

# Pushing the Frontier of Black-Box LVLM Attacks via Fine-Grained Detail Targeting

Xiaohan Zhao, Zhaoyi Li, Yaxin Luo, Jiacheng Cui, Zhiqiang Shen<sup>†</sup>

VILA Lab, Department of Machine Learning, MBZUAI

<https://vila-lab.github.io/M-Attack-V2-Website/>

{xiaohan.zhao, zhaoyi.li, yaxin.luo, jiacheng.cui, zhiqiang.shen}@mbzuai.ac.ae

## Abstract

Black-box adversarial attacks on Large Vision-Language Models (LVLMs) are challenging due to missing gradients and complex multi-modal boundaries. While prior state-of-the-art M-Attack performs well using local crop-level matching between source and target images, we find this induces high-variance, nearly orthogonal gradients across iterations, violating coherent local alignment and destabilizing optimization. We attribute this to (i) ViT translation sensitivity that yields spike-like gradients and (ii) structural asymmetry between source and target crops. We reformulate local matching as an asymmetric expectation over source transformations and target semantics, and build a gradient denoising upgrade to M-Attack. On the source side, *Multi-Crop Alignment (MCA)* averages gradients from multiple independently sampled local views per iteration to reduce variance. On the target side, *Auxiliary Target Alignment (ATA)* replaces aggressive target augmentation with a small auxiliary set from a semantically correlated distribution, producing a smoother, lower-variance target manifold. We further reinterpret momentum as *Patch Momentum*, replaying historical crop gradients, combined with a refined patch-size ensemble ( $PE^+$ ), this strengthens transferable directions. Together these modules form our M-Attack-V2 in this work, a simple, modular enhancement over M-Attack that substantially improves transfer-based black-box attacks on frontier LVLMs: boosting success rates on Claude-4.0 (✳) from **8%→30%**, Gemini-2.5-Pro (✧) from **83%→97%**, and GPT-5 (🟩) from **98%→100%**, outperforming all prior black-box LVLM attacks. Code and data are publicly available at [this link](https://vila-lab.github.io/M-Attack-V2-Website/).

<sup>†</sup>Corresponding Author. Preprint. February 7, 2026.

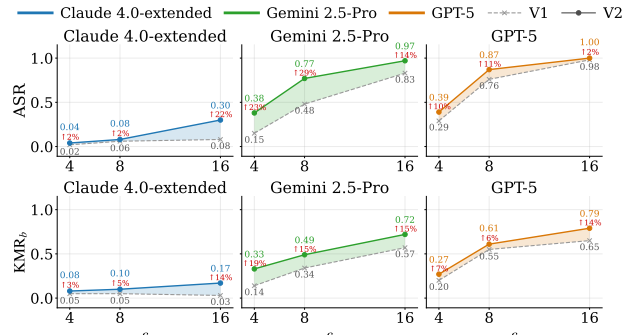


Figure 1. Improvement of M-Attack-V2 over M-Attack on up-to-date commercial black-box models (Claude 4, Gemini 2.5 and GPT-5). ASR and KMR stand for attack success rate and keyword matching rate, respectively.

## 1. Introduction

Large Vision-Language Models (LVLMs) have become foundational to modern AI systems, enabling multimodal tasks like image captioning (Hu et al., 2022; Salaberria et al., 2023; Chen et al., 2022b; Tschannen et al., 2023), VQA (Luu et al., 2024; Özdemir & Akagündüz, 2024), and visual reasoning (OpenAI, 2025). However, their visual modules remain vulnerable to adversarial attacks, subtle perturbations that mislead models while remaining imperceptible to humans. Prior efforts, including AttackVLM (Zhao et al., 2023), CWA (Chen et al., 2024), SSA-CWA (Dong et al., 2023a), AdvDiffVLM (Guo et al., 2024), and most effectively, recent state-of-the-art M-Attack (Li et al., 2025), which have exploited and addressed this weakness through local-level matching and surrogate model ensembles, surpassing 90% success rates on models like GPT-4o.

Despite promising performance of M-Attack, our analysis reveals that its gradient signals are highly unstable: Even overlapping large pixel regions, two consecutive local crops share *nearly orthogonal gradients*. In other words, high similarity in the pixel and embedding spaces does not translate to high similarity in the gradient space. The reason is that ViTs’ gradient pattern is sensitive to translation. A

tiny shift changes pixels contained in each token, altering self-attention. Moreover, patch-wise, spike-like gradient amplifies the mismatch within just a few pixels. We counter this effect by aggregating gradients from multiple crops within the same iteration, a strategy we call *Multi-Crop Alignment* (MCA). From a theoretical angle, MCA aggregates gradients across multiple views in a single iteration, smoothing local inconsistencies and improving cross-crop gradient stability.

We further observe that the source and target transformations in M-Attack operate in different semantic spaces: one emphasizing extraction, the other generalization. Aggressive target augmentation introduces harmful variance. Our *Auxiliary Target Alignment* (ATA) mitigates this by identifying semantically similar auxiliary images to create a low-variance embedding subspace, then applying only mild shifts to enhance transferability without destabilizing the optimization. Classic momentum is reinterpreted under this framework as *Patch Momentum* (PM), a replay mechanism that recycles past gradients across random crops to stabilize optimization. In parallel, we also re-examine and enrich M-Attack’s model selection criterion and choose a delicately selected ensemble set with diverse patch sizes to mitigate the difficulty in cross-patch transfer, of which we find that the attention concentrates more on the main object. We term it *Patch Ensemble*<sup>+</sup> (PE<sup>+</sup>).

Together, these components, MCA, ATA, PM, and PE<sup>+</sup>, form the basis of M-Attack-V2, a robust gradient denoising framework that significantly outperforms existing black-box attack methods. Our method raises attack success rates from 98%→100% on GPT-5, 8%→30% on Claude-4, and 83%→97% on Gemini-2.5-Pro, achieving state-of-the-art performance across the board. This study not only offers a practical, modular attack strategy but also sheds light on the gradient behavior of ViT-based LVLMs under local perturbations. To summarize, our contributions are:

- We show crop-level matching yields high-variance, near-orthogonal gradients (from ViT translation sensitivity and source/target crop asymmetry), destabilizing black-box optimization.
- We recast local matching as an asymmetric expectation and introduce MCA (multi-view gradient averaging) + ATA (auxiliary semantically correlated targets) to reduce variance and smooth the target manifold.
- We add Patch Momentum + refined PE<sup>+</sup> to amplify transferable directions, delivering large ASR gains on frontier LVLMs (e.g., Claude-4.0 8%→30%, Gemini-2.5-Pro 83%→97%, GPT-5 98%→100%)

## 2. Related Work

**Large Vision Language Models.** Transformer-based LVLMs learn visual-semantic representations from large-scale image-text data, enabling tasks like image caption-

ing (Salaberria et al., 2023; Hu et al., 2022; Chen et al., 2022b; Tschannen et al., 2023), visual QA (Luu et al., 2024; Özdemir & Akagündüz, 2024), and cross-modal reasoning (Wu et al., 2025; Ma et al., 2023; Wang et al., 2024). Open-source models such as BLIP-2 (Li et al., 2022), Flamingo (Alayrac et al., 2022), and LLaVA (Liu et al., 2023) show strong benchmark performance. Commercial models like GPT-4o, Claude-3.5 (Anthropic, 2024a), and Gemini-2.0 (Team et al., 2023) offer advanced reasoning and real-world adaptability, with their successors, GPT-o3 (OpenAI, 2025), Claude 3.7-Sonnet (Anthropic, 2024b), and Gemini-2.5-Pro, able to reason over both text and images.

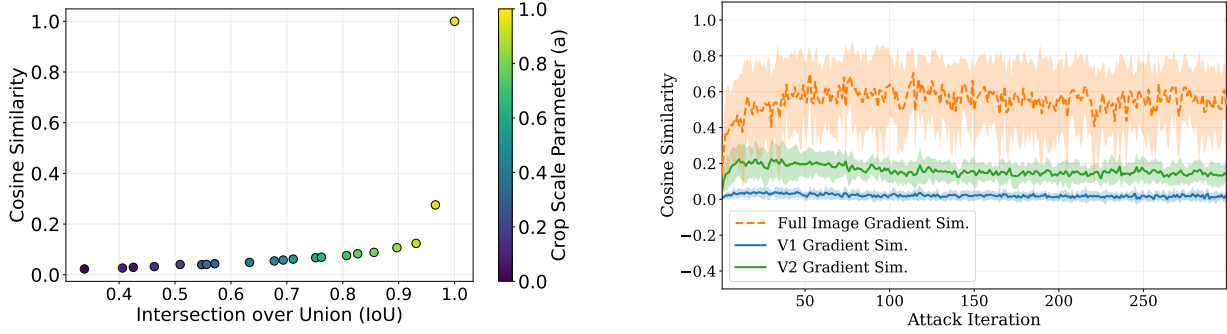
**LVLM transfer-based attack.** Black-box attacks include query-based (Dong et al., 2021; Ilyas et al., 2018) and transfer-based (Dong et al., 2018; Liu et al., 2017) (our focus in this work). AttackVLM (Zhao et al., 2023) pioneered transfer-based targeted LVLM attacks using CLIP (Radford et al., 2021) and BLIP (Li et al., 2022) surrogates, showing image-to-image feature matching beats cross-modal optimization, later adopted by (Chen et al., 2024; Guo et al., 2024; Dong et al., 2023a; Li et al., 2025). CWA (Chen et al., 2024) and SSA-CWA (Dong et al., 2023a) extended this to Bard (Team et al., 2023): CWA improves transferability via sharpness-aware minimization (Foret et al., 2021; Chen et al., 2022a), while SSA-CWA adds spectrum-guided augmentation with SSA (Long et al., 2022). AnyAttack (Zhang et al., 2024) uses image-image matching with large-scale pretraining and fine-tuning. AdvDiffVLM (Guo et al., 2024) integrates feature matching into diffusion guidance and proposes Adaptive Ensemble Gradient Estimation (AEGE) for smoother ensemble scores. M-Attack further surpasses these via local-level matching of source and target with an ensemble of surrogate models and diverse patch sizes, and FOA-Attack (Jia et al., 2025) extends alignment from CLS to local patch tokens for additional gains. However, local-level matching still has limitations, we next introduce its background before analyzing and addressing them.

## 3. Approach

### 3.1. Preliminaries and Limitations of Local-level Matching-Based Methods

**Local-level matching in M-Attack.** Consider a clean source image  $\tilde{\mathbf{X}}_{\text{sou}}$  and a target image  $\mathbf{X}_{\text{tar}}$ . The objective of black-box transfer attacks is to minimally perturb the source image by  $\delta$  so that the perturbed image  $\mathbf{X}_{\text{sou}} = \tilde{\mathbf{X}}_{\text{sou}} + \delta$  aligns semantically with the target under an inaccessible black-box model  $f_{\xi}$ . Due to the inaccessibility of  $f_{\xi}$ , surrogate models  $f_{\phi}$  approximate the semantic alignment via cosine similarity:

$$\arg \max_{\mathbf{X}_{\text{sou}}} \text{CS}(f_{\phi}(\mathbf{X}_{\text{sou}}), f_{\phi}(\mathbf{X}_{\text{tar}})) \quad \text{s.t.} \quad \|\delta\|_p \leq \epsilon, \quad (1)$$



(a) Gradient cosine similarity between two crops drops quickly with IoU and falls below 0.1 when IoU < 0.8, despite shared pixels. (b) Cosine similarity between consecutive source gradients ( $\nabla_{\hat{\mathbf{x}}_i^s}, \nabla_{\hat{\mathbf{x}}_{i+1}^s}$ ) across iterations.

Figure 2. Cosine similarity of gradients from random crops. (a) Similarity vs. IoU between two crops in a fixed iteration, showing rapid decay and values < 0.1 once IoU < 0.8. (b) Similarity between consecutive source gradients. V1’s gradient similarity is almost zero, while v2 improves it to around 0.2. Results are averaged over 200 runs.

where CS denotes cosine similarity. M-Attack enhances Eq. (1) using *local-level matching*. At iteration  $i$ , it applies predefined local transformations  $\mathcal{T}_s$  and  $\mathcal{T}_t$  to extract local area  $\hat{\mathbf{x}}_i^s$  from the source  $\mathbf{X}_{\text{sou}}$  and  $\hat{\mathbf{x}}_i^t$  from the target  $\mathbf{X}_{\text{tar}}$ , respectively. These transformations satisfy essential properties, such as spatial overlap and diversified coverage of extracted local regions  $\{\hat{\mathbf{x}}_i\}$  (Li et al., 2025). Formally, the local-level matching optimizes:

$$\mathcal{M}_{\mathcal{T}_s, \mathcal{T}_t} = \mathbb{E}_{f_{\phi_j} \sim \phi} [\text{CS}(f_{\phi_j}(\hat{\mathbf{x}}_i^s), f_{\phi_j}(\hat{\mathbf{x}}_i^t))], \quad (2)$$

where  $f_{\phi_j}$  is sampled from an ensemble of surrogate models  $\phi$ . Intuitively, matching local image regions instead of entire images enhances the semantic precision of perturbations by directing optimization towards semantically significant details. Despite its effectiveness, M-Attack encounters a critical challenge of *unexpectedly low* gradient similarity, which we investigate in detail next.

**Extremely low gradient overlap.** In M-Attack two random crops  $\hat{\mathbf{x}}_i^s$  and  $\hat{\mathbf{x}}_i^t$  are matched at every iteration. One would expect the gradients inside the shared region of two successive source crops, i.e.,  $(\nabla_{\hat{\mathbf{x}}_i^s} \mathcal{M}_{\mathcal{T}_s, \mathcal{T}_j}, \nabla_{\hat{\mathbf{x}}_{i+1}^s} \mathcal{M}_{\mathcal{T}_s, \mathcal{T}_j})$ , to correlate, because the underlying pixels partly coincide. Supprisingly, Fig. 2b shows the opposite: their cosine similarity is *almost zero*. We then keep the same fixed iteration and repeatedly draw two random crops at different scales and check the cosine similarity of their gradients (Fig. 2a).

Our finding reveals an exponential decay that plateaus below 0.1 once the overlap is smaller than 0.80 IoU. We find that this unexpectedly low gradient overlap mainly stems from two factors: ViTs’ inherent translation sensitivity and an overlooked asymmetry in the local matching framework. We first examine the translation effect.

1) *Patch-wise, spike-like gradient sensitive to translation.* Because ViTs tokenize images on a fixed, non-overlapping grid, even sub-pixel changes each patch’s token mix. These token changes ripple through self-attention, altering weights

and redirecting gradients for *all* tokens, so the resulting pixel-level gradient pattern diverges sharply. Worse, gradient magnitudes are uneven. Therefore, even similar patterns but missing a few pixels might completely break gradient similarity (Fig. 3b).

2) *Asymmetric Transform Branches.* In M-Attack, both the *source* and *target* images are cropped, yet playing distinct roles. Cropping the source acts directly in *pixel space*: it rearranges patch embeddings and attention weights in the forward pass, ending up with guidance of different views. By contrast, cropping the target solely translates the target representation, thereby shifting the reference embedding in *feature space*. One sculpts the perturbation, while another moves the goalpost, formulating asymmetric matching. M-Attack overlooked this and implementations target translation alternate between a *radical* crop and an identity map, struggles between explore-exploitation trade-off and potentially risk in high variance of target embedding.

### 3.2. Asymmetric Matching over Expectation

To mitigate the issues above, we begin by systematically reformulating the original objective function as an expectation over local transformations within an asymmetric matching framework:

$$\min_{\|\mathbf{X}_{\text{sou}}\|_p \leq \epsilon} \mathbb{E}_{\mathcal{T} \sim \mathcal{D}, y \sim \mathcal{Y}} [\mathcal{L}(f(\mathcal{T}(\mathbf{X}_{\text{sou}})), y)], \quad (3)$$

where  $\mathcal{D}$  represents the distribution of local transformations, and  $\mathcal{Y}$  denotes the distribution over target semantics.  $\|\cdot\|_p$  is  $\ell_p$  constraint for imperceptibility. Conceptually, this formulation corresponds to embedding specific semantic content  $y$  into a locally transformed area  $\mathcal{T}(\mathbf{X}_{\text{sou}})$ , thus highlighting the intrinsic asymmetry compared to M-Attack’s original formulation. Within this framework, our proposed enhancements, i.e., *Multi-Crop Alignment* (MCA) and *Auxiliary Target Alignment* (ATA), can be interpreted as strategies to improve the accuracy of the expectation estimation and the

sampling quality of the semantic distribution  $\mathcal{Y}$ .

### 3.3. Gradient Denoising via Multi-Crop Alignment

To obtain a low-variance estimate of expected loss gradient  $\mathbb{E}_{\mathcal{T} \sim \mathcal{D}, y \sim \mathcal{Y}} [\nabla_{\mathbf{X}_{\text{sou}}} \mathcal{L}(f(\mathcal{T}(\mathbf{X}_{\text{sou}})), y)]$ , we draw  $K$  independent crops  $\{\mathcal{T}\}_{k=1}^K$  and average their individual gradients:

$$\nabla_{\mathbf{X}_{\text{sou}}} \hat{\mathcal{L}}(\mathbf{X}_{\text{sou}}) = \frac{1}{K} \sum_{k=1}^K \nabla_{\mathbf{X}_{\text{sou}}} \mathcal{L}(f(\mathcal{T}_k(\mathbf{X}_{\text{sou}})), y). \quad (4)$$

This *Multi-Crop Alignment* is an unbiased Monte-Carlo estimator, reducing the variance with  $K > 1$ .

**Theorem 3.1.** Let  $g_k = \nabla_{\mathbf{X}_{\text{sou}}} \mathcal{L}(f(\mathcal{T}_k(\mathbf{X}_{\text{sou}})), y)$  denote the gradient from  $\mathcal{T}_k$ ,  $\mu = \mathbb{E}[g_k]$ ,  $\sigma^2 = \mathbb{E}[\|g_k - \mu\|_2^2]$  denote the mean and variance, and  $p_{k\ell}$  denote the pair-wise correlation  $p_{k\ell} = \frac{\langle g_k - \mu, g_\ell - \mu \rangle}{\|g_k - \mu\|_2^2 \|g_\ell - \mu\|_2^2}$ . The gradient variance from  $K$  averaged crops is bounded by:

$$\text{Var} \left( \frac{1}{K} \sum_{k=1}^K g_k \right) \leq \frac{\sigma^2}{K} + \frac{K-1}{K} \bar{p} \sigma^2, \quad (5)$$

where  $\bar{p} = \mathbb{E}[p_{k\ell}]$ ,  $k \neq \ell$  is the expectation of pair-wise correlation.

All crops share the same underlying image, so  $\bar{p} \neq 0$ . The ideal  $\sigma^2/K$  decay is therefore tempered by the correlation term  $\bar{p}\sigma^2$ . Empirically, averaging a modest number ( $K = 10$ ) of almost-orthogonal gradients still yields benefit, since the uncorrelated component of the variance shrinks as  $1/K$ . Simultaneously, the optimizer leverages multiple diverse transformations per update, with minimal interference among almost orthogonal gradients. Fig. 3a illustrates an accelerated convergence with  $K = 10$ , with margin improvement provided by  $K = 100$ .

This averaging also alleviates the known translation sensitivity of ViTs. As shown in Fig. 3b, using two crop sets yields noticeably higher gradient consistency than the single-crop alignment in M-Attack. In MCA, high-activity regions remain stable (upper left and center right), while the single-crop case shifts focus from center right to lower left. As a result, gradient similarity across iterations increases from near zero in M-Attack to around 0.2 (Fig. 2b).

### 3.4. Improved Sampling Quality via Auxiliary Target Alignment

Selecting a representative target embedding  $y \in \mathcal{Y}$  is challenging because the underlying distribution  $\mathcal{Y}$  is not observable. M-Attack mitigates this by seeding at the unaltered target embedding  $f(\mathbf{X}_{\text{tar}})$  and exploring its vicinity with transformed views  $f(\mathcal{T}_i(\mathbf{X}_{\text{tar}}))$  thereby sketching a locally semantic manifold that serves as a proxy for  $\mathcal{Y}$ . However, the exploration-exploitation trade-off remains problematic.

*Radical* transformations leap too far, dragging  $y$  outside the genuine target region; *conservative* transformations, while semantically faithful, barely shift the embedding, leaving the optimization starved of informative signal.

To stabilize this process, we introduce  $P$  auxiliary images  $\{\mathbf{X}_{\text{aux}}^{(p)}\}_{p=1}^P$  as an **auxiliary set** that acts as additional anchors, collectively forming a richer sub-manifold of aligned embeddings. During each update, we apply a *mild* random transformation  $\tilde{\mathcal{T}} \sim \tilde{\mathcal{D}}$  to every anchor, nudging the ensemble in a coherent yet restrained manner and thus providing low-variance, information-rich gradients for optimization. Let  $y_0 = f(\hat{\mathcal{T}}_0(\mathbf{X}_{\text{tar}}))$ ,  $\tilde{y}_p = f(\tilde{\mathcal{T}}_p(\mathbf{X}_{\text{aux}}^{(p)}))$  denote sampled semantics in one iteration. The objective  $\hat{\mathcal{L}}$  in Equ. (4) becomes

$$\hat{\mathcal{L}} = \frac{1}{K} \sum_{k=1}^n \left[ \mathcal{L}(f(\mathcal{T}_k(\mathbf{X}_{\text{sou}})), y_0) + \frac{\lambda}{P} \sum_{p=1}^P \mathcal{L}(f(\mathcal{T}_k(\mathbf{X}_{\text{sou}})), \tilde{y}_p) \right], \quad (6)$$

where  $\lambda \in [0, 1]$  interpolates between the original target and its auxiliary neighbors.  $\lambda = 0$  reduce to M-Attack local-local matching with single target. ATA trade-off exploration (auxiliary diversity) and exploitation (main-target fidelity), providing low-variance, semantics-preserving updates. The auxiliary set can be built in various ways, e.g., via image-image retrieval or diffusion methods. We now theoretically analyze ATA with three mild assumptions:

**Assumption 3.2** (Lipschitz surrogate). Surrogate  $f$  is  $L$ -continuous:  $\|f(y) - f(x)\| \leq L\|y - x\|$ .

**Assumption 3.3** (Bounded Auxiliary Data). For auxiliary data  $\mathbf{X}_{\text{aux}}^{(p)}$  retrieved via semantic similarity to a target  $\mathbf{X}_{\text{tar}}$ , we have:  $\mathbb{E}[\|f(\mathbf{X}_{\text{aux}}^{(p)}) - f(\mathbf{X}_{\text{tar}})\|] \leq \delta$  (justification in the appendix).

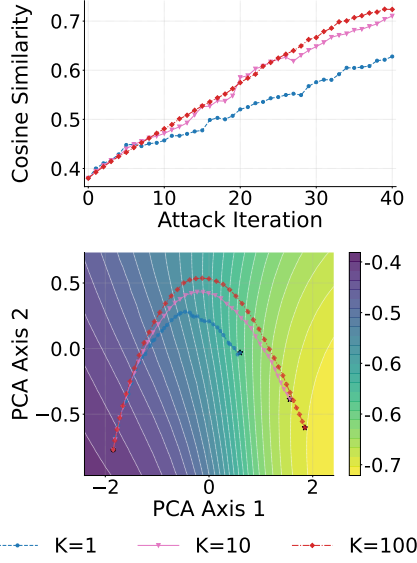
**Assumption 3.4** (Bounded transformation). Random transformation  $\mathcal{T} \sim D_\alpha f$  has bounded pixel-level distortion:  $\mathbb{E}[\|\mathcal{T}(\mathbf{X}) - \mathbf{X}\|] \leq \alpha$

**Theorem 3.5.** Let  $\mathcal{T} \sim D_\alpha$  denote the transformation used in M-Attack, and  $\tilde{\mathcal{T}} \sim D_{\tilde{\alpha}}$  with  $\tilde{\alpha} \ll \alpha$  the transformation in M-Attack-V2. Define **embedding drift** of transformation  $\mathcal{T}$  applied to  $\mathbf{X}$  on model  $f$  as:  $\Delta_{\text{drift}}(\mathcal{T}; \mathbf{X}) := \mathbb{E}_{\mathcal{T}}[\|f(\mathcal{T}(\mathbf{X})) - f(\mathbf{X}_{\text{tar}})\|]$ . Then, we have:

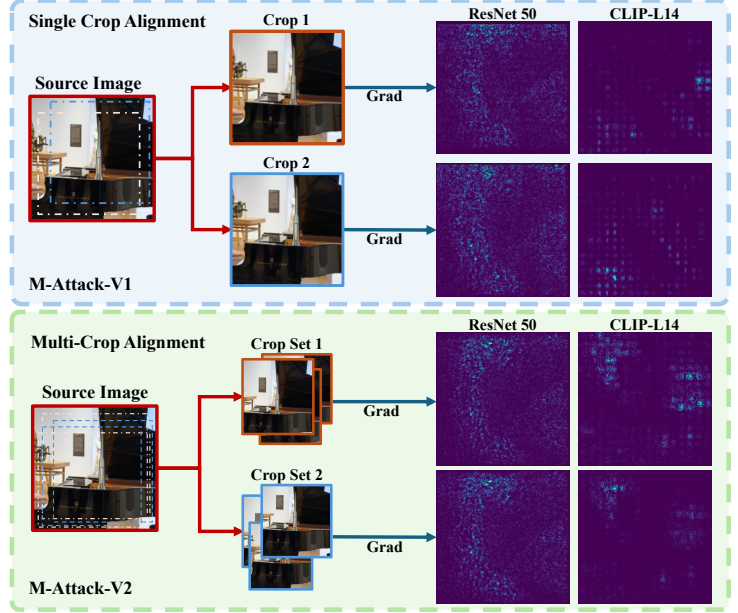
$$\begin{aligned} \Delta_{\text{drift}}(\mathcal{T}; \mathbf{X}_{\text{tar}}) &\leq L\alpha \\ \Delta_{\text{drift}}(\tilde{\mathcal{T}}; \mathbf{X}_{\text{aux}}^{(p)}) &\leq L\tilde{\alpha} + \delta. \end{aligned} \quad (7)$$

Specifically, the term  $L\alpha$  captures the inherent asymmetry caused by transformations in pixel space, requiring the multiplier  $L$  to map pixel-level perturbations into embedding space. In contrast, the auxiliary data *directly operates* in embedding space, leading to a manageable bound  $\delta$ . Practically,





(a) Comparison of optimization trajectories with different  $K$ ,  $K = 1$  refers to single crop alignment.



(b) Gradient pattern between different crop strategies in M-Attack and M-Attack-V2.

Figure 3. Comparison of: a) different trajectories against different  $K$ ; b) gradient pattern of single crop alignment against multi-crop alignment (MCA). The gradient pattern of ResNet 50 remains consistent when large pixels are overlapped, while the gradient pattern of ViTs changes dramatically. MCA helps to smooth out this impact.

estimating  $\delta$  is notably easier than estimating  $L\alpha$ . Lower  $\delta$  inherently indicates better semantic alignment, allowing M-Attack-V2 to operate effectively under reduced distortion ( $\tilde{\alpha} \ll \alpha$ ). Thus, ATA strategically allocates its shift budget toward more meaningful exploration via  $\delta$ , achieving a sweet spot between exploration and exploitation.

**Computation Analysis.** Each iteration back-propagates through the  $K$  source crops and only forward-propagates the  $P$  auxiliary targets. Since a backward pass is roughly twice as expensive as a forward pass, the per-iteration complexity is  $\mathcal{O}(K(3 + P))$ , doubling overhead when  $P = 3$ . Note that the additional overhead is parallelizable. The detailed analysis and comparison are in the appendix.

### 3.5. Patch Momentum with Built-in Replay Effect

Momentum, introduced in MI-FGSM (Dong et al., 2018), is widely adopted for transferability. Define the momentum buffer as:  $m_r = \beta_1 m_{r-1} + (1 - \beta_1) \nabla_{\hat{x}^s} \hat{\mathcal{L}}_r(\hat{x}^s)$ , where  $\beta_1 \in [0, 1]$  is the first-order momentum coefficient and  $\nabla_{\hat{x}^s} \hat{\mathcal{L}}_r(\hat{x}^s)$  is our MCA-ATA-estimated gradient  $g_r$  at iteration  $r$ . Under the local-matching view, this mechanism can be reinterpreted as formulating a streaming MCA to enforce temporal consistency across gradient directions in the space of random crops. Unrolling the EMA for pixel  $k$  exposes an alternative interpretation:

$$m_i(k) = (1 - \beta) \sum_{j=0}^i \beta^j \mathbf{1}\{k \in M_{i-j}\} g_{i-j}(k), \quad (8)$$

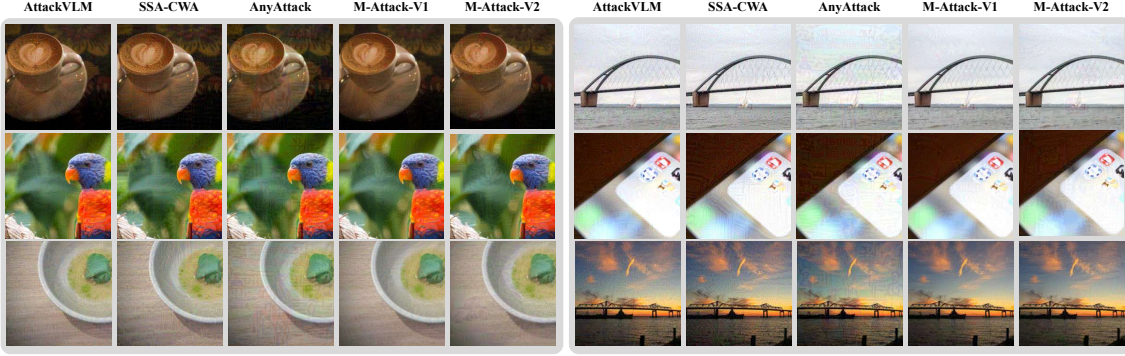
where  $M_i$  denotes the pixel indices included in iteration  $i$ ,  $m_i(k)$  and  $g_i(k)$  respectively denotes momentum and gradient for pixel  $k$ . Each crop involving pixel  $k$  is therefore replayed in future iterations with geometrically decaying weight, allowing rarely sampled regions (such as corners) to persist long enough to combat the gradient starvation. Spike-shaped gradients are further moderated by the Adam-style (Kingma, 2014) second moment,  $v_r = \beta_2 v_{r-1} + (1 - \beta_2) g_r^2$ , whose scaling effect is essential in our empirical study. The momentum does not directly improve gradient similarity but continuously re-injects historical crops across patches, effectively maintaining gradient directionality across local perturbation manifolds. We therefore term it *Patch Momentum* to distinguish.

The whole procedure, combining MCA, ATA, and PM, is detailed in Alg. 1. We use a different color to differentiate between M-Attack-V2 and M-Attack. We use PGD (Madry et al., 2018) with ADAM (Kingma, 2014) for line 12. Appendix presents analogous results for variants.

## 4. Experiments

### 4.1. Experimental Setup

**Metrics.** We adopt the evaluation protocol of M-Attack, reporting the *Attack Success Rate* (ASR) via *GPTScore* and the *Keywords Matching Rate* (KMR) at three thresholds  $\{0.25, 0.5, 1.0\}$ , denoted as  $\text{KMR}_a$ ,  $\text{KMR}_b$ , and  $\text{KMR}_c$  (Li et al., 2025). KMR measures semantic alignment using


 Figure 4. Visualization of adversarial samples under  $\epsilon = 8$ .

**Algorithm 1: M-Attack-V2**

**Input** : clean image  $\mathbf{X}_{\text{clean}}$ ; primary target  $\mathbf{X}_{\text{tar}}$ ;  
 auxiliary set  $\mathcal{A} = \{\mathbf{X}_{\text{aux}}^{(p)}\}_{p=1}^P$ ; **patch**  
 ensemble<sup>+</sup>  $\Phi^+ = \{\phi_j\}_{j=1}^m$ ; iterations  $n$ , step  
 size  $\alpha$ , perturbation budget  $\epsilon$ ; number of crops  
 $K$ , auxiliary weight  $\lambda$  ( $0 \leq \lambda \leq 1$ )  
**Output** : adversarial image  $\mathbf{X}_{\text{adv}}$

```

1  $\mathbf{X}_{\text{adv}} \leftarrow \mathbf{X}_{\text{clean}}$ 
2 for  $i = 1$  to  $n$  do
3     Draw  $K$  transforms  $\{\mathcal{T}_k\}_{k=1}^K \sim \mathcal{D}, g \leftarrow 0$ 
4     for  $k = 1$  to  $K$  (vectorizable) do
5         Draw  $\{\tilde{\mathcal{T}}_p\}_{p=0}^P \sim \tilde{\mathcal{D}}$ 
6         for  $j = 1$  to  $m$  do
7              $y_0 = f(\tilde{\mathcal{T}}_p(\mathbf{X}_{\text{tar}}))$ ,
8              $y_p = f(\tilde{\mathcal{T}}_p(\mathbf{X}_{\text{aux}}^{(p)})), p = 1, \dots, P$ 
9             Compute  $\hat{\mathcal{L}}_k = (f_{\phi_j}(\mathcal{T}_k(\mathbf{X}_{\text{sou}})), y_0) +$ 
10              $\frac{\lambda}{P} \sum_{p=1}^P \mathcal{L}(f_{\phi_j}(\mathcal{T}_k(x)), \tilde{y}_p)$ 
11              $g \leftarrow g + \frac{1}{Km} \nabla_{\mathbf{X}_{\text{sou}}} \hat{\mathcal{L}}_k$ 
12         end
13     end
14     Update  $\mathbf{X}_{\text{adv}}$  based on  $g$  with Patch Momentum
15 end
16 return  $\mathbf{X}_{\text{adv}}$ 
    
```

human-annotated keywords, considering a match successful if the rate exceeds threshold  $x$ . The evaluation follows M-Attack exactly.

**Surrogate candidates.** We follow surrogate selections from prior ensemble-based methods (Zhang et al., 2024; Dong et al., 2023a; Guo et al., 2024; Li et al., 2025). Our candidate pool covers CLIP variants (CLIP-B/16, B/32, L/14, CLIP<sup>†</sup>-G/14, CLIP<sup>†</sup>-B/32, CLIP<sup>†</sup>-H/14, CLIP<sup>†</sup>-B/16, CLIP<sup>†</sup>-BG/14), DinoV2 (Oquab et al., 2023) (Small, Base, Large), and BLIP-2 (Li et al., 2023)). **Victim models and dataset.** We evaluate state-of-the-art commercial MLLMs: GPT-4o/o3/5, Claude-3.7/4.0 (extended), and

Gemini-2.5-Pro-Preview (Team et al., 2023). Clean images are drawn from the NIPS 2017 Adversarial Attacks and Defenses Competition dataset (Kurakin et al., 2018a). Following SSA-CWA (Dong et al., 2023b) and M-Attack (Li et al., 2025), we randomly sample 100 images, retrieving *auxiliary sets* from the COCO training set (Lin et al., 2014) using CLIP-B/16 embedding similarity. Further results on a 1k-image subset are in the appendix, along with HuggingFace model identifiers. All the BLIP2 (Li et al., 2023) variants on Huggingface share the same vision encoder. Therefore, we use only one. The *milder target transformation* includes random resized crop ( $[0.9, 1.0]$ ), random horizontal flip ( $p = 0.5$ ), and random rotation ( $\pm 15^\circ$ ).

**Hyperparameters.** Unless noted, perturbations are bounded by  $\ell_\infty$  with  $\epsilon = 16$  and optimized for 300 steps. We set the step size to  $\alpha = 0.75$  for Claude and  $\alpha = 1.0$  for all other methods, mirroring M-Attack. For M-Attack-V2,  $\alpha = 1.275$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.99$  for momentum,  $K = 10$ ,  $P = 2$ , and  $\lambda = 0.3$  for MCA and ATA. Ablation of these parameters is presented in the appendix.

## 4.2. Selection of Surrogate Model

Ensembling multiple surrogate models is standard for improving black-box transferability, and recent work primarily designs advanced aggregation schemes over a *small* set of surrogates (Zhang et al., 2024; Guo et al., 2024). In contrast, we begin with a much larger candidate pool and find that *which* models are included already has a substantial impact: pre-selecting a few strong and complementary surrogates yields clear gains, even with plain averaging. We do not propose a novel aggregation rule; instead, we study a practical large-pool selection strategy *that has been less studied and reported before* but can serve as the very first stage of a complicated ensemble (pre-select useful candidates, then optionally apply more sophisticated aggregation). Concretely, we first profile embedding-level transferability across all candidates (Tab. 3), which shows that cross-model transfer, especially *cross-patch-size* transfer, is challenging. Guided by this, we retain only models that (i) perform well

Table 1. Comparison of attack methods on three black-box commercial LVLMs.  $\dagger$ : pre-trained on LAION (Schuhmann et al., 2022).

Method	Model	GPT-5				Claude 4.0-thinking				Gemini 2.5-Pro				Imperceptibility	
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	$\ell_1 \downarrow$	$\ell_2 \downarrow$
AttackVLM (Zhao et al., 2023)	<b>B/16</b>	0.08	0.03	0.02	0.05	0.03	0.00	0.00	0.00	0.08	0.04	0.00	0.00	0.034	0.040
	<b>B/32</b>	0.07	0.05	0.04	0.02	0.03	0.03	0.00	0.01	0.09	0.05	0.00	0.02	0.036	0.041
	<b>Laion<math>\dagger</math></b>	0.02	0.01	0.00	0.03	0.02	0.01	0.00	0.00	0.09	0.05	0.00	0.01	0.035	0.040
AdvDiffVLM (Guo et al., 2024)	<b>Ensemble</b>	0.04	0.02	0.01	0.01	0.04	0.01	0.01	0.01	0.03	0.01	0.00	0.00	0.064	0.095
SSA-CWA (Dong et al., 2023a)	<b>Ensemble</b>	0.08	0.04	0.00	0.08	0.03	0.02	0.01	0.05	0.05	0.03	0.01	0.08	0.059	0.060
AnyAttack (Zhang et al., 2024)	<b>Ensemble</b>	0.09	0.03	0.00	0.06	0.05	0.03	0.00	0.01	0.35	0.06	0.01	0.34	0.048	0.052
FOA-Attack (Jia et al., 2025)	<b>Ensemble</b>	0.90	0.67	0.23	0.94	0.13	0.09	0.00	0.13	0.61	0.80	0.15	0.86	0.031	0.036
M-Attack (Li et al., 2025)	<b>Ensemble</b>	0.89	0.65	0.25	0.98	0.12	0.03	0.00	0.08	0.81	0.57	0.15	0.83	<b>0.030</b>	<b>0.036</b>
M-Attack-V2 (Ours)	<b>Ensemble</b>	<b>0.92</b>	<b>0.79</b>	<b>0.30</b>	<b>1.00</b>	<b>0.27</b>	<b>0.17</b>	<b>0.04</b>	<b>0.30</b>	<b>0.87</b>	<b>0.72</b>	<b>0.22</b>	<b>0.97</b>	0.038	0.044

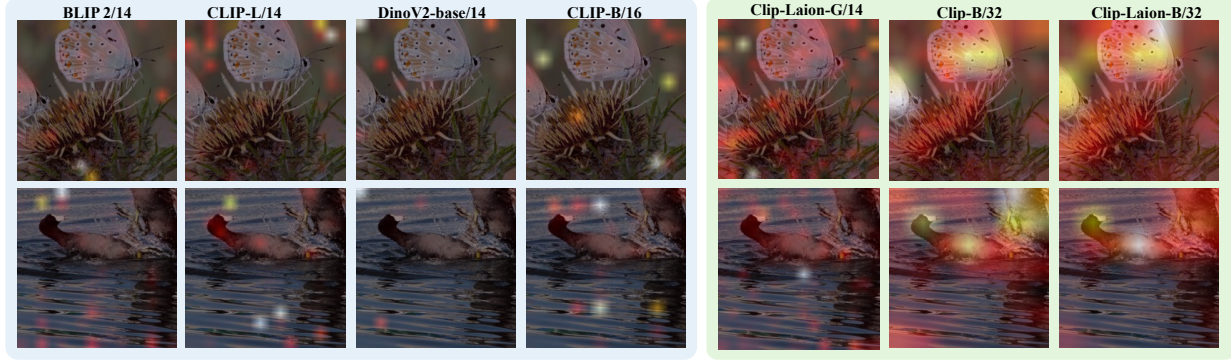


Figure 5. Comparison of two types of attention maps. Left: attention map that sparsely separates into different regions; right: attention map that focuses on the main object.

 Table 2. Comparison on open-source LVLMs (Qwen-2.5-VL and LLaVA-1.5). Higher KMR<sub>a/b/c</sub> and ASR are better.

Method	Qwen-2.5-VL				LLaVA-1.5			
	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
AttackVLM	0.12	0.04	0.00	0.01	0.11	0.03	0.00	0.07
SSA-CWA	0.36	0.25	0.04	0.38	0.29	0.17	0.04	0.34
AnyAttack	0.53	0.28	0.09	0.53	0.60	0.32	0.07	0.58
FOA-Attack	0.83	0.61	0.20	0.91	0.94	0.75	<b>0.29</b>	0.95
M-Attack	0.80	0.65	0.17	0.90	0.85	0.59	0.20	0.95
M-Attack-V2	<b>0.87</b>	<b>0.67</b>	<b>0.27</b>	<b>0.95</b>	<b>0.96</b>	<b>0.83</b>	<b>0.29</b>	<b>0.96</b>

in Tab. 3 and (ii) span diverse patch sizes to capture complementary inductive biases. A small ablation over these filtered models (see in the appendix) yields our final ensemble, *Patch Ensemble*<sup>+</sup> (PE<sup>+</sup>), comprising CLIP<sup>†</sup>-G/14, CLIP-B/16, CLIP-B/32, and CLIP<sup>†</sup>-B/32. We treat PE<sup>+</sup> as an efficient, sparse pre-selection that can be used directly or further plugged into aggregation methods. Qualitative attention maps offer an intuitive explanation: selected models consistently focus on the main object, whereas discarded ones tend to spread attention over background regions, suggesting that emphasizing core semantic content is more transferable than dispersed, model-specific patterns.

### 4.3. Evaluation Across LVLMs and Settings

**Transferability across LVLMs.** Tab. 1 illustrates the superiority of our M-Attack-V2 compared to the other black-box

LVLM attack method with Tab. 2 on open-source models. Our method outperforms others by a large margin, including M-Attack. On GPT-5 our M-Attack-V2 even achieves 100% ASR and 97% ASR on Gemini-2.5, with ASR on Claude 4.0-extended further improved by 22%, which is almost impossible for M-Attack to attack. There is also a notable improvement on the KMR, indicating that our method generates a perturbation that targets the semantics more effectively, thus more recognizable by the target black-box model. Note that these improvements are accompanied by a slight increase in the perturbation norms for  $l_1$  and  $l_2$ . Previous  $l_1$  and  $l_2$  norms are caused by insufficient optimization through near-orthogonal gradients. Our M-Attack-V2 mitigates this issue, exploring more sufficiently inside the  $l_\infty$  ball. Thus, it slightly increases the perturbation magnitude while keeping the visual appearance almost unchanged. Fig. 4 shows representative adversarial examples. Recognizing that the raw  $\ell_p$  norm may not translate into human imperceptibility, we further evaluate the human imperceptibility in user studies reported in the appendix, showing nearly identical performance between M-Attack and M-Attack-V2, which consistently outperform all other methods.

**Performance under varying budgets.** Fig. 6 compares performance under varying optimization budgets (total steps). Our method converges faster, approaching optimal results within 300 steps, whereas M-Attack requires an additional



Table 3. Comparison of embedding transferability over 1k images. MCA/ATA excluded to show standalone performance. C/D = CLIP/DinoV2. Gray denotes selected models.

Surrogate	C-L/14	C <sup>†</sup> -L/14	D-S/14	D-B/14	D-L/14	C-B/16	C <sup>†</sup> -B/16	C-B/32	C <sup>†</sup> -B/32	BLIP2	Avg/14	Avg/16	Avg/32	Avg/All
C-L/14	N/A	0.40	0.10	0.13	0.12	0.45	0.40	0.34	0.24	0.48	0.25	0.42	0.29	0.30
C <sup>†</sup> -L/14	0.44	N/A	0.24	0.24	0.21	0.55	0.57	0.37	0.33	0.61	0.35	0.56	0.35	0.39
D-S/14	0.25	0.39	N/A	0.45	0.38	0.41	0.45	0.32	0.25	0.46	0.39	0.43	0.28	0.37
D-B/14	0.29	0.36	0.33	N/A	0.51	0.37	0.39	0.31	0.23	0.47	0.39	0.38	0.27	0.36
D-L/14	0.26	0.31	0.12	0.32	N/A	0.31	0.34	0.30	0.21	0.42	0.29	0.33	0.26	0.29
C-B/16	0.44	0.43	0.21	0.18	0.13	N/A	0.53	0.37	0.27	0.51	0.32	0.53	0.32	0.34
C <sup>†</sup> -B/16	0.43	0.51	0.22	0.21	0.15	0.57	N/A	0.39	0.34	0.52	0.34	0.57	0.36	0.37
C-B/32	0.37	0.43	0.21	0.11	0.09	0.55	0.53	N/A	0.49	0.46	0.28	0.54	0.49	0.36
C <sup>†</sup> -B/32	0.31	0.49	0.27	0.18	0.12	0.53	0.61	0.58	N/A	0.50	0.31	0.57	0.58	0.40
BLIP2	0.39	0.43	0.15	0.20	0.26	0.45	0.43	0.33	0.25	N/A	0.29	0.44	0.29	0.32

Table 4. Effect of removing each component. Numbers below each value denote the change relative to the full model (first row). **X** marks the component(s) disabled.

Component			Gemini 2.5-Pro				Claude 3.7-extended			
MCA	ATA	PM	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
			0.87	0.72	0.22	0.97	0.56	0.32	0.11	0.67
X			0.85	0.70	0.21	0.92	0.52	0.35	0.08	0.66
			↓ 0.02	↓ 0.02	↓ 0.01	↓ 0.05	↓ 0.04	↑ 0.03	↓ 0.03	↓ 0.01
	X		0.85	0.68	0.21	0.93	0.55	0.22	0.10	0.62
			↓ 0.02	↓ 0.04	↓ 0.01	↓ 0.04	↓ 0.01	↓ 0.10	↓ 0.01	↓ 0.05
X	X		0.82	0.62	0.22	0.93	0.44	0.31	0.08	0.62
			↓ 0.05	↓ 0.10	-	↓ 0.04	↓ 0.12	↓ 0.01	↓ 0.03	↓ 0.05
		X	0.82	0.71	0.21	0.96	0.52	0.32	0.10	0.66
			↓ 0.05	↓ 0.01	↓ 0.01	↓ 0.01	↓ 0.04	↓ 0.00	↓ 0.01	↓ 0.01

Table 5. Results of M-Attack-V2 on the vision reasoning model.

Model	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
GPT-o3 (o3-2025-04-16)	0.91	0.71	0.23	0.98

200 steps, suggesting slower convergence. At fewer steps (100 and 200), M-Attack exhibits a notable performance drop. Meanwhile, our method maintains stable ASR and KMR<sub>b</sub> due to a more coherent optimization trajectory than M-Attack, which is more sensitive to random cropping and aggressive target transformations. Additional results on varying  $\epsilon$  is presented in the appendix.

**Robustness Against Vision-Reasoning Models.** Reasoning in text modality does not extend to alter information from the vision backbone. Instead, we further evaluate M-Attack-V2 against GPT-o3, a model enhanced with visual reasoning capabilities. As shown in Tab. 5, GPT-o3 exhibits slightly better robustness than GPT-4o. However, the limited improvement suggests that its reasoning module is not explicitly trained to detect adversarial manipulations in the image. Thus, even after reasoning, GPT-o3 remains susceptible to M-Attack-V2. The reasoning process is presented in the appendix, where it shows certain degrees of suspect in some images, and further utilize python tools for zooming.

#### 4.4. Ablation Study

Tab. 4 isolates the effect of each module beyond PE<sup>+</sup> (GPT-4o is omitted due to negligible differences). On both Gemini-2.5-Pro and Claude-3.7-extended, activating MCA or ATA alone delivers ~5% gains on average, most visi-

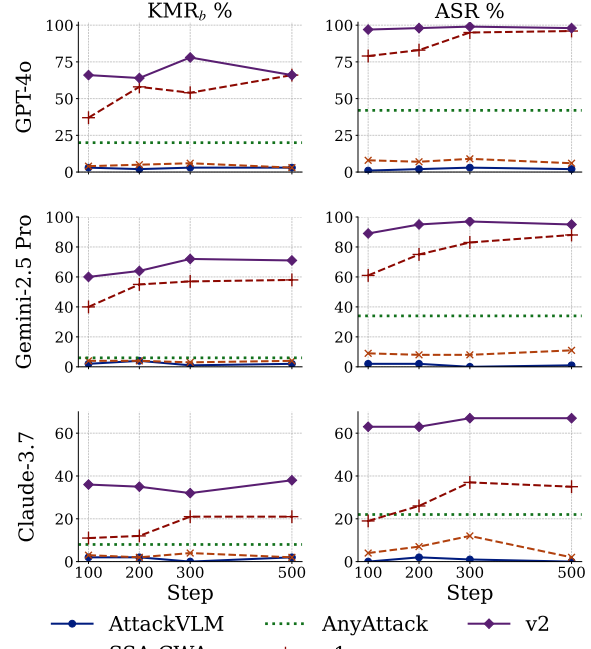


Figure 6. Comparison under different step budgets.

ble in ASR and KMR<sub>b</sub>, with consistent improvements on KMR<sub>a</sub>/KMR<sub>c</sub>. Additionally, removing PM yields only a minor drop in performance, suggesting it is complementary rather than fundamental. Overall, MCA and ATA constitute the principal mechanisms for variance reduction. At the same time, PM serves as a low-cost memory that extends the effective momentum horizon via a biased gradient, further suppressing variance and adding robustness.

## 5. Conclusion

In this work, we diagnosed M-Attack’s unstable gradients as arising from high variance and overlooked asymmetric matching, and address them with a principled gradient-denoising framework. Multi-Crop Alignment (MCA), Auxiliary Target Alignment (ATA), and Patch Momentum (PM), together with a refined surrogate ensemble (PE<sup>+</sup>), form our proposed framework, which achieves state-of-the-art transfer-based black-box attacks on LVLNs. We hope these insights help achieve more stable and transferable adversarial optimization under realistic black-box constraints.



## Impact Statement

This work strengthens transfer-based black-box attacks on large vision–language models, improving the ability to stress-test real-world multimodal systems used in assistants, search, and content generation. By revealing key instabilities in local-level matching and proposing simple gradient-denoising fixes, our methods can help researchers and practitioners build more reliable defenses, develop better robustness benchmarks, and identify failure modes before deployment. At the same time, more effective attacks can be misused to bypass safety filters, induce targeted hallucinations, or manipulate model outputs in high-stakes settings. To mitigate misuse, we have emphasized responsible disclosure, i.e., evaluate primarily on controlled benchmarks and public models, and will make full code/data/models publicly available in a way that supports reproducibility and defense research (e.g., including detection/mitigation baselines and clear guidance on safe use of our optimized generations), while avoiding instructions or configurations that lower the barrier to real-world harm.

## References

- Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: a visual language model for few-shot learning. In *International Conference on Advanced Neural Information Processing Systems*, pp. 23716–23736, 2022.
- Anthropic. Introducing claude 3.5 sonnet, 2024a. URL <https://www.anthropic.com/news/claude-3-5-sonnet>. Accessed: 2025-02-22.
- Anthropic. Claude 3.7 sonnet and claude code, 2024b. URL <https://www.anthropic.com/news/claude-3-7-sonnet>. Accessed: 2025-02-22.
- Chen, H., Shao, S., Wang, Z., Shang, Z., Chen, J., Ji, X., and Wu, X. Bootstrap generalization ability from loss landscape perspective. In *European Conference on Computer Vision*, pp. 500–517, 2022a.
- Chen, H., Zhang, Y., Dong, Y., Yang, X., Su, H., and Zhu, J. Rethinking model ensemble in transfer-based adversarial attacks. In *International Conference on Learning Representations*, 2024.
- Chen, J., Guo, H., Yi, K., Li, B., and Elhoseiny, M. Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 18030–18040, 2022b.
- Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 9185–9193, 2018.
- Dong, Y., Cheng, S., Pang, T., Su, H., and Zhu, J. Query-efficient black-box adversarial attacks guided by a transfer-based prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9536–9548, 2021.
- Dong, Y., Chen, H., Chen, J., Fang, Z., Yang, X., Zhang, Y., Tian, Y., Su, H., and Zhu, J. How robust is google’s bard to adversarial image attacks? *arXiv preprint arXiv:2309.11751*, 2023a.
- Dong, Y., Chen, H., Chen, J., Fang, Z., Yang, X., Zhang, Y., Tian, Y., Su, H., and Zhu, J. How robust is google’s bard to adversarial image attacks? *arXiv preprint arXiv:2309.11751*, 2023b.
- Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.
- Guo, Q., Pang, S., Jia, X., Liu, Y., and Guo, Q. Efficient generation of targeted and transferable adversarial examples for vision-language models via diffusion models. *IEEE Transactions on Information Forensics and Security*, 20: 1333–1348, 2024.
- Hu, X., Gan, Z., Wang, J., Yang, Z., Liu, Z., Lu, Y., and Wang, L. Scaling up vision-language pre-training for image captioning. In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 17980–17989, 2022.
- Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., and Schmidt, L. Openclip, July 2021. URL <https://doi.org/10.5281/zenodo.5143773>.
- Ilyas, A., Engstrom, L., Athalye, A., and Lin, J. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pp. 2137–2146, 2018.
- Jia, X., Gao, S., Qin, S., Pang, T., Du, C., Huang, Y., Li, X., Li, Y., Li, B., and Liu, Y. Adversarial attacks against closed-source mllms via feature optimal alignment. *arXiv preprint arXiv:2505.21494*, 2025.
- Kingma, D. P. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Xie, C., et al.

- Adversarial attacks and defences competition. In *The NIPS'17 Competition: Building Intelligent Systems*, pp. 195–231. Springer, 2018a.
- Kurakin, A., Goodfellow, I. J., and Bengio, S. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pp. 99–112. Chapman and Hall/CRC, 2018b.
- Li, H., Ellis, J. G., Zhang, L., and Chang, S.-F. Patternnet: Visual pattern mining with deep neural network. In *Proceedings of the 2018 ACM on international conference on multimedia retrieval*, pp. 291–299, 2018.
- Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900, 2022.
- Li, J., Li, D., Savarese, S., and Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.
- Li, Z., Zhao, X., Wu, D.-D., Cui, J., and Shen, Z. 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.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.
- Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *International Conference on Advanced Neural Information Processing Systems*, pp. 34892–34916, 2023.
- Liu, Y., Chen, X., Liu, C., and Song, D. Delving into transferable adversarial examples and black-box attacks. In *International Conference on Learning Representations*, 2017.
- Long, Y., Zhang, Q., Zeng, B., Gao, L., Liu, X., Zhang, J., and Song, J. Frequency domain model augmentation for adversarial attack. In *European Conference on Computer Vision*, pp. 549–566, 2022.
- Luu, D.-T., Le, V.-T., and Vo, D. M. Questioning, answering, and captioning for zero-shot detailed image caption. In *Asian Conference on Computer Vision*, pp. 242–259, 2024.
- Ma, Z., Hong, J., Gul, M. O., Gandhi, M., Gao, I., and Krishna, R. Crepe: Can vision-language foundation models reason compositionally? In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 10910–10921, 2023.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.
- OpenAI. Introducing o3 and o4-mini, April 2025. URL <https://openai.com/index/introducing-o3-and-o4-mini/>. OpenAI Blog.
- Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.
- Özdemir, Ö. and Akagündüz, E. Enhancing visual question answering through question-driven image captions as prompts. In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, pp. 1562–1571, 2024.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.
- Salaberria, A., Azkune, G., de Lacalle, O. L., Soroa, A., and Agirre, E. Image captioning for effective use of language models in knowledge-based visual question answering. *Expert Systems with Applications*, 212:118669, 2023.
- Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35: 25278–25294, 2022.
- Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.
- Tschannen, M., Kumar, M., Steiner, A., Zhai, X., Houlsby, N., and Beyer, L. Image captioners are scalable vision learners too. In *International Conference on Advanced Neural Information Processing Systems*, pp. 46830–46855, 2023.
- Wang, T., Li, F., Zhu, L., Li, J., Zhang, Z., and Shen, H. T. Cross-modal retrieval: a systematic review of methods and future directions. *arXiv preprint arXiv:2308.14263*, 2024.
- Wu, J., Zhong, M., Xing, S., Lai, Z., Liu, Z., Chen, Z., Wang, W., Zhu, X., Lu, L., Lu, T., et al. Visionllm v2: An end-to-end generalist multimodal large language model for hundreds of vision-language tasks. In *International*

*Conference on Advanced Neural Information Processing Systems*, pp. 69925–69975, 2025.

Yang, J., Shi, R., and Ni, B. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 191–195, 2021.

Zhang, J., Ye, J., Ma, X., Li, Y., Yang, Y., Sang, J., and Yeung, D.-Y. Anyattack: Towards large-scale self-supervised generation of targeted adversarial examples for vision-language models. *arXiv preprint arXiv:2410.05346*, 2024.

Zhao, Y., Pang, T., Du, C., Yang, X., Li, C., Cheung, N.-M. M., and Lin, M. On evaluating adversarial robustness of large vision-language models. In *International Conference on Advanced Neural Information Processing Systems*, pp. 54111–54138, 2023.



# Appendix

## Contents

<b>A</b>	<b>Additional Details for Theoretical Analysis</b>	<b>13</b>
A.1	Proof for Theorem 1 . . . . .	13
A.2	Proof of Theorem 2 . . . . .	13
A.3	Justification for Assumptions . . . . .	14
<b>B</b>	<b>More Details on Our Algorithm</b>	<b>15</b>
<b>C</b>	<b>More Details of Experimental Setup</b>	<b>15</b>
<b>D</b>	<b>Full Process of Surrogate Model Selection</b>	<b>15</b>
<b>E</b>	<b>More Ablation Study</b>	<b>16</b>
E.1	Ablation Study for Step Size . . . . .	16
<b>F</b>	<b>Success and Failure Examples</b>	<b>17</b>
F.1	Ablation Study on MCA and ATA Hyperparameters . . . . .	17
<b>G</b>	<b>Additional Results</b>	<b>18</b>
G.1	Additional Results on 1K image . . . . .	18
G.2	Additional Results on FGSM framework . . . . .	18
G.3	Additional Results on Other GPT-5 Reasoning Modes . . . . .	18
G.4	Cross-Domain Evaluation on Medical and Overhead Imagery . . . . .	19
G.5	Robustness to input-preprocessing defenses . . . . .	19
G.6	Human Perceptual Study . . . . .	19
<b>H</b>	<b>Computational Cost and Runtime Analysis</b>	<b>20</b>
<b>I</b>	<b>Additional Visualization</b>	<b>22</b>
I.1	Visualization of Adversarial Samples . . . . .	22
I.2	Visualization of Reasoning Models . . . . .	23

## A. Additional Details for Theoretical Analysis

### A.1. Proof for Theorem 1

This section provides detailed proof of the upper bound in Equ. (5). For variance, we have

$$\begin{aligned}
 \text{Var}(\hat{g}_K) &:= \mathbb{E} \|\hat{g}_K - \mu\|^2 \\
 &= \mathbb{E} \left\| \frac{1}{K} \sum_{k=1}^K (g_k - \mu) \right\|^2 \\
 &= \frac{1}{K^2} \sum_{k=1}^K \sum_{\ell=1}^K \mathbb{E}[(g_k - \mu)^\top (g_\ell - \mu)] \\
 &= \frac{1}{K^2} \left( \underbrace{\sum_{k=1}^K \mathbb{E} \|g_k - \mu\|_2^2}_{K\sigma^2} + \underbrace{2 \sum_{1 \leq k < \ell \leq K} \mathbb{E}[\langle g_k - \mu, g_\ell - \mu \rangle]}_{\text{cross terms}} \right) \tag{9}
 \end{aligned}$$

The diagonal part is reduced to the mean. We now provide an upper bound for the cross terms. Recall  $p_{k\ell} = \frac{\langle g_k - \mu, g_\ell - \mu \rangle}{\|g_k - \mu\|^2 \|g_\ell - \mu\|^2}$ , we have

$$\mathbb{E}[\langle g_k - \mu, g_\ell - \mu \rangle] = \mathbb{E}[\rho_{k\ell} \|g_k - \mu\|_2 \|g_\ell - \mu\|_2]. \tag{10}$$

Since all crops share the same marginal distribution, i.e.  $\mathbb{E} \|g_k - \mu\|_2 = \mathbb{E} \|g_\ell - \mu\|_2 = \sigma$ , applying the Cauchy-Schwarz inequality to Equ. (10) yields

$$\begin{aligned}
 \mathbb{E}[\langle g_k - \mu, g_\ell - \mu \rangle] &\leq \mathbb{E}[\rho_{k\ell}] \sqrt{\mathbb{E} \|g_k - \mu\|_2^2} \sqrt{\mathbb{E} \|g_\ell - \mu\|_2^2} \\
 &= \bar{\rho} \sigma^2, \tag{11}
 \end{aligned}$$

where  $\bar{\rho}$  is  $\mathbb{E}[\rho_{k\ell}]$ ,  $k \neq \ell$ . Plugging this into the double sum term yields

$$\sum_{1 \leq k < \ell \leq K} \mathbb{E}[\langle g_k - \mu, g_\ell - \mu \rangle] \leq \frac{K(K-1)}{2} \bar{\rho} \sigma^2. \tag{12}$$

The  $\frac{K(K-1)}{2}$  appears since there are total  $\frac{K(K-1)}{2}$  terms for  $\sum_{k < \ell}$ . Thus substituting Equ. (12) back to the cross item part in the Equ. (9) yields

$$\begin{aligned}
 \text{Var}(\hat{g}_K) &\leq \frac{1}{K^2} (K\sigma^2 + K(K-1)\bar{\rho}\sigma^2) \\
 &= \frac{1}{K} \sigma^2 + \frac{K-1}{K} \bar{\rho} \sigma^2 \tag{13}
 \end{aligned}$$

Therefore, we have the upper bound provided in the Sec. 3.3.

### A.2. Proof of Theorem 2

We begin with the drift analysis for M-Attack:

$$\begin{aligned}
 \Delta_{\text{drift}}(\mathcal{T}; \mathbf{X}_{\text{tar}}) &= \mathbb{E}_{\mathcal{T} \sim D_\alpha} [\|f(\mathcal{T}(\mathbf{X}_{\text{tar}})) - f(\mathbf{X}_{\text{tar}})\|] \\
 &\leq L \cdot \mathbb{E}_{\mathcal{T} \sim D_\alpha} [\|\mathcal{T}(\mathbf{X}_{\text{tar}}) - \mathbf{X}_{\text{tar}}\|] \\
 &\leq L\alpha. \tag{14}
 \end{aligned}$$

**Algorithm 2:** M-Attack-V2 (Adam variant)

---

**Input** : clean image  $\mathbf{X}_{\text{clean}}$ ; primary target  $\mathbf{X}_{\text{tar}}$ ; **auxiliary set**  $\mathcal{A} = \{\mathbf{X}_{\text{aux}}^{(p)}\}_{p=1}^P$ ; **patch ensemble**<sup>+</sup>  $\Phi^+ = \{\phi_j\}_{j=1}^m$ ; iterations  $n$ , step size  $\alpha$ , perturbation budget  $\epsilon$ ; Adam  $\beta_1, \beta_2, \eta$ ; number of crops  $K$ , auxiliary weight  $\lambda$

**Output** : adversarial image  $\mathbf{X}_{\text{adv}}$

---

```

1  $\mathbf{X}_{\text{adv}} \leftarrow \mathbf{X}_{\text{clean}}, m \leftarrow 0, v \leftarrow 0$ 
2 for  $i = 1$  to  $n$  do
3   Draw  $K$  transforms  $\{\mathcal{T}_k\}_{k=1}^K \sim \mathcal{D}$ 
4    $g \leftarrow 0$  // accumulate over crops
5   for  $k = 1$  to  $K$  do // — crop loop —
6     Draw  $\{\tilde{\mathcal{T}}_p\}_{p=0}^P \sim \tilde{\mathcal{D}}$ 
7     for  $j = 1$  to  $m$  do
8        $y_0 = f(\tilde{\mathcal{T}}_0(\mathbf{X}_{\text{tar}}))$ 
9        $y_p = f(\tilde{\mathcal{T}}_p(\mathbf{X}_{\text{aux}}^{(p)})), p = 1, \dots, P$ 
10       $\hat{\mathcal{L}}_k = \mathcal{L}(f_{\phi_j}(\mathcal{T}_k(\mathbf{X}_{\text{adv}})), y_0) + \frac{\lambda}{P} \sum_{p=1}^P \mathcal{L}(f_{\phi_j}(\mathcal{T}_k(\mathbf{X}_{\text{adv}})), y_p)$ 
11       $g \leftarrow g + \frac{1}{Km} \nabla_{\mathbf{X}_{\text{adv}}} \hat{\mathcal{L}}_k$ 
12    end
13  end
14  // — Adam update —
15   $m \leftarrow \beta_1 m + (1 - \beta_1)g$ 
16   $v \leftarrow \beta_2 v + (1 - \beta_2)g^{\odot 2}$ 
17   $\hat{m} \leftarrow m / (1 - \beta_1^i)$ 
18   $\hat{v} \leftarrow v / (1 - \beta_2^i)$ 
19   $\mathbf{X}_{\text{adv}} \leftarrow \text{clip}_{\mathbf{X}_{\text{clean}}, \epsilon}(\mathbf{X}_{\text{adv}} + \alpha \hat{m} / (\sqrt{\hat{v}} + \eta))$ 
20 end
21 return  $\mathbf{X}_{\text{adv}}$ 

```

---

Next, we analyze the drift for M-Attack-V2 using the triangle inequality and the above assumptions:

$$\begin{aligned}
 \Delta_{\text{drift}}(\tilde{\mathcal{T}}; \mathbf{X}_{\text{aux}}^{(p)}) &= \mathbb{E}_{\tilde{\mathcal{T}} \sim D_{\tilde{\alpha}}} [\|f(\tilde{\mathcal{T}}(\mathbf{X}_{\text{aux}}^{(p)})) - f(\mathbf{X}_{\text{tar}})\|] \\
 &\leq \mathbb{E}_{\tilde{\mathcal{T}}} [\|f(\tilde{\mathcal{T}}(\mathbf{X}_{\text{aux}}^{(p)})) - f(\mathbf{X}_{\text{aux}}^{(p)})\| + \\
 &\quad \|f(\mathbf{X}_{\text{aux}}^{(p)}) - f(\mathbf{X}_{\text{tar}})\|] \\
 &= \mathbb{E}_{\tilde{\mathcal{T}}} [\|f(\tilde{\mathcal{T}}(\mathbf{X}_{\text{aux}}^{(p)})) - f(\mathbf{X}_{\text{aux}}^{(p)})\|] + \\
 &\quad \mathbb{E} [\|f(\mathbf{X}_{\text{aux}}^{(p)}) - f(\mathbf{X}_{\text{tar}})\|] \\
 &\leq L \mathbb{E}_{\tilde{\mathcal{T}}} [\|\tilde{\mathcal{T}}(\mathbf{X}_{\text{aux}}^{(p)}) - \mathbf{X}_{\text{aux}}^{(p)}\|] + \delta \\
 &\leq L\tilde{\alpha} + \delta
 \end{aligned}$$

Thus, we have completed the proof of Theorem 3.5.

### A.3. Justification for Assumptions

Assumption 3.3 is derived from the retrieval mechanism for auxiliary data. Specifically,  $X_{\text{aux}}^{(p)}$  represents the  $p$ -th closest embedding to the target  $X_{\text{tar}}$  from a database  $\mathcal{D}$ , defined explicitly by:

$$\mathbf{X}_{\text{aux}}^{(p)} \in \arg \text{top}_P \left\{ \mathbf{X} \in \mathcal{D} : \frac{f(\mathbf{X})^\top f(\mathbf{X}_{\text{tar}})}{\|f(\mathbf{X})\| \|f(\mathbf{X}_{\text{tar}})\|} \right\}, \quad (15)$$

where  $\text{top}_P$  denotes selecting the top- $P$  nearest neighbors according to cosine similarity. Given that embeddings  $f(\mathbf{X})$  are typically normalized, semantic closeness naturally bounds the expected distance between  $f(\mathbf{X}_{\text{aux}}^{(p)})$  and  $f(\mathbf{X}_{\text{tar}})$  by  $\delta$ , thus



**Algorithm 3: M-Attack-V2 (MI-FGSM variant)**


---

**Input** : clean image  $\mathbf{X}_{\text{clean}}$ ; primary target  $\mathbf{X}_{\text{tar}}$ ; **auxiliary set**  $\mathcal{A} = \{\mathbf{X}_{\text{aux}}^{(p)}\}_{p=1}^P$ ; **patch ensemble**  $\Phi^+ = \{\phi_j\}_{j=1}^m$ ; iterations  $n$ , step size  $\alpha$ , perturbation budget  $\epsilon$ ; momentum decay  $\gamma$ ; number of crops  $K$ , auxiliary weight  $\lambda$

**Output** : adversarial image  $\mathbf{X}_{\text{adv}}$

---

```

1  $\mathbf{X}_{\text{adv}} \leftarrow \mathbf{X}_{\text{clean}}, \mu \leftarrow 0$ 
2 for  $i = 1$  to  $n$  do
3     Draw  $K$  transforms  $\{\mathcal{T}_k\}_{k=1}^K \sim \mathcal{D}$ 
4      $g \leftarrow 0$ 
5     for  $k = 1$  to  $K$  do
6         Draw  $\{\tilde{\mathcal{T}}_p\}_{p=0}^P \sim \tilde{\mathcal{D}}$ 
7         for  $j = 1$  to  $m$  do
8              $y_0 = f(\tilde{\mathcal{T}}_0(\mathbf{X}_{\text{tar}}))$ 
9              $y_p = f(\tilde{\mathcal{T}}_p(\mathbf{X}_{\text{aux}}^{(p)})), p = 1, \dots, P$ 
10             $\hat{\mathcal{L}}_k = \mathcal{L}(f_{\phi_j}(\mathcal{T}_k(\mathbf{X}_{\text{adv}})), y_0) + \frac{\lambda}{P} \sum_{p=1}^P \mathcal{L}(f_{\phi_j}(\mathcal{T}_k(\mathbf{X}_{\text{adv}})), y_p)$ 
11             $g \leftarrow g + \frac{1}{Km} \nabla_{\mathbf{X}_{\text{adv}}} \hat{\mathcal{L}}_k$ 
12        end
13    end
14    // — MI-FGSM update —
15     $\mu \leftarrow \gamma \mu + \frac{g}{\|g\|_1}$ 
16     $\mathbf{X}_{\text{adv}} \leftarrow \text{clip}_{\mathbf{X}_{\text{clean}}, \epsilon}(\mathbf{X}_{\text{adv}} + \alpha \text{sign}(\mu))$ 
17 end
18 return  $\mathbf{X}_{\text{adv}}$ 

```

---

validating Assumption 3.3. In such a case, to estimate  $\delta$ , use  $2(1 - f(\mathbf{X}_{\text{aux}}^{(P)})^\top f(\mathbf{X}_{\text{tar}}))$

## B. More Details on Our Algorithm

Alg. 2 and Alg. 3 provide detailed update rule of line 13 in Alg. 1. Fig. 7 provides a comparison between the entire procedure of M-Attack and our M-Attack-V2 under the local-matching framework. Notably, M-Attack utilizes a radical crop on the target image, risking unrelated or broken semantics for the source image to align. Our ATA anchors more points inside the semantic manifold (blue), and provides a mild transformation to provide a coherence sampling from the target semantic manifold.

## C. More Details of Experimental Setup

The experiment’s seed is 2023. It is conducted on a Linux platform (Ubuntu 22.04) with 6 NVIDIA RTX 4090 GPUs. The temperatures of all LLMs are set to 0. The threshold of the ASR is set to 0.3, following M-Attack. Tab. 9 provides a map from model names in this paper to their identifiers in HuggingFace. We use GPT-5-thinking-low (setting reasoning effort to low in the API) for all results in the main paper, with results on other reasoning budgets presented in the Appx. G.3

## D. Full Process of Surrogate Model Selection

This section details the process of selecting our final ensemble,  $\text{PE}^+$ . Exhaustively testing all model combinations is computationally infeasible, so we employ a heuristic-driven approach. We begin by excluding DiNO-large and BLIP2 due to their poor transferability, as shown in Tab. 3. Our initial experiments focus on evaluating the effectiveness of homogeneous ensembles—comprising models with the same patch size—versus mixed patch size ensembles. Specifically, we construct five ensembles: (1) patch-14 CLIP (CLIP-L/14, CLIP<sup>†</sup>-G/14), (2) patch-14 DiNOv2 (Dino-base, Dino-large), (3) patch-16 CLIP (CLIP-B/16, CLIP<sup>†</sup>-B/16), and (4) patch-32 CLIP (CLIP-B/32, CLIP<sup>†</sup>-B/32). Results are presented in Tab. 7. These results reveal that the patch-32 CLIP ensemble performs best on Claude 3.7, while GPT-4o and Gemini 2.5 Pro favor models with patch sizes 14 and 16. This supports the findings in Sec. 4.2: although using a fixed patch size can mitigate architectural bias, it still inherits the intrinsic bias of the patch size itself.

To address this, we adopt a cross-patch size strategy. Starting from the patch-32 CLIP ensemble, due to its strong performance

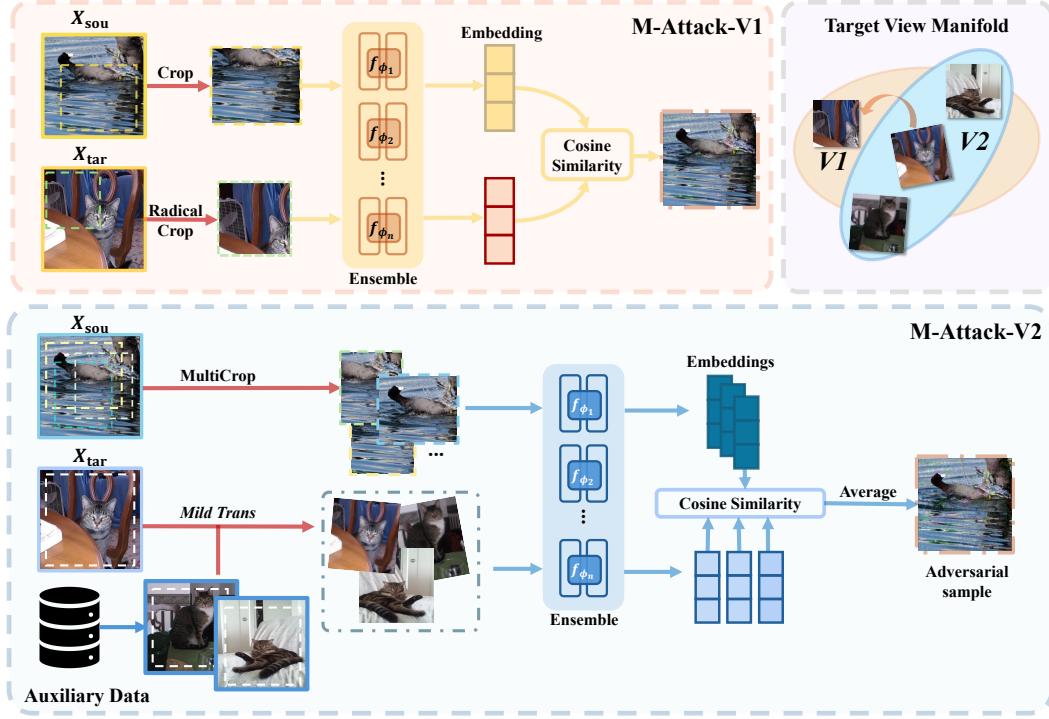
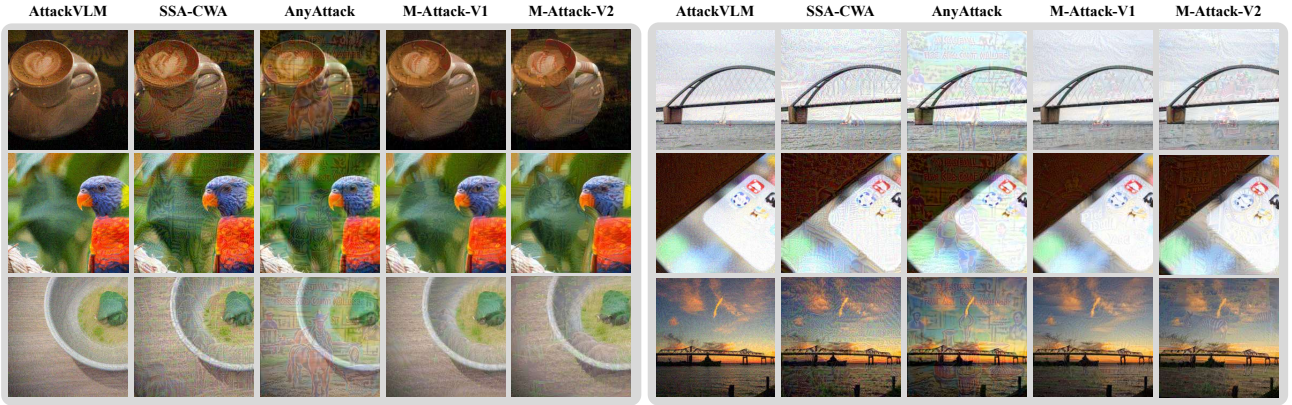


Figure 7. Comparison of one step between M-Attack and M-Attack-V2.


 Figure 8. Visualization of adversarial samples under  $\epsilon = 16$ .

on Claude and consistent transferability across patch-16 and patch-32 models. We incrementally incorporate one model each from patch sizes 14 and 16. We evaluate various combinations, with results summarized in Tab. 8. The resulting ensemble,  $PE^+$ , achieves the most balanced performance, ranking first on 7 metrics and a close second on 3 others, across 12 evaluation metrics.

## E. More Ablation Study

### E.1. Ablation Study for Step Size

This section provides an ablation study for the step size parameter  $\alpha$  to view its impact on the performance. Overall, selecting  $\alpha \in [0.5, 1.0]$  provides better performance for SSA-CWA, M-Attack. Our M-Attack-V2 prefer stepsize at 1.275, since it adopts ADAM as optimizer.

Table 6. Ablation study on the impact of perturbation budget ( $\epsilon$ ).

$\epsilon$	Method	GPT-4o				Claude 3.7-thinking				Gemini 2.5-Pro				Imperceptibility	
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	$\ell_1 \downarrow$	$\ell_2 \downarrow$
4	AttackVLM (Zhao et al., 2023)	0.08	0.04	0.00	0.02	0.04	0.01	0.00	0.00	0.10	0.04	0.00	0.01	0.010	0.011
	SSA-CWA (Dong et al., 2023a)	0.05	0.03	0.00	0.03	0.04	0.01	0.00	0.02	0.04	0.01	0.00	0.04	0.015	0.015
	AnyAttack (Zhang et al., 2024)	0.07	0.02	0.00	0.05	0.05	<b>0.05</b>	<b>0.02</b>	<b>0.06</b>	0.05	0.02	0.00	0.10	0.014	0.015
	M-Attack (Li et al., 2025)	0.30	0.16	0.03	0.26	0.06	0.01	0.00	0.01	0.24	0.14	0.02	0.15	<b>0.009</b>	<b>0.010</b>
	M-Attack-V2 (Ours)	<b>0.59</b>	<b>0.34</b>	<b>0.10</b>	<b>0.58</b>	<b>0.06</b>	0.02	0.00	0.02	<b>0.48</b>	<b>0.33</b>	<b>0.07</b>	<b>0.38</b>	0.012	0.013
8	AttackVLM (Zhao et al., 2023)	0.08	0.02	0.00	0.01	0.04	0.02	0.00	0.01	0.07	0.01	0.00	0.01	0.020	0.022
	SSA-CWA (Dong et al., 2023a)	0.06	0.02	0.00	0.04	0.04	0.02	0.00	0.02	0.02	0.00	0.00	0.05	0.030	0.030
	AnyAttack (Zhang et al., 2024)	0.17	0.06	0.00	0.13	0.07	0.07	0.02	0.05	0.12	0.04	0.00	0.13	0.028	0.029
	M-Attack (Li et al., 2025)	0.74	0.50	0.12	0.82	0.12	0.06	0.00	0.09	0.62	0.34	0.08	0.48	<b>0.017</b>	<b>0.020</b>
	M-Attack-V2 (Ours)	<b>0.87</b>	<b>0.69</b>	<b>0.20</b>	<b>0.93</b>	<b>0.23</b>	<b>0.14</b>	<b>0.02</b>	<b>0.22</b>	<b>0.72</b>	<b>0.49</b>	<b>0.21</b>	<b>0.77</b>	0.023	0.023
16	AttackVLM (Zhao et al., 2023)	0.08	0.02	0.00	0.02	0.01	0.00	0.00	0.01	0.03	0.01	0.00	0.00	0.036	0.041
	SSA-CWA (Dong et al., 2023a)	0.11	0.06	0.00	0.09	0.06	0.04	0.01	0.12	0.05	0.03	0.01	0.08	0.059	0.060
	AnyAttack (Zhang et al., 2024)	0.44	0.20	0.04	0.42	0.19	0.08	0.01	0.22	0.35	0.06	0.01	0.34	0.048	0.052
	M-Attack (Li et al., 2025)	0.82	0.54	0.13	0.95	0.31	0.21	0.04	0.37	0.81	0.57	0.15	0.83	<b>0.030</b>	<b>0.036</b>
	M-Attack-V2 (Ours)	<b>0.91</b>	<b>0.78</b>	<b>0.40</b>	<b>0.99</b>	<b>0.56</b>	<b>0.32</b>	<b>0.11</b>	<b>0.67</b>	<b>0.87</b>	<b>0.72</b>	<b>0.22</b>	<b>0.97</b>	0.038	0.044

Table 7. Ablation on two-model surrogate sets. Bold numbers are the best in each column; underlined numbers are the second-best.

Variant	Surrogate Set (2 models)	GPT-4o				Claude 3.7-extended				Gemini 2.5-Pro			
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
Pair <sub>1</sub>	Dino-B, Dino-S	<u>0.84</u>	0.57	0.15	0.91	0.09	0.04	0.00	0.05	<b>0.84</b>	0.53	0.11	0.81
Pair <sub>2</sub>	L16, B/16	<b>0.86</b>	<b>0.69</b>	<u>0.21</u>	<b>0.96</b>	<u>0.16</u>	<u>0.10</u>	<u>0.01</u>	<u>0.16</u>	<b>0.84</b>	<u>0.59</u>	<u>0.15</u>	<u>0.91</u>
Pair <sub>3</sub>	L32, B/32	0.76	0.52	0.13	0.79	<b>0.46</b>	<b>0.29</b>	<b>0.06</b>	<b>0.70</b>	0.58	0.37	0.07	0.59
Pair <sub>4</sub>	G/14, L14	<b>0.86</b>	<u>0.61</u>	<b>0.24</b>	<u>0.94</u>	0.07	0.02	0.00	0.06	<u>0.82</u>	<b>0.64</b>	<b>0.23</b>	<b>0.92</b>

## F. Success and Failure Examples

Fig. 9 illustrates typical adversarial samples for successful and failed attacks of M-Attack-V2 across GPT/Claude and Gemini. In failure cases, the adversarial perturbations show weaker semantic correlation with the target label, whereas in successful cases, rough shapes of the target (such as trees or animals) can be clearly identified. We also observe that shared successful target images tend to appear neater and more centralized.

### F.1. Ablation Study on MCA and ATA Hyperparameters

Fig. 10(left) shows transferability peaks around  $K = 10 \sim 20$ , beyond which increased stability reduces beneficial noise regularization. Fig. 10(right) demonstrates larger  $\lambda$  boosts diversity by aligning semantics closer to auxiliary data but risks impairing semantic accuracy (as measured by KMR). Fig. 11(a,b) indicates minor impacts from  $P$  and momentum coefficient  $\beta$ ; setting  $P = 2$  optimizes performance and efficiency, and the default  $\beta = 0.9$  consistently yields robust results.

 Table 8. Ablation on surrogate-set selection. Each row swaps one model in or out of a four-model ensemble. The fully grey PE<sup>+</sup> line is our final patch-diverse surrogate set (CLIP<sup>+</sup>-G/14, CLIP-B/16, CLIP-B/32, CLIP<sup>+</sup>-B/32). Bold numbers denote the best score in each metric column across all variants, underline denote second best with neglectable gap of 0.01

Variant	Surrogate Set	GPT-4o				Claude 3.7-extended				Gemini 2.5-Pro			
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
PE <sub>1</sub>	B/16, B/32, L32, L16	0.87	0.65	0.26	<b>0.99</b>	0.54	0.32	0.07	0.68	0.80	0.57	0.16	0.90
PE <sub>2</sub>	Dino-B, B/32, L32, G/14	0.87	0.69	0.28	0.97	0.56	0.37	0.09	0.65	<b>0.88</b>	0.71	0.22	0.93
PE <sub>3</sub>	L16, B/32, L32, G/14	0.85	0.65	0.23	<b>0.99</b>	<b>0.57</b>	<u>0.40</u>	0.09	<b>0.73</b>	0.84	0.61	0.19	0.93
PE <sub>4</sub>	B/16, B/32, L32, Dino-B	0.89	0.67	0.19	0.98	0.55	<b>0.41</b>	0.07	0.63	0.87	0.67	<b>0.23</b>	0.96
PE <sub>5</sub>	B/16, B/32, L32, Dino-S	0.90	0.72	0.25	0.97	0.48	0.33	0.08	0.59	0.83	0.63	0.17	0.90
<b>PE<sup>+</sup> (Ours)</b>	B/16, B/32, L32, G/14	<b>0.91</b>	<b>0.78</b>	<b>0.40</b>	<b>0.99</b>	<u>0.56</u>	0.32	<b>0.11</b>	0.67	<u>0.87</u>	<b>0.72</b>	<u>0.22</u>	<b>0.97</b>



Table 9. Surrogate models and their corresponding HuggingFace identifier in our main paper.

Surrogate (paper notation)	Implementation (HuggingFace identifier)
CLIP <sup>†</sup> -B/32 (Ilharco et al., 2021; Schuhmann et al., 2022)	laion/CLIP-ViT-B-32-laion2B-s34B-b79K
CLIP <sup>†</sup> -H/14 (Ilharco et al., 2021; Schuhmann et al., 2022)	laion/CLIP-ViT-H-14-laion2B-s32B-b79K
CLIP-L/14 (Radford et al., 2021)	openai/clip-vit-large-patch14
CLIP <sup>†</sup> -B/16 (Ilharco et al., 2021; Schuhmann et al., 2022)	laion/CLIP-ViT-B-16-laion2B-s34B-b88K
CLIP <sup>†</sup> -BG/14 (Ilharco et al., 2021; Schuhmann et al., 2022)	laion/CLIP-ViT-bigG-14-laion2B-39B-b160k
Dino-Small (Oquab et al., 2023)	facebook/dinov2-small
Dino-Base (Oquab et al., 2023)	facebook/dinov2-base
Dino-Large (Oquab et al., 2023)	facebook/dinov2-large
BLIP-2 (2.7 B) (Li et al., 2023)	Salesforce/blip2-opt-2.7b

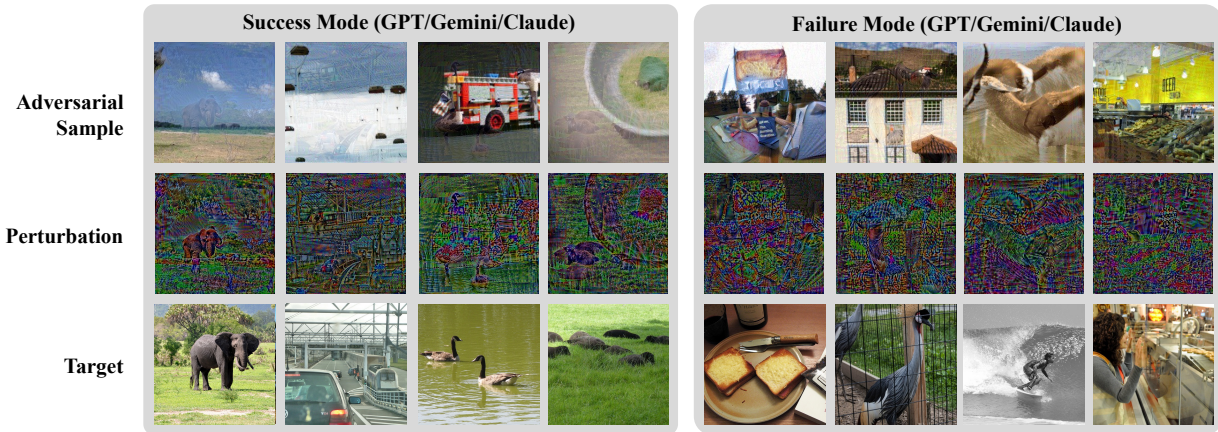


Figure 9. Success and failure modes of M-Attack-V2 shared by GPT 5/Claude 4.0-extended/Gemini 2.5-Pro.

## G. Additional Results

### G.1. Additional Results on 1K image

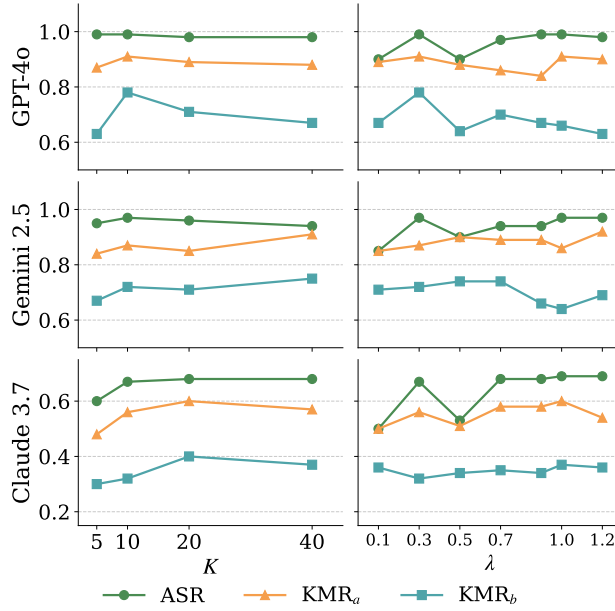
We compare M-Attack and M-Attack-V2 across 1K images to improve statistical stability. We changed the threshold to multiple values since no additional keywords were added to the 900 images, thereby replacing the KMR with ASR at different matching levels. Our M-Attack-V2 consistently outperforms M-Attack, demonstrating the superiority of our proposed strategy.

### G.2. Additional Results on FGSM framework

We provide the results of the I-FGSM (Kurakin et al., 2018b) and MI-FGSM (Dong et al., 2018) under our M-Attack framework as complementary, presented in Tab. 12. Results show that even under the FGSM framework, where the patchy gradient matter is smoothed by assigning  $\text{sign}(\nabla \mathcal{L})$ , M-Attack-V2 still benefit from momentum. Moreover, MI-FGSM still provides results comparable to those of the ADAM version. However, using PGD framework with ADAM optimizer is generally the better choice to unleash the potential of black-box attack fully since it can better explore in the space while also reducing scale issue with second-order momentum.

### G.3. Additional Results on Other GPT-5 Reasoning Modes

GPT-5 provides four reasoning modes: *minimum*, *low*, *medium*, and *high*. While the main paper presents results using GPT-5-thinking-low, additional experiments on other reasoning modes are summarized in Tab. 13. Our proposed M-Attack-V2 consistently achieves superior performance across all modes. Interestingly, providing additional thinking budget generally enhances model robustness, evidenced by a reduction in ASR and KMR. However, this improvement is not strictly monotonic: ASR first decreases from 100% (*low*) to 96% (*medium*) before slightly rebounding to 99% (*high*). A similar non-monotonic trend can also be observed elsewhere in the table.


 Figure 10. ASR and  $KMR_a/KMR_b$  vs. different  $K$  and  $\lambda$ .

#### G.4. Cross-Domain Evaluation on Medical and Overhead Imagery

Beyond the general-domain datasets, we further probe transferability to domains that are notoriously challenging for closed-source VLMs: chest X-rays and overhead remote sensing. Concretely, we augment the *NIPS 2017 adversarial competition* evaluation with images from *ChestMNIST*, from *MedMNIST* (Yang et al., 2021) and *PatternNet* (Li et al., 2018). We keep the target set unchanged and reuse the same attack budget and optimization hyper-parameters as in the main experiments. These domains are non-photographic and typically elicit generic captions from off-the-shelf VLMs, making them a stringent test of cross-domain transfer.

We report  $KMR_a/KMR_b/KMR_c$  and ASR (higher is better) on GPT-4o, Claude 3.7, and Gemini 2.5 in Tables 14 and 18. Across both datasets, M-Attack-V2 consistently surpasses M-Attack and prior baselines. On *PatternNet*, M-Attack-V2 improves Claude 3.7 ASR from 0.48 to 0.73 (+0.25) and raises GPT-4o  $KMR_{a/b/c}$  to 0.83/0.71/0.24. On *ChestMNIST*, the gains are even larger on Claude 3.7 (ASR 0.31  $\rightarrow$  0.83, +0.52), while M-Attack-V2 also achieves the highest  $KMR_{a/b/c}$  on Gemini 2.5 (0.89/0.76/0.33). The only exception is ChestMNIST ASR on Gemini 2.5, where M-Attack is marginally higher (0.96 vs. 0.95), despite M-Attack-V2 yielding stronger keyword-match rates.

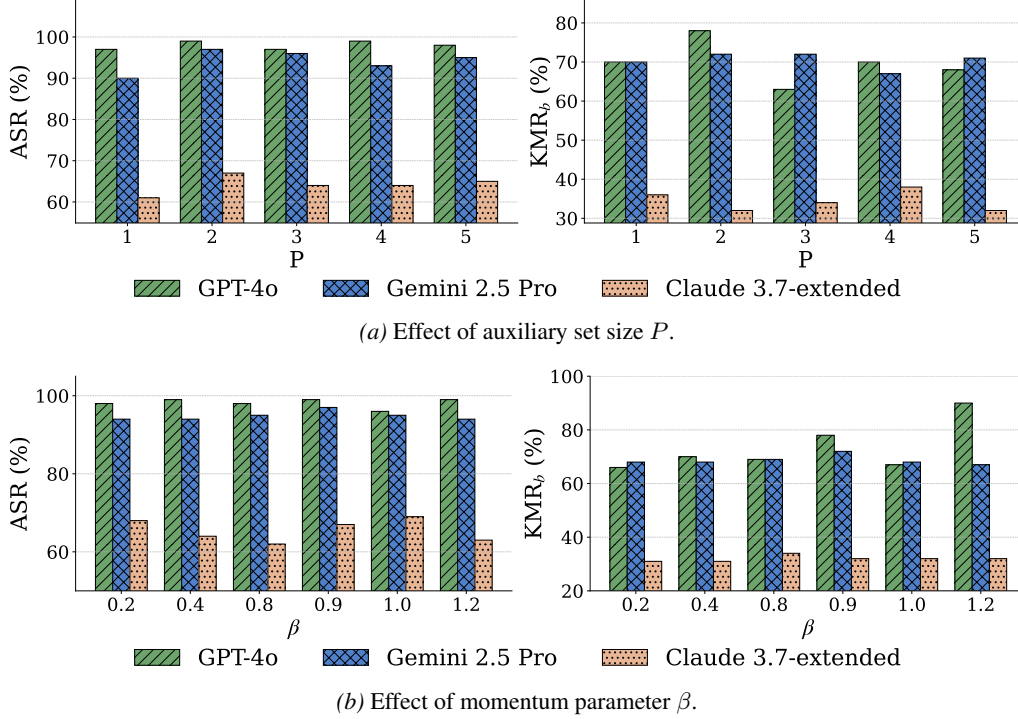
#### G.5. Robustness to input-preprocessing defenses

We evaluate two input-preprocessing defenses—JPEG recompression (quality  $Q=75$ ) and diffusion-based purification (DiffPure) with denoising budgets  $t=25$  and  $t=75$ . As summarized in Table 17, the JPEG results show that M-Attack-V2 remains strong while prior attacks substantially degrade, suggesting resilience to quantization and mild photometric shifts. DiffPure reduces success rates for all methods; however, M-Attack-V2 preserves a clear margin at  $t=25$  and remains the most effective even under the aggressive  $t=75$  setting, where purification approaches image regeneration.

#### G.6. Human Perceptual Study

To evaluate the perceptual stealth of the perturbations beyond static metrics such as the  $\ell_p$  norm, we conducted two user studies comparing adversarial samples from M-Attack-V2 and several baseline attacks against clean images.

**M-Attack-V2 against Clean Images.** Participants were shown 50 images (25 perturbed by M-Attack-V2 and 25 unmodified) in a random order and asked to label each image as “perturbed” or “clean”. Adversarial images were generated with


 Figure 11. Ablation study on auxiliary set size  $P$  and momentum parameter  $\beta$ .

a perturbation budget  $\epsilon = 16$  or  $\epsilon = 8$ . Results, averaged over 10 distinct user groups, are summarized in Table 15. On average, only 42% of M-Attack-V2 adversarial images are correctly identified as corrupted, meaning that 58% of them pass the human check even under explicit supervision. We further repeat this study at a smaller perturbation budget  $\epsilon = 8$ , since M-Attack-V2 still exceeds other methods by a large margin even under  $\epsilon = 8$ . As shown in Table 15, the proportion of adversarial images identified by users drops from 42% to 27.4% when reducing  $\epsilon$  from 16 to 8, while the confusion between adversarial and real images also increases, and the error of identifying clean images also increases. These results highlight the potential threat of M-Attack-V2 in real-world scenarios where human inspection is relied upon.

**Comparison across Attack Methods.** In a second study, each participant (10 in total) was shown 40 images: 10 adversarial examples from each of AnyAttack, SSA-CWA, M-Attack-V1, and M-Attack-V2 (again at  $\epsilon = 16$ ). Participants were told that exactly half of the images were corrupted and asked to select the 20 images they believed were mostly perturbed. This protocol directly compares the perceptual stealthiness of different attacks. As reported in Table 16, AnyAttack is the most easily detected method, with 84% of its images identified as perturbed. SSA-CWA is somewhat less detectable (54%), while M-Attack-V1 and M-Attack-V2 are flagged as perturbed only about 30% of the time, indicating substantially higher perceptual stealth. Notably, this M-Attack and M-Attack-V2 share a similar portion, showing that the slight differences in perturbations’  $\ell_p$  norm do not necessarily translate to the final human imperceptibility.

## H. Computational Cost and Runtime Analysis

This section analyzes the computational cost of M-Attack and related baselines, first in terms of FLOPs and then in terms of wall-clock runtime.

Let  $d$  denote the hidden dimension,  $d_{\text{ff}} = 4d$  the feed-forward expansion, and  $N$  the sequence length. In one Transformer layer, the feed-forward network (FFN) incurs  $2Nd d_{\text{ff}} = 8Nd^2$  FLOPs, while multi-head attention adds  $4Nd^2 + 4N^2d$ , giving a per-layer forward cost

$$M \triangleq 12Nd^2 + 4N^2d.$$

Since the backward pass is empirically about twice as expensive as the forward pass, a complete forward-backward iteration for a single VLM requires approximately  $3M$  FLOPs.



Table 10. Ablation study on the impact of perturbation budget ( $\alpha$ ).

$\alpha$	Method	GPT-4o				Claude 3.7-thinking				Gemini 2.5-Pro			
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
0.25	SSA-CWA (Dong et al., 2023a)	0.08	0.08	0.04	0.10	0.06	0.03	0.00	0.03	0.06	0.03	0.00	0.01
	M-Attack (Li et al., 2025)	0.62	0.39	0.09	0.71	0.12	0.03	0.01	0.16	0.55	0.33	0.08	0.55
	M-Attack-V2 (Ours)	0.86	0.61	0.21	0.96	0.43	0.28	0.5	0.52	0.82	0.29	0.18	0.89
0.50	SSA-CWA (Dong et al., 2023a)	0.10	0.10	0.04	0.07	0.08	0.04	0.00	0.05	0.09	0.05	0.00	0.04
	M-Attack (Li et al., 2025)	0.73	0.48	0.17	0.77	0.20	0.13	0.06	0.22	0.79	0.53	0.10	0.80
	M-Attack-V2 (Ours)	0.87	0.64	0.23	0.96	0.58	0.34	0.13	0.67	0.83	0.59	0.17	0.94
1.00	SSA-CWA (Dong et al., 2023a)	0.11	0.06	0.00	0.09	0.06	0.04	0.01	0.12	0.05	0.03	0.01	0.08
	M-Attack (Li et al., 2025)	0.82	0.54	0.13	0.95	0.31	0.21	0.04	0.37	0.81	0.57	0.15	0.83
	M-Attack-V2 (Ours)	0.92	0.77	0.42	0.98	0.55	0.36	0.08	0.67	0.85	0.73	0.22	0.98
1.275	SSA-CWA (Dong et al., 2023a)	0.09	0.09	0.04	0.03	0.06	0.03	0.00	0.03	0.05	0.02	0.00	0.02
	M-Attack (Li et al., 2025)	0.00	0.00	0.00	0.00	0.25	0.18	0.06	0.34	0.85	0.55	0.19	0.84
	M-Attack-V2 (Ours)	0.91	0.78	0.40	0.99	0.56	0.32	0.11	0.67	0.87	0.72	0.22	0.97

Table 11. Comparison of results on 1K images. We provide ASR based on different thresholds as a surrogate for KMR following M-Attack (Li et al., 2025).

threshold	GPT-4o		Gemini-2.5-Pro		Claude-3.7-extended	
	M-Attack	M-Attack-V2	M-Attack	M-Attack-V2	M-Attack	M-Attack-V2
0.3	0.868	0.983	0.714	0.915	0.289	0.632
0.4	0.614	0.965	0.621	0.870	0.250	0.437
0.5	0.614	0.871	0.539	0.673	0.057	0.127
0.6	0.399	0.423	0.310	0.556	0.015	0.127
0.7	0.399	0.412	0.245	0.342	0.013	0.089
0.8	0.234	0.328	0.230	0.289	0.008	0.009
0.9	0.056	0.150	0.049	0.087	0.001	0.005

Table 12. Ablation study of M-Attack-V2 under different optimizer/attack variants.

Method	Model	GPT-5 (low)				GPT-5 (medium)				GPT-5 (high)			
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
SSA-CWA (Dong et al., 2023a)	Ensemble	0.08	0.04	0.00	0.08	0.09	0.05	0.01	0.06	0.10	0.05	0.01	0.07
FOA-Attack (Jia et al., 2025)	Ensemble	0.90	0.67	0.23	0.94	0.90	0.69	0.21	0.96	0.87	0.68	0.24	0.96
M-Attack (Li et al., 2025)	Ensemble	0.89	0.65	0.25	0.98	0.85	0.61	0.16	0.96	0.80	0.60	0.20	0.93
M-Attack-V2 (Ours)	Ensemble	0.92	0.79	0.30	1.00	0.90	0.73	0.25	0.96	0.88	0.71	0.27	0.99

Table 13. Comparison on GPT-5 under three budget settings (low/medium/high).

Method	Model	GPT-4o				Claude 3.7-extended				Gemini 2.5-Pro			
		KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR	KMR <sub>a</sub>	KMR <sub>b</sub>	KMR <sub>c</sub>	ASR
M-Attack-V2-ADAM (Ours)	Ensemble	0.91	0.78	0.40	0.99	0.56	0.32	0.11	0.67	0.87	0.72	0.22	0.97
M-Attack-V2-FGSM	Ensemble	0.85	0.64	0.19	0.98	0.40	0.26	0.08	0.46	0.83	0.65	0.17	0.90
M-Attack-V2-MIFGSM	Ensemble	0.90	0.66	0.23	0.96	0.45	0.30	0.07	0.57	0.84	0.64	0.15	0.87

Exhaustively accounting FLOPs for each architecture in an  $N$ -model ensemble is impractical. Instead, we introduce an empirically measured inflation factor

$$\rho_N \triangleq \frac{\text{per-iteration FLOPs of the } N\text{-model ensemble}}{\text{per-iteration FLOPs of one CLIP-B/16}},$$

with  $\rho_1 = 1$ . Under this convention:

- **AttackVLM** costs  $3M$  FLOPs per iteration.
- **M-Attack-V1** costs  $3\rho_N M$  FLOPs per iteration.

Table 14. Cross-domain results on *PatternNet* (Li et al., 2018). We report  $\text{KMR}_a/\text{KMR}_b/\text{KMR}_c$  and ASR (higher is better). **Bold** = best in column; underline = second best. The shaded row is our method.

Method	GPT-4o				Claude 3.7				Gemini 2.5			
	$\text{KMR}_a$	$\text{KMR}_b$	$\text{KMR}_c$	ASR	$\text{KMR}_a$	$\text{KMR}_b$	$\text{KMR}_c$	ASR	$\text{KMR}_a$	$\text{KMR}_b$	$\text{KMR}_c$	ASR
AttackVLM	0.06	0.01	0.00	0.02	0.06	0.02	0.00	0.00	0.09	0.04	0.00	0.03
SSA-CWA	0.05	0.02	0.00	0.13	0.04	0.03	0.00	0.07	0.08	0.02	0.01	0.15
AnyAttack	0.06	0.03	0.00	0.05	0.03	0.01	0.00	0.05	0.06	0.02	0.00	0.05
M-Attack	<u>0.79</u>	<u>0.66</u>	<u>0.21</u>	<b>0.93</b>	<u>0.33</u>	<u>0.17</u>	<u>0.04</u>	<u>0.48</u>	<u>0.86</u>	<b>0.71</b>	<b>0.23</b>	<u>0.91</u>
M-Attack-V2	<b>0.83</b>	<b>0.71</b>	<b>0.24</b>	<b>0.93</b>	<b>0.58</b>	<b>0.40</b>	<b>0.09</b>	<b>0.73</b>	<b>0.88</b>	<u>0.68</u>	<u>0.22</u>	<b>0.97</b>

Table 15. Human study on the imperceptibility of M-Attack-V2 under different perturbation budgets. We report the proportion (%) of images identified by users; results are averaged over 10 user groups (mean  $\pm$  std).

Proportion	$\epsilon = 16$ , Mean $\pm$ Std	$\epsilon = 8$ , Mean $\pm$ Std
Adversarial images correctly identified	42 $\pm$ 1.7	27.4 $\pm$ 1.6
Original images correctly identified	98 $\pm$ 1.6	93.1 $\pm$ 2.3

- **SSA-CWA** adds an inner sampling loop of  $\hat{K}$  steps for sharpness-aware minimization (SAM), lifting the complexity to  $3\rho_N\hat{K}M$ .
- **M-Attack-V2** evaluates  $K$  local crops and, for each crop, forwards  $P$  auxiliary examples to reduce the variance of M-Attack. This gives a complexity of  $\rho_N K(3M + PM) = \rho_N K(3 + P)M$ , where  $P$  is typically a small integer (e.g.,  $P = 2$ ).

In practice, GPUs parallelize many of these operations, so wall-clock time per image does not scale linearly with FLOPs. On a single NVIDIA RTX 4090 GPU, we measure the average time per attacked image (while running a batched optimization with 32 images at the same time) as follows:

- SSA-CWA: 545.80  $\pm$  4.21 s per image;
- M-Attack-V1: 22.04  $\pm$  0.11 s per image;
- M-Attack-V2 (with  $K = 2$ ,  $P = 2$ ): 24.13  $\pm$  0.84 s per image.

Thus, with  $K = 2$ ,  $P = 2$ , M-Attack increases the runtime of M-Attack-V1 by only 9.4%, while yielding substantial gains in attack quality: on Claude 3.7, we observe improvements of +20%, +17%, +3%, and +13% on  $\text{KMR}_a$ ,  $\text{KMR}_b$ ,  $\text{KMR}_c$ , and ASR, respectively; on GPT-4o, the corresponding gains are +8%, +3%, +1%, and +6%. More results are presented in the Appx. F.1. Therefore, our M-Attack-V2 offers a configurable option between efficient yet effective and highly effective attack (as presented in the main paper).

We omit AttackVLM and AnyAttack from the empirical runtime comparison: AttackVLM reports relatively low attack performance, while AnyAttack devotes most of its computational budget to its novel pre-training stage, making its overall cost not directly comparable to inference-time attacks like ours.

## I. Additional Visualization

### I.1. Visualization of Adversarial Samples

Fig. 4 and Fig. 8 visualize adversarial samples of different black-box attack algorithms under different perturbation constraints. Under  $\epsilon = 8$ , no significant difference exists between M-Attack and M-Attack-V2. On the  $\epsilon = 16$  setting, the visual effect is still very close between M-Attack and M-Attack-V2. Since our M-Attack-V2 also greatly improves the results under  $\epsilon = 8$ , future directions might be improving the imperceptibility by adding constraints besides the  $\ell_\infty$ . We also provide all 100 images in the supplementary material for further reference.

Table 16. Proportion of adversarial images from each attack that participants judged as perturbed in Study II (all at  $\epsilon = 16$ ). Lower values indicate more perceptually stealthy perturbations.

Method	Identified as perturbed images (%)
AnyAttack	$84 \pm 4.47$
SSA-CWA	$54 \pm 8.49$
M-Attack-V1	$30 \pm 10.1$
M-Attack-V2	$32 \pm 8.0$

Table 17. Unified robustness under input-preprocessing defenses. We report  $KMR_a$ ,  $KMR_b$ ,  $KMR_c$ , and ASR ( $\uparrow$ ) for GPT-4o, Claude-3.7, and Gemini-2.5. Bold indicates the best value within each metric column for the given defense block; shaded cells highlight M-Attack-V2 (numeric cells only).

Setting	Method	GPT-4o				Claude 3.7				Gemini 2.5			
		$KMR_a$	$KMR_b$	$KMR_c$	ASR	$KMR_a$	$KMR_b$	$KMR_c$	ASR	$KMR_a$	$KMR_b$	$KMR_c$	ASR
JPEG ( $Q=75$ )	AttackVLM	0.06	0.02	0.00	0.03	0.07	0.02	0.00	0.02	0.08	0.04	0.00	0.04
	SSA-CWA	0.08	0.04	0.01	0.10	0.07	0.02	0.00	0.05	0.09	0.06	0.01	0.09
	AnyAttack	0.06	0.03	0.00	0.05	0.04	0.01	0.00	0.03	0.08	0.03	0.00	0.05
	M-Attack	0.76	0.54	0.16	0.91	0.28	0.17	0.03	0.34	<b>0.75</b>	0.51	0.11	0.76
	M-Attack-V2	<b>0.89</b>	<b>0.69</b>	<b>0.20</b>	<b>0.97</b>	<b>0.55</b>	<b>0.36</b>	<b>0.09</b>	<b>0.68</b>	<b>0.75</b>	<b>0.56</b>	<b>0.18</b>	<b>0.82</b>
DiffPure ( $t=25$ )	AttackVLM	0.05	0.02	0.00	0.01	0.05	0.02	0.00	0.01	0.08	0.03	0.00	0.01
	SSA-CWA	0.07	0.03	0.00	0.02	0.04	0.02	0.00	0.03	0.07	0.01	0.00	0.05
	AnyAttack	0.07	0.03	0.00	0.04	0.02	0.02	0.00	0.04	0.09	0.04	0.00	0.07
	M-Attack	0.42	0.20	0.03	0.43	0.10	0.05	0.01	0.10	0.39	0.22	0.01	0.32
	M-Attack-V2	<b>0.73</b>	<b>0.47</b>	<b>0.15</b>	<b>0.72</b>	<b>0.19</b>	<b>0.11</b>	<b>0.04</b>	<b>0.20</b>	<b>0.61</b>	<b>0.42</b>	<b>0.06</b>	<b>0.56</b>
DiffPure ( $t=75$ )	AttackVLM	0.08	0.05	0.00	0.02	0.04	0.02	0.00	0.00	0.04	0.01	0.00	0.01
	SSA-CWA	0.05	0.03	<b>0.01</b>	0.06	0.05	<b>0.03</b>	0.00	0.03	0.07	0.02	0.00	0.05
	AnyAttack	0.05	0.00	0.00	0.06	0.04	0.02	0.00	0.03	0.04	0.02	0.00	0.07
	M-Attack	0.10	0.02	0.00	0.04	0.03	0.02	0.00	0.02	0.05	0.05	0.00	0.05
	M-Attack-V2	<b>0.13</b>	<b>0.06</b>	<b>0.01</b>	<b>0.07</b>	<b>0.07</b>	0.02	0.00	<b>0.06</b>	<b>0.12</b>	<b>0.06</b>	<b>0.01</b>	<b>0.08</b>

Table 18. Cross-domain results on *ChestMNIST*, from MedMNIST (Yang et al., 2021). We report  $KMR_a/KMR_b/KMR_c$  and ASR (higher is better). Bold = best in column; underline = second best. The shaded row is our method.

Method	GPT-4o				Claude 3.7				Gemini 2.5			
	$KMR_a$	$KMR_b$	$KMR_c$	ASR	$KMR_a$	$KMR_b$	$KMR_c$	ASR	$KMR_a$	$KMR_b$	$KMR_c$	ASR
AttackVLM	0.06	0.01	0.00	0.03	0.05	0.02	0.00	0.02	0.08	0.03	0.00	0.02
SSA-CWA	0.06	0.03	0.00	0.15	0.04	0.03	0.00	0.07	0.08	0.02	0.01	0.14
AnyAttack	0.06	0.02	0.00	0.05	0.03	0.01	0.00	0.04	0.07	0.02	0.00	0.05
M-Attack	<u>0.89</u>	0.70	<u>0.22</u>	<u>0.92</u>	<u>0.31</u>	0.18	<u>0.07</u>	<u>0.31</u>	<u>0.85</u>	<u>0.67</u>	<u>0.23</u>	<b>0.96</b>
M-Attack-V2	<b>0.90</b>	<b>0.74</b>	<b>0.27</b>	<b>0.97</b>	<b>0.70</b>	<b>0.51</b>	<b>0.21</b>	<b>0.83</b>	<b>0.89</b>	<b>0.76</b>	<b>0.33</b>	<u>0.95</u>

## I.2. Visualization of Reasoning Models

Fig. 12 illustrates how GPT-o3 (OpenAI, 2025) responds to our adversarial samples. The model’s visual reasoning behaviors can be broadly categorized into three types: *no reasoning* (response (d)), *simple reasoning* (responses (b) and (c)), and *zoom-in reasoning* (response (a)). Notably, in response (a), GPT-o3 already identifies the central area as uncertain and zooms in on it. However, its reasoning mechanism is not well-equipped to handle adversarial perturbations, resulting in a response that remains semantically close to the target image despite the perturbation. This observation suggests that vision reasoning offers a degree of robustness by detecting uncertainty and taking subsequent actions. During training, incorporating explicit behaviors, such as refusing to answer or flagging potential adversarial inputs, could further enhance the utility of vision-based inference under adversarial conditions.



Figure 12. Visualization of GPT-o3’s response towards M-Attack-V2 adversarial samples. The underlined ‘glitchy’ denotes that O3 notices something unusual.