a predefined distribution:

$$x_i^{adv} = \mathcal{T}_i(x^{adv}) = \mathcal{T}(x^{adv};\theta_i), \quad i = 1,\dots,N. \tag{2}$$

Each transformation includes randomized block-wise scaling, rotation, and edge cropping, allowing x^{adv} to be observed under diverse geometric conditions.

Block-wise Geometric Transformation. To enrich local geometric diversity, the image is first divided into M non-overlapping blocks $\{B_j\}_{j=1}^{M}$. Each block is transformed independently using a set of randomly sampled parameters $\theta_{i,j} = \{\alpha_{i,j}, r_{i,j}, c_{i,j}\}$, where $\alpha_{i,j}$ denotes a random scaling factor, $r_{i,j}$ a random rotation angle, and $c_{i,j}$ parameters for randomized edge cropping.

(1) Random scaling. For each block B_j in view i, the scaling factor $\alpha_{i,j}$ is sampled from a continuous interval:

$$\alpha_{i,j} \sim \text{Uniform}(\alpha_{\min}, \alpha_{\max}), \tag{3}$$

where typical values are $\alpha_{\min} = 1.0$ and $\alpha_{\max} = 1.3$. The block is resized using bilinear interpolation:

$$B_{i,j}^{(\text{sc})} = \text{Resize}(B_j, \alpha_{i,j}). \tag{4}$$

(2) Random rotation. A subset of image blocks is randomly selected and rotated by a small angle to create local orientation variations. The rotation angle is sampled as:

$$r_{i,j} \sim \mathrm{Uniform}(-\theta_{\max}, \theta_{\max}), \tag{5}$$

where $\theta_{\max}$ is typically 20°–30°. Rotation is performed around the block center with zero-padding:

$$B_{i,j}^{(\mathrm{rot})} = \mathrm{Rotate}\left(B_{i,j}^{(\mathrm{sc})}, r_{i,j}\right). \tag{6}$$

(3) Random edge cropping. To simulate viewpoint misalignment, a random subset of image edges is cropped. For each view, we sample the cropping offsets:

$$c_{i,j} = (t_{i,j}, b_{i,j}, l_{i,j}, r_{i,j}), \tag{7}$$

where each offset is drawn from a bounded integer range. The resized–rotated block is then cropped and resized back to its original shape:

$$B_{i,j}^{(\mathrm{geo})} = \mathrm{Resize}\big(B_{i,j}^{(\mathrm{rot})}[t_{i,j} : b_{i,j},\, l_{i,j} : r_{i,j}],\, H_j, W_j\big). \tag{8}$$

View Reconstruction. After all blocks are transformed independently, bilinear interpolation is applied to ensure spatial continuity, and blocks are reassembled to produce the final view:

$$\mathcal{T}_i(x^{adv}) = \bigcup_{j=1}^{M} B_{i,j}^{(\mathrm{geo})}. \tag{9}$$

This block-wise transformation space injects rich local geometric variations while preserving the global structure of the image, enabling the perturbation to be optimized under a wide range of spatial configurations and reducing its overfitting to any specific surrogate model view.

Multi-View Feature Optimization Objective. Let $f(\cdot)$ denote a surrogate model that maps an input image to a normalized visual embedding $f(x) \in \mathbb{R}^d$. Under a single surrogate model, the multi-view feature optimization objective is defined as:

$$\mathcal{L}_{\mathrm{MV}}(\delta) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(f(\mathcal{T}_i(x+\delta)),\, f(x_t)), \quad \text{s.t.} \quad \|\delta\|_\infty \le \epsilon. \tag{10}$$

This objective enforces consistent feature alignment between adversarial and target images across multiple geometric views, thereby encouraging the perturbation to induce stable feature shifts under view variations.

C. Memory-Guided Ensemble Weighting

Ensembling multiple surrogate models for joint optimization is a common strategy to enhance the transferability of adversarial examples. However, different visual encoders often exhibit substantial discrepancies in adversarial vulnerability, which may stem from variations in network architectures, training data, or multimodal fusion mechanisms. As a result, assigning equal importance to all surrogate models during optimization can be suboptimal. To address this issue, we propose a memory-guided ensemble weighting strategy that adaptively balances the contributions of different surrogate models by incorporating both current and historical attack difficulty information.

Model-wise Attack Difficulty Estimation. For each surrogate model f_k, we first estimate the similarity between the adversarial image and the target image under the current multi-view optimization setting:

$$s_k = \frac{1}{N}\sum_{i=1}^{N} \mathrm{sim}(f_k(\mathcal{T}_i(x+\delta)),\, f_k(x_t)), \tag{11}$$

where $\mathrm{sim}(\cdot,\cdot)$ is a similarity function. A higher similarity score indicates better attack performance on the surrogate model. To quantify the attack difficulty for each model, we invert and normalize the similarity scores:

$$\rho_k = 1 - \frac{s_k - \min_j s_j}{\max_j s_j - \min_j s_j + \eta}, \tag{12}$$

where η is a small constant introduced for numerical stability. After normalization, models with lower similarity scores are assigned higher difficulty values.

Memory-Guided Difficulty Aggregation. Although the instantaneous difficulty estimation reflects the current attack state, it may fluctuate significantly across optimization iterations, potentially destabilizing the optimization process. To mitigate this issue, we introduce a memory-guided aggregation mechanism that smooths difficulty estimates over recent iterations.

Let $\rho_k^{(t)}$ denote the instantaneous difficulty score of model k at iteration t. We maintain a difficulty memory buffer and compute a weighted average of historical difficulty estimates:

$$\bar{\rho}_k^{(t)} = \frac{\sum_{l=0}^{L-1} \lambda^l\, \rho_k^{(t-l)}}{\sum_{l=0}^{L-1} \lambda^l}, \tag{13}$$

where $\lambda \in (0,1)$ is a decay factor that controls the contribution of historical information, assigning larger weights to more recent iterations.

To combine the current and historical difficulty estimates, we introduce a scheduling parameter $\gamma_t \in [0,1]$, initially $\gamma_0 = 0$, which gradually increases as optimization proceeds. The final difficulty score for model k at iteration t is computed as:

$$\tilde{\rho}_k^{(t)} = \gamma_t\, \bar{\rho}_k^{(t)} + (1-\gamma_t)\, \rho_k^{(t)}. \tag{14}$$

In the early stages of optimization, the strategy relies more heavily on instantaneous feedback, while in later stages it progressively emphasizes stable historical difficulty patterns.

Adaptive Ensemble Weighting. Once the difficulty score $\tilde{\rho}_k^{(t)}$ for each surrogate model is obtained, we compute the adaptive ensemble weight of each model using a softmax function:

$$w_k^{(t)} = \frac{\exp\left(\beta \tilde{\rho}_k^{(t)}\right)}{\sum_{j=1}^{K} \exp\left(\beta \tilde{\rho}_j^{(t)}\right)}, \tag{15}$$

where β controls the sharpness of the weight distribution. A larger β leads to more pronounced differences among model weights, thereby emphasizing surrogate models that are more difficult to attack.

Algorithm 1 MVM-Attack

Require: Clean image x, target image x_t; surrogate models $\{f_k\}_{k=1}^{K}$; ℓ_∞ budget ϵ; iterations T; step size α; momentum μ; views N; block-wise transformations $\{\mathcal{T}_i\}_{i=1}^{N}$; temperature β; decay λ.
Ensure: Adversarial example x^{adv}.

$\delta \leftarrow \mathbf{0}$; $g \leftarrow \mathbf{0}$
Pre-compute target embeddings $y_k \leftarrow f_k(x_t)$, $\forall k \in \{1, \ldots, K\}$
Initialize difficulty memory buffers $\mathcal{M}_k \leftarrow [\,]$, $\forall k$
for $t = 0$ **to** $T - 1$ **do**
  $x^{adv} \leftarrow x + \delta$
  Multi-view sampling: generate $\{x_i^{adv}\}_{i=1}^{N}$ via $x_i^{adv} \leftarrow \mathcal{T}_i(x^{adv})$ (Eq. (2)–(9))
  for $k = 1$ **to** K **do**
    Multi-view match:
    $s_k \leftarrow \frac{1}{N}\sum_{i=1}^{N} \mathrm{sim}\big(f_k(x_i^{adv}), y_k\big)$ (Eq. (11))
  end for
  Difficulty scoring: normalize $\{s_k\}$ to $\{\rho_k\}$ using min–max inversion (Eq. (12))
  for $k = 1$ **to** K **do**
    Update memory buffer $\mathcal{M}_k \leftarrow \mathrm{Push}(\mathcal{M}_k, \rho_k)$
    Memory smoothing: $\bar{\rho}_k \leftarrow \mathrm{EMA}(\mathcal{M}_k; \lambda)$ (Eq. (13))
    History fusion: $\tilde{\rho}_k \leftarrow \gamma_t \bar{\rho}_k + (1 - \gamma_t)\rho_k$ (Eq. (14))
  end for
  Ensemble weighting: $w_k \leftarrow \mathrm{Softmax}(\beta \tilde{\rho}_k)$ over $k = 1..K$ (Eq. (15))
  Final objective: $\mathcal{L} \leftarrow \sum_{k=1}^{K} w_k \mathcal{L}_{\mathrm{MV}}^{(f_k)}(\delta)$ (Eq. (16))
  Momentum update: $g \leftarrow \mu g + \nabla_\delta \mathcal{L}$
  Projected step: $\delta \leftarrow \Pi_{\|\delta\|_\infty \le \epsilon}(\delta + \alpha \cdot \mathrm{sign}(g))$
end for
return $x + \delta$

By incorporating memory-smoothed difficulty estimates and adaptively adjusting ensemble weights, the proposed weighting strategy stabilizes the optimization process and effectively mitigates gradient conflicts and optimization noise commonly encountered in multi-model ensemble attacks. As a result, the adversarial perturbation is guided toward semantic vulnerability regions that are consistently shared across multiple surrogate models, thereby improving both attack robustness and transferability.

D. Overall Optimization Procedure

Given an input image x, we initialize the adversarial perturbation as $\delta^{(0)} = \mathbf{0}$. At iteration t, the adversarial example is $x^{adv} = x + \delta^{(t)}$. For each surrogate model f_k, we compute the multi-view feature optimization loss $\mathcal{L}_{\mathrm{MV}}^{(f_k)}(\delta^{(t)})$ and the corresponding memory-guided ensemble weight $\{w_k^{(t)}\}_{k=1}^{K}$. The final objective at iteration t is defined as:

$$\mathcal{L}_{\mathrm{final}}^{(t)}(\delta) = \sum_{k=1}^{K} w_k^{(t)}\, \mathcal{L}_{\mathrm{MV}}^{(f_k)}(\delta). \tag{16}$$

We then compute the gradient of the objective with respect to δ and update the perturbation using a momentum-based signed gradient update [25]:

$$g^{(t)} = \mu g^{(t-1)} + \nabla_\delta \mathcal{L}_{\mathrm{final}}^{(t)}, \tag{17}$$

$$\delta^{(t+1)} = \Pi_{\|\delta\|_\infty \le \epsilon}\Big(\delta^{(t)} + \alpha \cdot \mathrm{sign}\Big(g^{(t)}\Big)\Big), \tag{18}$$

where μ is the momentum factor, α is the step size, and $\Pi_{\|\delta\|_\infty \le \epsilon}(\cdot)$ denotes the projection operator onto the ℓ_∞-ball. The above procedure is repeated for T iterations, producing the final adversarial example $x^{adv} = x + \delta^{(T)}$. The complete pipeline is summarized in Algorithm 1.

IV. Experiments

A. Setup

Datasets. Following prior work [17], we use 1,000 clean images from the NIPS 2017 Adversarial Attacks and Defenses Competition dataset, with a resolution of $224 \times 224 \times 3$. Target images are selected from the MSCOCO dataset [27] to form 1,000 source–target image pairs for targeted attack evaluation.

Experimental Settings. Following [16], [17], we adopt three CLIP-based visual encoders as surrogate models, including ViT-B/16, ViT-B/32, and ViT-g/14 trained on the LAION-2B dataset. The perturbation budget is set to $\epsilon = 16/255$ under the ℓ_∞ norm, and the attack step size is $1/255$. Other hyperparameters include: temperature $\beta = 1.0$, decay factor $\lambda = 0.9$, and 300 iterations for each attack. The effects of block size (M) and view count (N) on attack performance and computational overhead are examined in the ablation study. We evaluate transferability on eight mainstream VLMs: four open-source models (MiniGPT4-7B, LLaVA-1.5-7B, Qwen2.5-VL-7B, and Qwen2.5-VL-32B) and four closed-source commercial models (GPT-4o, Gemini-2.5-flash, Gemini-3.0-flash, and Claude-3.5-Sonnet). All experiments are conducted on an Ubuntu system equipped with an NVIDIA A6000 GPU (48 GB RAM).

Task Settings. We evaluate the transferability of targeted attacks under the image captioning task. For all target VLMs, we use a unified prompt: *"Describe this image in one concise sentence, no longer than 25 words."* The temperature parameter is set to a value close to deterministic decoding to reduce sampling noise and improve evaluation reproducibility.

Baseline Methods. We compare the proposed MVM-Attack with six representative baseline methods for transferable targeted attacks, including AttackVLM [22], AdvDiffVLM [23], AnyAttack [24], SSA-CWA [15], M-Attack [16], and FOA-Attack [17].

Evaluation Metrics. We adopt the widely used *LLM-as-a-judge* evaluation framework and select DeepSeek-V3.2 [28] as the judge model. A unified evaluation prompt is applied for all assessments. We measure attack effectiveness from two aspects: *average similarity score (AvgSim)* and *attack success rate (ASR)*.

Specifically, we first use the same target VLM to generate textual descriptions for the clean image x, the adversarial image x^{adv}, and the target image x_t, denoted as $\hat{y}(x)$, $\hat{y}(x^{adv})$, and $\hat{y}(x_t)$, respectively.

(1) Similarity Score. The judge model evaluates the semantic similarity between $\hat{y}(x^{adv})$ and $\hat{y}(x_t)$, producing a similarity score:

$$S\big(\hat{y}(x^{adv}),\, \hat{y}(x_t)\big) \in [0, 1]. \tag{19}$$

(2) Attack Success Rate. The judge model compares the generated captions $\hat{y}(x^{adv})$, $\hat{y}(x)$, and $\hat{y}(x_t)$ and outputs a

TABLE I
PERFORMANCE OF ASR AND AVGSIM ON DIFFERENT OPEN-SOURCE VLMS UNDER $\epsilon = 16/255$.

Method	Model	MiniGPT4-7B		LLaVA-1.5-7B		Qwen2.5-VL-7B		Qwen2.5-VL-32B	
		AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑
AttackVLM [22]	B/16	0.07	0.00	0.11	0.01	0.11	0.00	0.09	0.00
	B/32	0.06	0.00	0.09	0.00	0.10	0.00	0.09	0.00
	Laion	0.08	0.00	0.11	0.00	0.12	0.01	0.10	0.00
AdvDiffVLM [23]	Ensemble	0.10	0.01	0.14	0.03	0.11	0.01	0.11	0.02
AnyAttack [24]	Ensemble	0.17	0.02	0.23	0.08	0.17	0.04	0.36	0.02
SSA-CWA [15]	Ensemble	0.56	0.67	0.23	0.18	0.31	0.21	0.20	0.09
M-Attack [16]	Ensemble	0.61	0.71	0.67	0.81	0.60	0.66	0.64	0.53
FOA-Attack [17]	Ensemble	0.62	0.75	0.71	0.83	0.64	0.69	0.56	0.52
MVM-Attack(Ours)	Ensemble	**0.67**	**0.84**	**0.79**	**0.89**	**0.76**	**0.92**	**0.74**	**0.88**

TABLE II
PERFORMANCE OF ASR AND AVGSIM ON DIFFERENT CLOSED-SOURCE VLMS UNDER $\epsilon = 16/255$.

Method	Model	GPT-4o		Gemini-2.5-flash		Gemini-3.0-flash		Claude-3.5-Sonnet	
		AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑	AvgSim ↑	ASR ↑
AttackVLM [22]	B/16	0.10	0.00	0.09	0.00	0.10	0.00	0.10	0.02
	B/32	0.09	0.00	0.10	0.00	0.10	0.00	0.10	0.01
	Laion	0.11	0.00	0.10	0.00	0.11	0.00	0.09	0.00
AdvDiffVLM [23]	Ensemble	0.11	0.01	0.12	0.02	0.13	0.07	0.08	0.01
AnyAttack [24]	Ensemble	0.36	0.02	0.13	0.08	0.14	0.24	0.12	0.04
SSA-CWA [15]	Ensemble	0.20	0.09	0.16	0.04	0.46	0.24	0.13	0.01
M-Attack [16]	Ensemble	0.64	0.71	0.54	0.46	0.51	0.34	0.15	0.07
FOA-Attack [17]	Ensemble	0.62	0.64	0.56	0.48	0.56	0.34	0.16	0.06
MVM-Attack(Ours)	Ensemble	**0.70**	**0.79**	**0.67**	**0.61**	**0.61**	**0.52**	**0.34**	**0.28**

classification label $z \in \{A, B, C, D\}$, where A corresponds to the original description, B to the target description, C to a mixture of both, and D to neither. An attack is considered successful if and only if the output label is B. The attack success rate is defined as:

$$\mathrm{ASR} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[z_i = B], \tag{20}$$

where N is the number of evaluated samples and $\mathbb{I}[\cdot]$ denotes the indicator function. All evaluations adopt deterministic temperature settings, and the judge model follows the prompt templates described in [16], [29].

B. Comparative Results

Under a fixed perturbation budget of $\epsilon = 16/255$, the proposed method consistently outperforms existing baseline approaches on both open-source and closed-source VLMs. The quantitative results are reported in Table I and Table II. On open-source models such as MiniGPT-4, LLaVA-1.5, and Qwen2.5, our method achieves attack success rates exceeding 80%, and further surpasses 90% on Qwen2.5-VL-7B, significantly outperforming all compared baselines.

On mainstream closed-source commercial models, including GPT-4o, Gemini-2.5 and Gemini-3.0, the proposed method attains the highest attack success rates of 76%, 61% and 52%, respectively. These results indicate that our approach is capable of effectively compromising advanced models equipped with strong multimodal reasoning and safety alignment mechanisms. In contrast, the attack success rate on Claude-3.5 is relatively lower (28%), reflecting notable differences in robustness across commercial models, potentially arising from variations in visual encoders or cross-modal fusion strategies.

Overall, these experimental results demonstrate the effectiveness and strong transferability of the proposed method in targeted attack scenarios.

C. Visualization Analysis

Fig. 3 presents the adversarial images and corresponding perturbations produced by different methods. As shown, the perturbation patterns of baseline methods exhibit notable dispersion and irregular spatial distributions. In contrast, the proposed method generates perturbations that are more spatially concentrated and display clearer structural consistency,

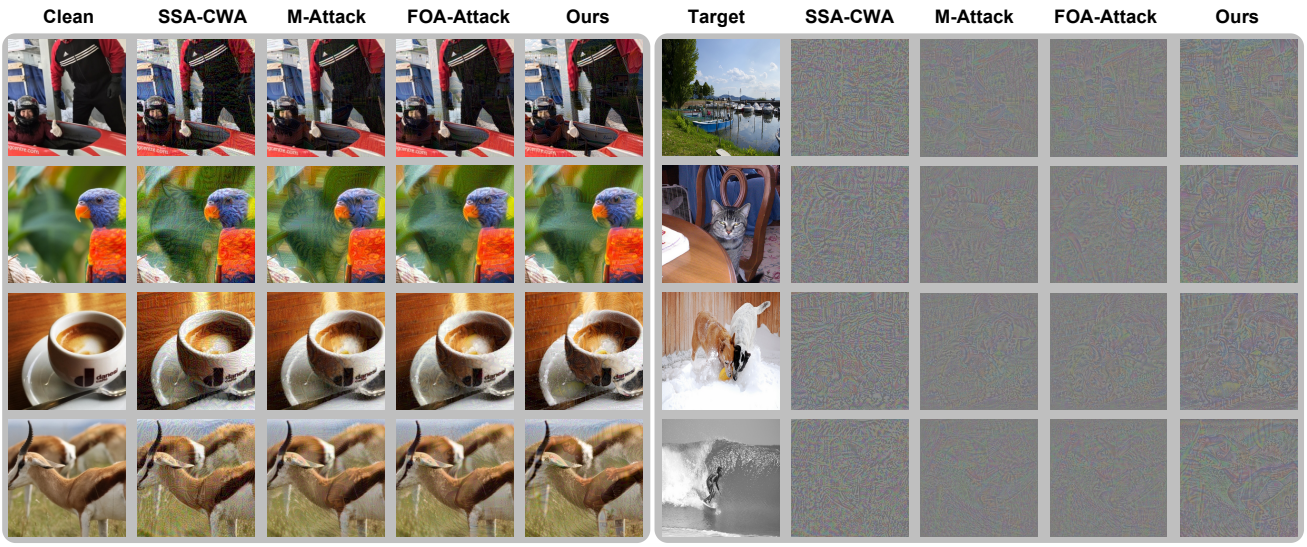

Fig. 3. Visualization of adversarial images and perturbation.

indicating that the multi-view feature optimization encourages perturbations to remain stable across diverse geometric views.

D. Ablation Study

As shown in Fig. 4, the complete MVM-Attack achieves the highest attack success rates on Qwen-2.5, LLaVA-1.5, GPT-4o, and Gemini-2.5-flash. Removing the memory-guided ensemble weighting component (w/o MGEW) results in a noticeable and stable degradation in performance across all models.When the multi-view feature optimization module is further removed (w/o MVFO), the method degenerates into the basic MI-FGSM attack [25], leading to a drastic drop in attack success rates to 0.02 and 0.01, effectively rendering the attack ineffective. A plausible explanation is that conventional attack methods optimize perturbations by treating the image as a whole, neglecting the hierarchical visual–semantic understanding implicitly encoded in VLMs. Moreover, simple unstructured perturbations are insufficient to penetrate the strongly aligned feature representations learned through vision–language pretraining.

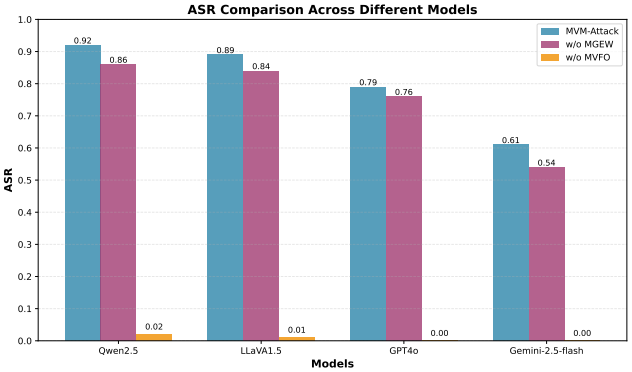

Fig. 4. Ablation study of different components.

E. Hyperparameter Analysis

We analyze the impact of key hyperparameters in MVM-Attack on attack performance, specifically the number of image blocks M and the number of views N. The number of blocks determines the granularity of the spatial transformations. As shown in Fig. 5, when the number of blocks is smaller, the attack performance is highly sensitive to slight perturbations, whereas increasing the number of blocks improves the attack stability, though it may weaken the attack in some cases. We observe that the best performance is achieved when using 4×4 blocks, which strikes a balance between local feature stability and overall robustness.

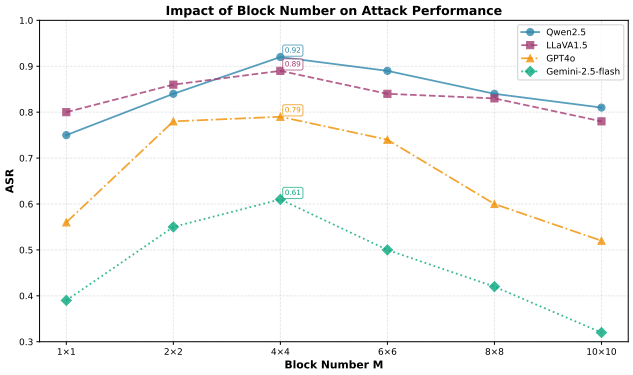

Fig. 5. Impact of block number on attack performance.

As shown in Table III, we examine the impact of the number of views N. Increasing N from 1 to 15 leads to significant gains in both AvgSim and ASR across models. In contrast, further increasing N from 15 to 25 results in only marginal performance improvement, while the computational cost rises nearly linearly. This indicates that $N = 15$ provides sufficiently diverse geometric transformations to achieve robust multi-view feature alignment and capture most

TABLE III
IMPACT OF VIEW NUMBER N ON ATTACK PERFORMANCE AND COMPUTATIONAL COST. TIME DENOTES THE AVERAGE GENERATION TIME PER ADVERSARIAL IMAGE (IN SECONDS).

N	Time (s)	GPT-4o		Gemini-2.5-flash	
		AvgSim	ASR	AvgSim	ASR
1	40	0.38	0.24	0.36	0.16
5	147	0.61	0.59	0.59	0.46
10	273	0.66	0.68	0.64	0.54
15	408	0.66	0.79	0.67	0.61
20	512	0.69	0.77	0.68	0.63
25	663	0.70	0.78	0.69	0.64

transferable adversarial patterns. Beyond this point, additional views contribute little to generalization but incur substantial overhead. Therefore, $N = 15$ is selected as the default setting to optimally balance attack effectiveness and computational efficiency.

V. CONCLUSION

In this work, we introduced MVM-Attack, a transferable targeted adversarial attack framework for vision–language models that integrates multi-view feature optimization with a memory-guided ensemble weighting mechanism. Extensive experiments across diverse open-source and closed-source VLMs show that our method consistently outperforms state-of-the-art targeted attack approaches under identical perturbation constraints, yielding higher attack success rates and more reliable target alignment. These results indicate that, despite their strong task performance, VLMs remain vulnerable to targeted adversarial manipulation, highlighting the need for more comprehensive robustness evaluation. Looking forward, future research may further improve the stealthiness of adversarial perturbations and reduce the computational overhead associated with multi-view optimization, thereby enhancing the practicality and broader applicability of transferable targeted attacks in real-world scenarios.

REFERENCES

[1] Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision[C]//International conference on machine learning. PmLR, 2021: 8748-8763.

[2] Li J, Li D, Xiong C, et al. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation[C]//International conference on machine learning. PMLR, 2022: 12888-12900.

[3] Zhu D, Chen J, Shen X, et al. Minigpt-4: Enhancing vision-language understanding with advanced large language models[J]. arXiv preprint arXiv:2304.10592, 2023.

[4] Liu H, Li C, Wu Q, et al. Visual instruction tuning[J]. Advances in neural information processing systems, 2023, 36: 34892-34916.

[5] Bai J, Bai S, Yang S, et al. Qwen-vl: A frontier large vision-language model with versatile abilities[J]. arXiv preprint arXiv:2308.12966, 2023, 1(2): 3.

[6] Salaberria A, Azkune G, de Lacalle O L, et al. Image captioning for effective use of language models in knowledge-based visual question answering[J]. Expert Systems with Applications, 2023, 212: 118669.

[7] Luu D T, Le V T, Vo D M. Questioning, answering, and captioning for zero-shot detailed image caption[C]//Proceedings of the Asian Conference on Computer Vision. 2024: 242-259.

[8] Li Z, Liu D, Zhang C, et al. Enhancing Advanced Visual Reasoning Ability of Large Language Models[C]//Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. 2024: 1915-1929.

[9] Achiam J, Adler S, Agarwal S, et al. Gpt-4 technical report[J]. arXiv preprint arXiv:2303.08774, 2023.

[10] Team G, Anil R, Borgeaud S, et al. Gemini: a family of highly capable multimodal models[J]. arXiv preprint arXiv:2312.11805, 2023.

[11] Schlarmann C, Hein M. On the adversarial robustness of multi-modal foundation models[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 3677-3685.

[12] Cui X, Aparcedo A, Jang Y K, et al. On the robustness of large multimodal models against image adversarial attacks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 24625-24634.

[13] Yang Y, Wang L, Zhang J, et al. Multi-Faceted Attack: Exposing Cross-Model Vulnerabilities in Defense-Equipped Vision-Language Models[J]. arXiv preprint arXiv:2511.16110, 2025.

[14] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.

[15] Dong Y, Chen H, Chen J, et al. How robust is google's bard to adversarial image attacks?[J]. arXiv preprint arXiv:2309.11751, 2023.

[16] Li Z, Zhao X, Wu D D, et al. A Frustratingly Simple Yet Highly Effective Attack Baseline: Over 90% Success Rate Against the Strong Black-box Models of GPT-4.5/4o/o1[C]//ICML 2025 Workshop on Reliable and Responsible Foundation Models.

[17] Jia X, Gao S, Qin S, et al. Adversarial Attacks against Closed-Source MLLMs via Feature Optimal Alignment[J]. arXiv preprint arXiv:2505.21494, 2025.

[18] Ilyas A, Engstrom L, Athalye A, et al. Black-box adversarial attacks with limited queries and information[C]//International conference on machine learning. PMLR, 2018: 2137-2146.

[19] Dong Y, Cheng S, Pang T, et al. Query-efficient black-box adversarial attacks guided by a transfer-based prior[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 44(12): 9536-9548.

[20] Wu L, Zhao L, Pu B, et al. Boosting the Transferability of Adversarial Examples via Adaptive Attention and Gradient Purification Methods[C]//2024 International Joint Conference on Neural Networks (IJCNN). IEEE, 2024: 1-7.

[21] Zhang J, Huang Y, Xu Z, et al. Improving the adversarial transferability of vision transformers with virtual dense connection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(7): 7133-7141.

[22] Zhao Y, Pang T, Du C, et al. On evaluating adversarial robustness of large vision-language models[J]. Advances in Neural Information Processing Systems, 2023, 36: 54111-54138.

[23] Guo Q, Pang S, Jia X, et al. Efficient generation of targeted and transferable adversarial examples for vision-language models via diffusion models[J]. IEEE Transactions on Information Forensics and Security, 2024.

[24] Zhang J, Ye J, Ma X, et al. AnyAttack: Towards Large-scale Self-supervised Adversarial Attacks on Vision-language Models[C]//Proceedings of the Computer Vision and Pattern Recognition Conference. 2025: 19900-19909.

[25] Dong Y, Liao F, Pang T, et al. Boosting adversarial attacks with momentum[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 9185-9193.

[26] Kurakin A, Goodfellow I, Bengio S, et al. Adversarial attacks and defences competition[M]//The NIPS'17 Competition: Building Intelligent Systems. Cham: Springer International Publishing, 2018: 195-231.

[27] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Cham: Springer International Publishing, 2014: 740-755.

[28] Liu A, Mei A, Lin B, et al. Deepseek-v3. 2: Pushing the frontier of open large language models[J]. arXiv preprint arXiv:2512.02556, 2025.

[29] Hu K, Yu W, Zhang L, et al. Transferable adversarial attacks on black-box vision-language models[J]. arXiv preprint arXiv:2505.01050, 2025.